

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

Ruan M. Carvalho¹, Gisele G. T. Silva¹, Priscila V. Z. C. Goliatt^{1,2}, Bernardo M. Rocha^{1,2}, Carlos C. H. Borges^{1,2}

¹Graduate Program in Computational Modeling, Federal University of Juiz de Fora
ruan.medina@engenharia.ufjf.br, giselegoulart@ice.ufjf.br

²Department of Computer Science, Federal University of Juiz de Fora
capriles@ice.ufjf.br, bernardo.rocha@ice.ufjf.br, cchborges@ice.ufjf.br

Abstract. The demand for methods capable of extracting knowledge from data and organizing large amounts of information has been growing in recent decades. Images are present in several applications in biomechanics and biomedicine, and usually require the use of computational methods to reduce complexity without compromising quality. This paper proposes the application of computational methods for image compression, 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. We use a referencial way of obtaining some metrics for compression without excessive loss of information. The computational experiments in this work consisted of fixing compression metrics and evaluating the performances of DCT and SVD. 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.

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. Autores como Sarkar [1] defendem que o desenvolvimento e aplicação de métodos da álgebra linear computacional fomentam áreas como a Ciência de Dados, provendo o ferramental para organizar e avaliar a quantidade de informação presente nos dados coletados. 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 perda excessiva da 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 estatística e aprendizado baseado em dados. Contudo, a avaliação do desempenho dos métodos em imagens tem como vantagem a visualização direta e intuitiva dos resultados obtidos pelas transformações associadas. Neste artigo investigam-se métodos computacionais para compressão de imagens, considerando questões de armazenamento, transferência de dados e reconhecimento de padrões. Em particular, apresenta-se um estudo comparativo entre as técnicas de Decomposição em Valores Singulares (SVD) e Transformada Discreta de Cossenos (DCT).

Existem diferentes técnicas para compressão de imagens, portanto, estudos envolvendo comparações entre as diferentes abordagens são comuns na literatura. O trabalho de Majumder and Hussain [2], em específico, apresenta um estudo comparativo completo entre métodos que envolve testes envolvendo os métodos DCT e SVD. Mais recentemente, trabalhos como os de Sadek [3] apresentaram as evoluções e modificações consideradas para o estado da arte do SVD, e trabalhos como o de Banday and Shah [4] desenvolvem novas abordagens para a utilização do DCT para compressão de imagens, inclusive avaliando 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.

2 Material e Métodos

2.1 Decomposição em Valores Singulares (SVD)

A Decomposição em Valores Singulares, Stewart [5] e Chen [6], consiste em uma fatoração de uma matriz \mathbf{M} , $m \times n$, real ou complexa em três matrizes, de acordo com a eq. (1), onde \mathbf{U} é uma matriz $m \times m$ ortogonal se real (ou unitária se complexa), $\mathbf{\Sigma}$ 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). São denominados valores singulares de \mathbf{M} os valores na diagonal de $\mathbf{\Sigma}$. 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} , respetivamente. Nessa construção, os valores singulares são ordenados do maior para o menor valor na diagonal principal de $\mathbf{\Sigma}$, ou seja, $\sigma_1 > \sigma_2 > \dots > \sigma_t$; $t = \min\{n, m\}$.

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

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} , $\mathbf{\Sigma}'$ e \mathbf{V}^* , onde $\mathbf{\Sigma}'$ possui dimensões menores que $\mathbf{\Sigma}$. Além disso, geralmente poucos dos valores singulares da decomposição tem valores relevantes, ou seja, relativamente altos. Por isso, mesmo que as imagens originais tenham posto completo, muitas dimensões que compõem esse posto não agregam informação de relevância da imagem, como explicitado em Sadek [3]. Com isso, em uma compressão, representa-se a matriz \mathbf{M} como uma combinação linear de matrizes de posto 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, como mostrado em Easton [7]. A Transformada Discreta de Cossenos (DCT), proposta por Ahmed et al. [8], é a versão discreta da parte real da Transformada de Fourier. A versão bidimensional do DCT, para uma matriz \mathbf{M} $n \times n$, é determinada pela eq. (2). 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, como descrito por Watson [9].

Esse processo é vastamente aplicado em compressões de dados, pois posiciona os coeficientes mais significativos no início da matriz transformada. Desta forma, valores que são muitos pequenos, carregam pouca informação da imagem e podem passar por um processo de eliminação, como proposto por Sayood [10]. Assim, os componentes mais relevantes são os referentes a parcela superior esquerda da matriz transformada.

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

$$\text{Onde: } C(f) = \begin{cases} \frac{1}{\sqrt{2}} & ; \text{se } f = 0 \\ 1 & ; \text{se } f > 0 \end{cases}$$

2.3 Descrição dos benchmarks de imagens

BDI: The USC-SIPI Image Database

Como primeira fonte de imagens para testes, o banco de imagens USC-SIPI (<http://sipi.usc.edu/database/>) é um conjunto de imagens para apoio em pesquisas de processamento, 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 imagens com caráter básico. Todas as imagens utilizadas foram consideradas em escala de cinza com 8 bits para representação das tonalidades. As imagens utilizadas nos experimentos são provenientes do Volume 3: *Miscellaneous*, consistindo em 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 imagens mosaico do Volume 1: *Textures* foram adicionadas à base para refinamento dos testes. A Fig. 1 apresenta cinco exemplos de imagens da base.

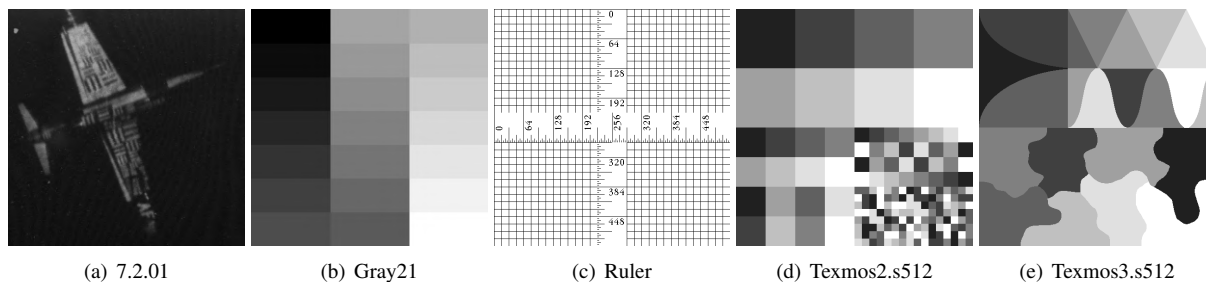


Figura 1. Exemplos de Imagens Utilizadas da Base USC-SIPI.

BD2: Rawzor Image Compression Benchmark

O banco de imagens proposto por Rawzor [11] (<http://imagecompression.info/>) apresenta-se como uma alternativa às imagens clássicas utilizadas em pesquisas de compressão ao disponibilizarem imagens com resoluções superiores e escolhidas de forma técnica, a fim de 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 e 16 bits, em padrões RGB e cinza. Para esse trabalho, foram consideradas apenas as versões das imagens em tons de cinza. A Fig. 2 apresenta alguns exemplos das imagens utilizadas.



Figura 2. Exemplos de Imagens Utilizadas da Base de Rawzor [11].

2.4 Descrição dos experimentos computacionais

As implementações e os resultados foram obtidos por meio de um *script* desenvolvido em *Python3.5*. A fim de obter um código otimizado para o fluxo de execução, 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 em SVD).

As comparações são realizadas a fim de gerar avaliar os desempenhos dos métodos para as imagens considerando métricas de erro médio, como utilizadas por Easton [7] e Putalapattu [12]. 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 eq. (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 (*Compression rate - Cr*) a razão entre o número de posições de memória utilizados pela matriz original em comparação com a matriz após a redução. A eq. (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 (matricial) 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 se reflete no valor do erro ϵ .

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

Pretende-se, ainda, 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 uma referência 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_i) de uma matriz \mathbf{M} nos indica a Quantidade de Informação ($Qinf$) da imagem, como visto na eq. (5).

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

Com isso é possível definir um percentual da quantidade de informação desejada nas compressões e encontrar o número de dimensões necessárias para tal representação [6]. Visto isso, o Percentual de Quantidade de Informação Representada ($\%Qinf_M$), descrito na eq. (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 \mathbf{Mc} 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 calculado somando gradativamente os valores singulares obtidos na fatoração SVD, e assim podemos estimar o erro da compressão em termos relativos à imagem.

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

A Fig. 3 apresenta o fluxograma de execução seguido para a realização dos testes. Por meio desse fluxo é possível variar a medida de percentual de informação representada na compressão que se é desejada e obter comparações entre os métodos com relação às métricas de erro consideradas.

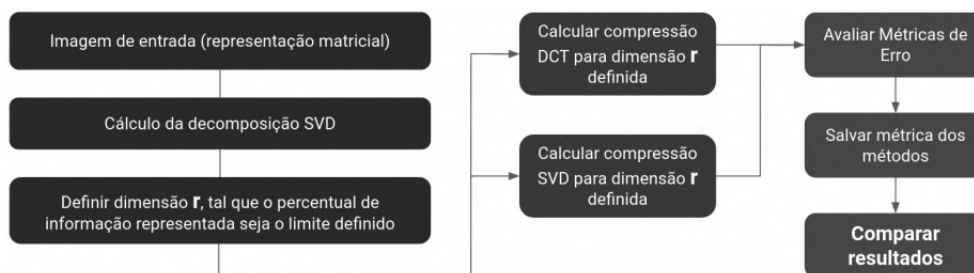


Figura 3. Fluxograma de execução dos testes.

3 Resultados e Discussão

Nesta seção, são apresentados os resultados para os experimentos computacionais descritos anteriormente. Para as imagens das bases BD1 e BD2, a execução do *script* de compressão foi realizada variando $\%Qinf_M \in \{0, 5; 0, 6; 0, 7; 0, 8; 0, 9\}$. A Fig. 4 apresenta as distribuições da métrica RMSE para as duas bases de dados testadas. Para BD1, observa-se alguns *outliers* alinhados para $\%Qinf_M \in \{0, 5; 0, 6; 0, 7; 0, 8\}$ para o método DCT. Esses *outliers* apresentam valores de RMSE próximos de 150, e o aumento de $\%Qinf_M$ não melhora o valor de RMSE de forma significativa. Esses *outliers* foram gerados como resultado de uma mesma figura (*Gray21*), e há diminuição do RMSE apenas em $\%Qinf_M = 0, 9$. A segunda observação de *outliers* alinhados para BD1 acontece com valores de RMSE próximos de 80 (*Ruler*) comprimida pelo método DCT. No caso da BD2, foram calculados *outliers* com valores próximos de 90 em todos os percentuais de informação testados, sendo esses referentes a compressão da imagem *Zone Plate* pelo método DCT. Para os valores de $\%Qinf_M \in \{0, 5; 0, 6; 0, 7\}$ foram encontrados *outliers* também referentes a imagem *Zone Plate*.

A partir da Fig. 4 também percebe-se uma tendência de decaimento da variância no conjunto de medidas com o aumento de $\%Qinf_M$, para ambos os métodos. Em ambas as bases de dados, os valores de RMSE para o SVD são menores do que os encontrados pelo DCT e os intervalos de variação dos *boxplots* do SVD são menores que os do método DCT, indicando que o método SVD consegue representar a imagem mais claramente com menor quantidade de informação. Vale ressaltar que, apesar do SVD apresentar menores valores de RMSE em todos os $\%Qinf_M$, mais memória é utilizada durante seu processo de compressão (como veremos a seguir).

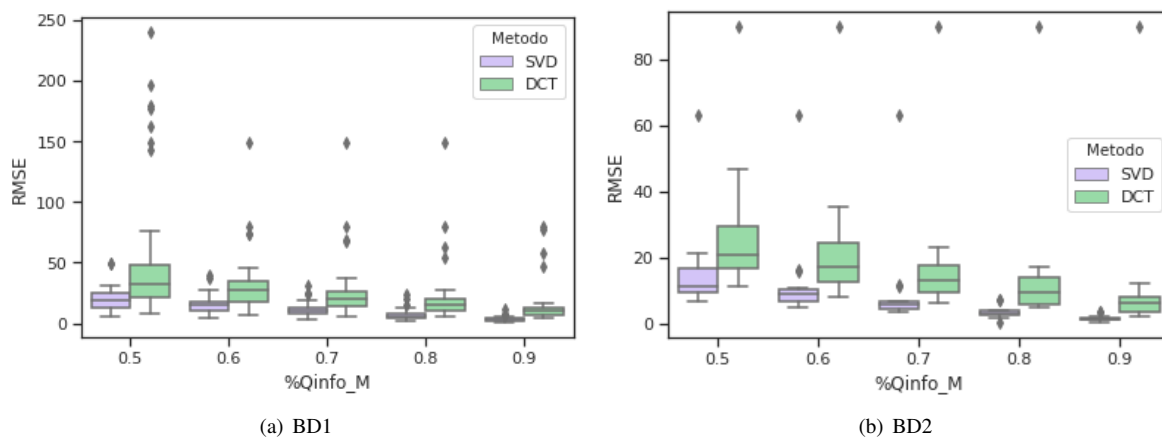


Figura 4. Distribuição dos RMSE obtidos para as compressões de realizadas por cada método com a variação dos percentuais de informação representada de cada imagens dos benchmarks.

A Fig. 5 apresenta os gráficos das distribuições das taxas de compressão obtidas pelos métodos. Para BD1, as Fig. 5 (a) e (b) mostram um decaimento da variação da taxa de compressão a medida que $\%Qinf_M$ aumenta. Porém, comparando os valores das medianas de Cr obtidas por SVD e DCT em $\%Qinf_M = 0,5$; pode-se perceber uma diferença significativa de valores. Isto ocorre pela forma como os métodos são implementados, de modo que o DCT alcança altas taxas de compressão, porém resultando em valores mais altos de RMSE. Importante ressaltar que, para a BD1, os boxplots foram apresentados sem os outliers, pois, para a imagem *Ruler* em $\%Qinf_M = 0,5$, o método DCT apresentou valores de Cr próximos de $2,6 \times 10^5$, comprometendo a visualização dos gráficos.

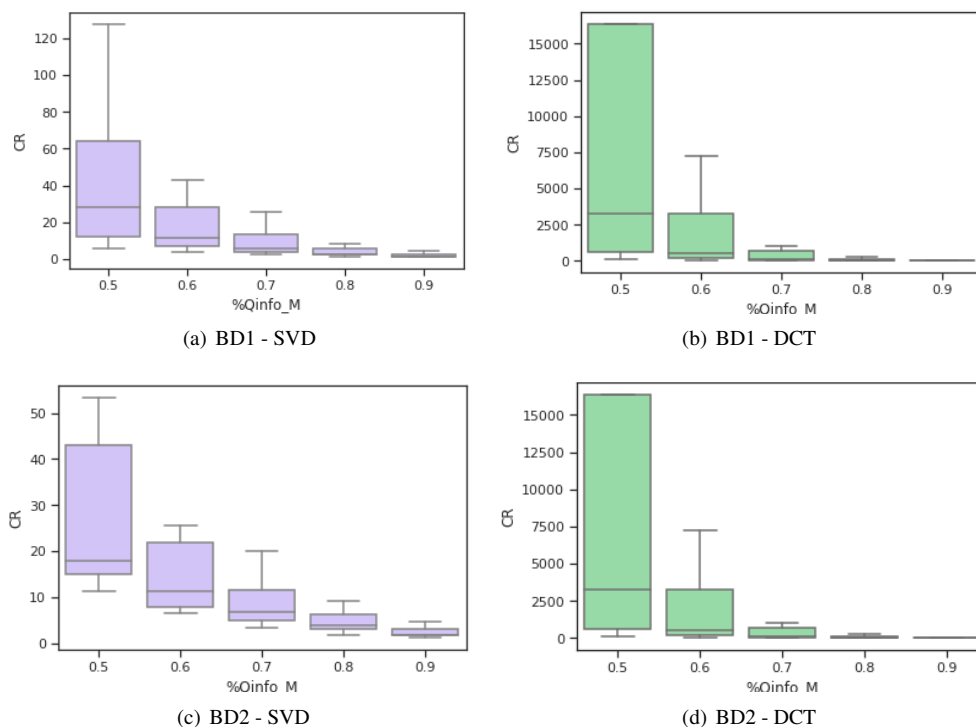


Figura 5. Distribuição para as taxas de compressão (CR) obtidas para as compressões realizadas por cada método. Os percentuais de informação representada foram variados de 0,5 a 0,9.

Para a BD2, as Fig. 5 (c) e (d) mostram como ocorre o decaimento do Cr a medida que a quantidade de informação aumenta. A ordem de compressão entre os métodos também foi considerável nessa base. Os boxplots também foram apresentados excluindo os outliers, pois a imagem *Zone Plate* para $\%Qinf_M = 0,5$, obteve altos valores de Cr no DCT e SVD, aproximadamente 1×10^6 e $5,9 \times 10^2$, respectivamente.

No geral, o intervalo de variação da métrica RMSE para os métodos testados foi próximo, de modo que, apesar dos *outliers*, os valores de erro para uma dada imagem não divergem. Apesar disso, quando consideramos um mesmo $\%Qinf_M$, a qualidade e nitidez das imagens comprimidas através do SVD é bastante superior. A medida que aumentamos $\%Qinf_M$, essa discrepância na qualidade visual das imagens diminui, pois começamos a obter imagens mais nítidas e com mais informação. Porém, ainda é possível perceber que a compressão realizada pelo SVD retorna imagens com maior qualidade, e a compressão do DCT resulta em imagens com altíssimas taxas de compressão (Cr) quando comparadas ao SVD.

Quando tratamos de compressão de imagens, é importante evitar a perda excessiva de informação durante o processo, de forma que a imagem ainda seja útil dentro de sua aplicação. Porém, é importante ter em mente que, por se tratar de um processo de compressão, é preciso garantir que o dado final ocupe menos espaço em memória que o dado original. Para o DCT a discussão é simplificada, uma vez que o processo apenas transforma a matriz original para a redução posterior de sua dimensionalidade. Assim, sempre que reduzirmos o posto da matriz transformada, estamos reduzindo o espaço em disco utilizado. Já para o SVD é importante ressaltar que o processo de fatoração substitui uma matriz (M) por três matrizes ($U\Sigma V^*$). Assim, para uma compressão efetiva é preciso que a redução de dimensionalidade justifique tal representação. Seja a matriz original quadrada $m \times m$, assim são necessários m^2 posições de memória para armazenar essa imagem. Aplicando o SVD na matriz original e utilizando r valores singulares na representação reduzida, são necessários $2mr + r$ de posições de memória. Tal número de posições é resultante das duas matrizes cheias U e V^* e da diagonal da matriz Σ . Para que o uso de memória seja reduzido neste processo, o valor de r deve ser tal que: $2mr + r < m^2 \Rightarrow r(2m + 1) < m^2 \Rightarrow \boxed{r < \frac{m^2}{2m+1}}$. Por exemplo, considerando uma imagem de resolução 512×512 pixels, para que haja uma compressão mínima dos dados na decomposição SVD, r deve possuir valor máximo de 254. Ou seja, no máximo, apenas 254 posições da diagonal principal da matriz Σ deverão ser utilizadas, descartando parte da informação original constante na imagem.

A Fig. 6 apresenta os piores e melhores resultados para as bases BD1 e BD2 considerando a métrica RMSE. Na BD1 o método SVD apresentou imagens de melhor e pior casos com características próximas, possuindo alguns retângulos com tonalidades diferentes. Para o DCT, além das duas imagens originais possuírem características distintas, no pior caso não é possível visualizar nenhuma característica nítida da imagem original. Para a BD2, o melhor e pior resultados do SVD aconteceram na mesma imagem, com diferença de apenas uma unidade em r , indicando que grande parte da informação desta imagem se encontra nas posições iniciais da matriz Σ . Novamente, na imagem de pior caso do DCT na BD2 não é possível reconhecer visualmente características originais da imagem.

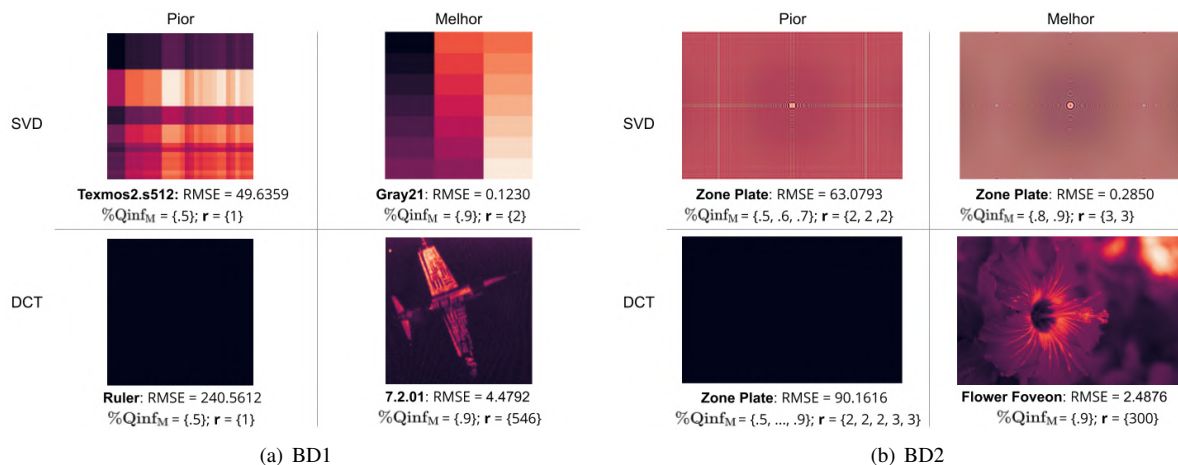


Figura 6. Piores e melhores resultados obtidos por cada método tendo como base o RMSE descrito na eq. (3).

A Fig. 7 mostra os piores e melhores resultados encontrados com as bases considerando a métrica de compressão obtida. Na BD1, o melhor caso de Cr para SVD e DCT ocorreu na mesma imagem (*Ruler*), mas na imagem apresentada pela DCT não é possível visualizar nenhuma características nítida da imagem original. Os piores valores de Cr na BD1 também ocorreram na mesma imagem, porém, neste caso conseguimos identificar o objeto apresentado em ambas as imagens. Analogamente a BD1, na BD2, o melhores Cr dos métodos ocorrem com a mesma imagem (*Zone Plate*) e na compressão realizada pelo DCT não é possível identificar características da imagem original. Considerando os piores Cr na BD2, o piores casos para SVD e DCT ocorrem na mesma imagem e ambas conseguem representar com nitidez a informação original.

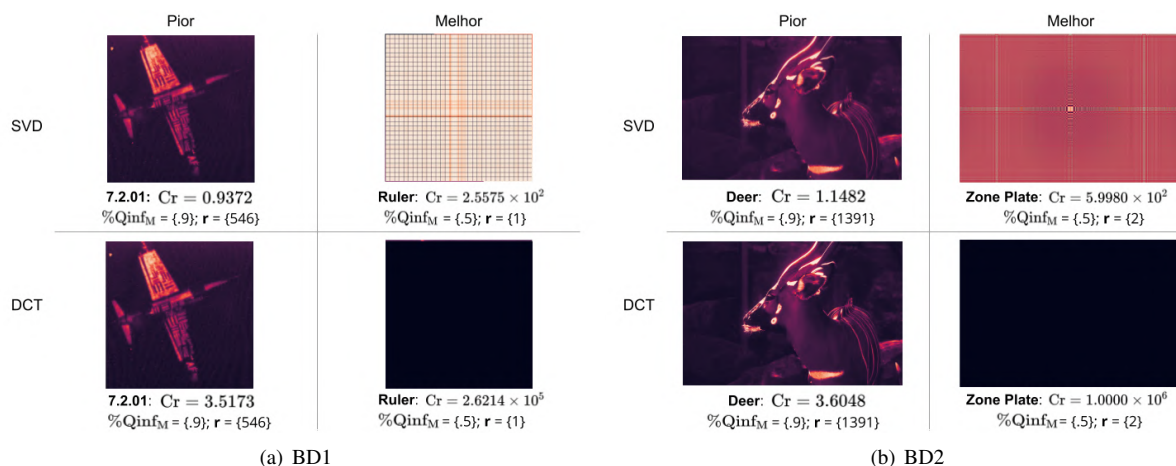


Figura 7. Piores e melhores resultados obtidos por cada método tendo como base o Cr descrito na eq. (4).

4 Conclusão

O presente trabalho apresentou um estudo comparativo entre os métodos SVD e DCT aplicados na compressão de imagens. Utilizou-se de uma abordagem baseada em valores singulares da matriz da imagem para definir um percentual de quantidade de informação das compressões. Tal metodologia se mostrou aplicável, contudo favorece o método SVD ao tender a altas taxas de compressão do DCT. Os resultados demonstram que o DCT apresenta melhor performance na redução no uso de memória na compressão, contudo resulta em imagens nem sempre nítidas. Contudo, ambos os métodos se mostraram aplicáveis a depender do contexto. O DCT realiza uma compactação com perdas e com redução do nível de nitidez, e tem sido atualmente utilizada em uma das etapas do processo de compactação do padrão JPEG. Sua aplicação possibilita detectar redundância espacial ao longo da imagem, identificando a semelhanças na vizinhança de determinado *pixel*. O SVD distribui os coeficientes mais significativos ao longo da matriz diagonal Σ em ordem decrescente, explorando a compactação da imagem através da representação de frações desses coeficientes. Nos experimentos realizados, o SVD se mostrou particularmente eficiente para compressão de imagens com altos níveis de simetria e padrões de formação. Por conta das restrições impostas no uso do SVD para que haja redução no uso de memória, o DCT tem sido mais utilizado em aplicações de compressão de imagens.

Agradecimentos. Este trabalho foi financiado pela Universidade Federal de Juiz de Fora (UFJF), FAPEMIG e pela Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES).

Declaração de autoria. Os autores confirmam que são as únicas pessoas responsáveis pela autoria deste trabalho, e que todo o material aqui incluído como parte deste artigo é de propriedade (e autoria) dos autores ou tem a permissão dos proprietários a serem incluídos aqui.

Referências

- [1] Sarkar, T., 2018. Essential math for data science - 'why' and 'how'.
- [2] Majumder, S. & Hussain, M., 2008. A comparative study of img. compression tech.: Svd, dwt-svd and dwt-dct.
- [3] 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*, volume 3, pp. 26–34.
- [4] Bandy, M. T. & Shah, T., 2013. A comparative study of image compression techniques.
- [5] Stewart, G., 1993. On the early history of the singular value decomposition. In Siam, ed, *Society for Industrial and Applied Mathematics*, volume 35, pp. 551–566.
- [6] Chen, J., 2000. Image compression with svd. ecs 289k scientific computation.
- [7] Easton, J. R. L., 2010. *Fundamentals of Digital Image Processing*. Jr. Roger L. Easton, online edition.
- [8] Ahmed, N., Natarajan, T., & Rao, K. R., 1974. Discrete cosine transform. *IEEE transactions on Computers*.
- [9] Watson, A. B., 1994. Image compression using the dct. *Mathematica journal*, vol. 4, n. 1, pp. 81.
- [10] Sayood, K., 2017. *Introduction to data compression*. Morgan Kaufmann.
- [11] Rawzor, Image Compression-Benchmark. <http://imagecompression.info/>. Acesso em 15 de agosto de 2020.
- [12] Putalapattu, R., 2017. Jupyter, python, image compression and svd. an interactive exploration.