

AULA PRÁTICA 6 IMAGENS E SVD

Consideramos uma imagem em tons de cinza (*grayscale* em inglês) onde cada píxel é codificado por um número inteiro de 0 a 255. A imagem é então representada por uma matriz de inteiros com m linhas e n colunas. As funções Scilab implementadas serão testadas com as imagens disponibilizadas no eClass: cada aluno deverá escolher pelo menos uma imagem correspondente a um rosto e todos deverão fazer também com a imagem *'marinha.png'*. Além das duas imagens obrigatórias, podem usar outras imagens em preto e branco à vontade: fotos pessoais, desenhos, etc.

- 1) **Visualização.** Faça a leitura da imagem usando o comando Scilab *A=imread('imagem.png')* (a imagem também pode estar no formato *.jpg*) A matriz A será então uma matriz $m \times n$ com elementos inteiros de 0 a 255. Escreva uma função Scilab chamada *visualization* que recebe uma matriz A representativa de uma imagem em tons de cinza e mostra essa imagem na tela. Use os comandos *subplot* e *imshow*.
(Veja os “helps” desses comandos no Scilab).
- 2) **Compressão e SVD.** A decomposição em valores singulares pode comprimir uma imagem. Usando a função do Scilab *svd* que fornece a decomposição em valores singulares de uma matriz A (Veja o “help” desse comando no Scilab) e sendo r o número de valores singulares positivos, podemos comprimir a imagem representada pela matriz A usando apenas os s ($s < r$) maiores valores singulares de A , conforme visto em aula.

Escreva uma função Scilab com variáveis de entrada A e p , onde $0 < p < 1$ representa um percentual do número de valores singulares positivos de A a serem utilizados, para fazer uma compressão da imagem representada por A .

Faça *s=max(1; o maior inteiro menor ou igual a pr)* e use *[U,S,V]=svd(A)*. Para isso, você precisará primeiro converter os elementos de A para reais. Use o comando *double* do Scilab. (Veja o “help” desse comando no Scilab).

Testar esta função de compressão de dados com vários valores de p : 0.05, 0.10, 0.15, 0.20, 0.25, 0.30, ... Em cada caso visualize lado a lado a imagem original e a comprimida: *C=U(:,1:s)*S(1:s,1:s)*V'(1:s,:); //s é um inteiro entre 1 e r.*

Você precisará reconverter os elementos de C para inteiros de 0 a 255. Para isso, use a função Scilab *C=iconvert(C,11); //converte para o format inteiro com 1 byte (0 a 255).* (Veja o “help” desse comando no Scilab).

A partir de que valor de p a imagem comprimida é suficientemente “boa” comparada com a original?