



UNIVERSIDADE FEDERAL DA PARAÍBA – UFPB
CENTRO DE INFORMÁTICA – CI
CIÊNCIA DA COMPUTAÇÃO

RELATÓRIO DO TRABALHO PRÁTICO II
Introdução ao Processamento Digital de Imagens

PEDRO JOÃO DA SILVA JÚNIOR – 20190112416

INTRODUÇÃO

Este relatório consiste em detalhar o desenvolvimento do segundo trabalho prático realizado na disciplina de Introdução ao Processamento Gráfico de Imagens. O trabalho proposto tem como objetivo o processar imagens e áudios utilizando a Transformada Discreta de Cosseno (DCT) para transformar imagens e áudios do domínio espacial para o domínio da frequência.

A DCT é uma classe de operações matemáticas que tem como objetivo pegar um sinal e transformá-lo de um tipo de representação para outro. Por exemplo, uma imagem é um sinal bidimensional que é percebido pelo sistema visual humano. A DCT pode ser usada para converter o sinal (informação espacial) em dados numéricos (informações de "frequência" ou "espectral") para que a informação da imagem exista em uma forma quantitativa que possa ser manipulada por aplicações de filtros ou técnicas de compreensão de dados. As fórmulas da DCT e sua inversa, IDCT, são representadas abaixo:

$$x[n] = \left(\frac{2}{N}\right)^{1/2} \sum_{k=0}^{N-1} c_k X[k] \cos\left[2\pi \frac{k}{2N} n + \frac{k\pi}{2N}\right], \quad n = 0, 1, \dots, N-1$$

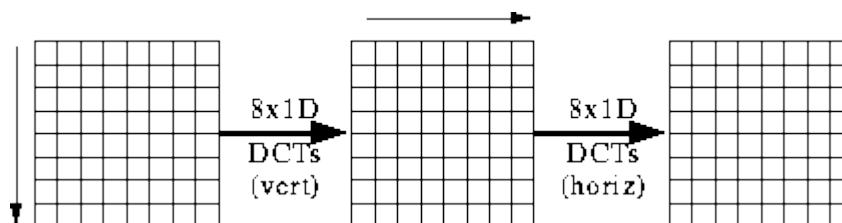
$$X[k] = \left(\frac{2}{N}\right)^{1/2} c_k \sum_{n=0}^{N-1} x[n] \cos\left[2\pi \frac{k}{2N} n + \frac{k\pi}{2N}\right], \quad k = 0, 1, \dots, N-1$$

Já as fórmulas da DCT-2D e a IDCT-2D são representadas de forma semelhante:

$$X[k, l] = \frac{2}{\sqrt{RC}} c_k c_l \sum_{m=0}^{R-1} \sum_{n=0}^{C-1} x[m, n] \cos\left[\frac{(2m+1)k\pi}{2R}\right] \cos\left[\frac{(2n+1)l\pi}{2C}\right]$$

$$x[m, n] = \frac{2}{\sqrt{RC}} \sum_{k=0}^{R-1} \sum_{l=0}^{C-1} c_k c_l X[k, l] \cos\left[\frac{(2m+1)k\pi}{2R}\right] \cos\left[\frac{(2n+1)l\pi}{2C}\right]$$

A partir do princípio da paridade, é possível construir a DCT-2D de uma representação espacial 2D a partir da DCT unidimensional, de forma que é calculada a DCT na vertical e, posteriormente, na horizontal da matriz resultante da DCT vertical.



DESENVOLVIMENTO

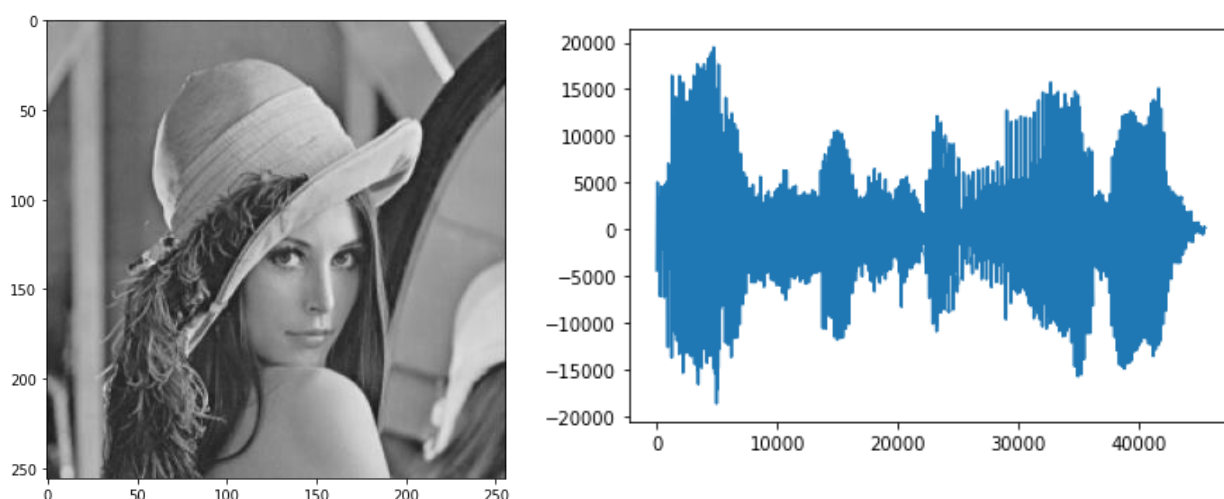
Para este trabalho é requerido que as DCT, DCT 2-D e suas inversas sejam implementados utilizando as fórmulas apresentadas. No desenvolvimento da DCT bidimensional foi utilizado o conceito da paridade, de forma não foi necessário implementar versões diferentes do cálculo da DCT para ambas as dimensões 1D e 2D.

IMPLEMENTAÇÃO

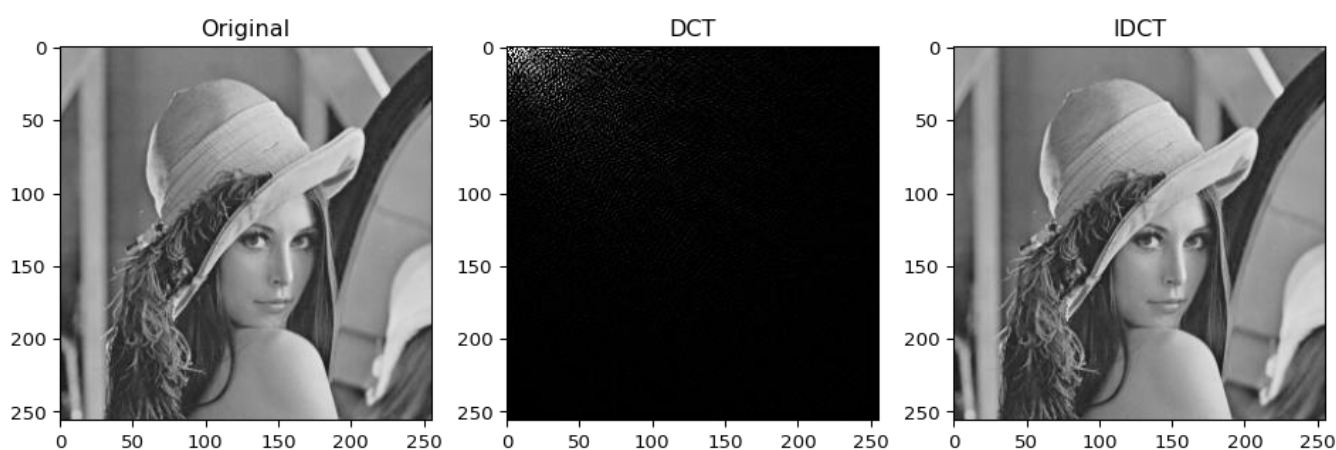
Este trabalho foi desenvolvido em Python no ambiente Jupyter Notebook e utilizando algumas bibliotecas para auxiliar na implementação das questões propostas. As bibliotecas utilizadas foram:

- Numpy – usado em processamento numérico de vetores e matrizes.
- Scipy – utilizado para ler/salvar arquivos de áudio locais em formato .wav (módulo `scipy.io.wavfile`).
- PIL – utilizado para abrir imagens locais e transformar imagem em matriz e vice-versa.
- Matplotlib – utilizado para exibir output de processamentos realizados e plotagem de gráficos.

Para testes, foi disponibilizado uma imagem de dimensões 256x256 de Lena e um arquivo de áudio em formato .wav.



DCT-2D E IDCT-2D de Lena



1.0 MÓDULO NORMALIZADO E VALOR NÚMÉRICO DO NÍVEL DC DA IMAGEM

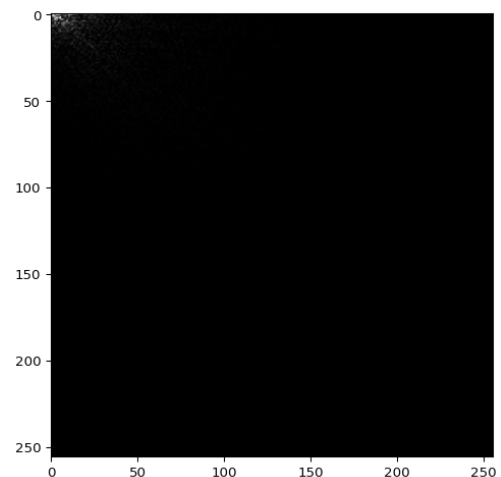
Nessa questão é requerido a exibição do módulo normalizado com uma expansão de histograma para o intervalo de 0 a 255 sem o nível DC e, também, exibir o valor numérico do nível DC de Lena.

Sabendo que o nível DC (Direct Current ou Corrente Direta) é o primeiro coeficiente da DCT e todo os outros são coeficientes AC (Alternate Current ou Corrente Alternada), é possível exibir o valor numérico DC da DCT-2D de Lena acessando o coeficiente de index [0, 0] da matriz DCT. Uma vez que o index do nível DC é conhecido, podemos zerá-lo para então obter a o modulo seguido de uma normalização através da técnica de expansão de histograma para o intervalo de 0 a 255.

Resultado obtido:

Valor numérico do nível DC: 31883.60937500001

É observável que a concentração de coeficientes não nulos é bastante reduzida para o canto superior esquerdo após o módulo da DCT ter sido normalizada.

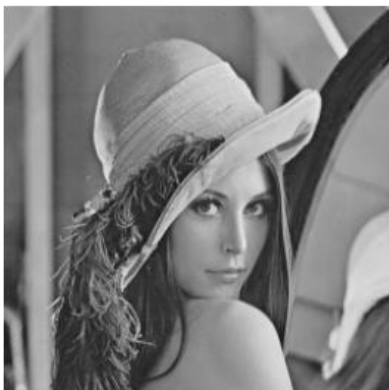


1.1 APROXIMAÇÃO PRESERVANDO O COEFICIENTE DC E OS N AC COEFICIENTES MAIS IMPORTANTES DA IMAGEM

Nesta questão é necessário exibir uma aproximação de Lena preservando o coeficiente de nível DC e os N coeficientes AC mais importantes da imagem. O parâmetro N é um valor inteiro que varia entre 0 a $RxC-1$. Para encontrar uma aproximação de Lena dado um N qualquer entre o intervalo permitido, foi feito uma iteração na DCT-2D da imagem onde todos os coeficientes AC cujo modulo é menor que N é zerado.

Resultados obtidos com algumas variações de N:

N >= 0



N >= 50



N >= 100



É bastante perceptível que à medida que o valor de N aumenta, a qualidade da imagem

resultante decair. Isso ocorre porque ao aumentar o valor de N, estamos matando cada vez mais coeficientes AC que têm informações da imagem.

2.0 FILTRO BUTTERWORTH PASSA-BAIXAS APLICADO EM IMAGEM

A aplicação do filtro Butterworth passa-baixas no domínio da frequência de imagens é alcançada a partir de sua função de transferência.

$$H(d(k,l)) = \frac{1}{\sqrt{1 + \left(\frac{d(k,l)}{f_c}\right)^{2n}}}$$

Onde $d(k,l)$ é a distância euclidiana do coeficiente (k, l) até a origem, f_c é a distância de corte até a origem e n é a ordem do filtro com $n \geq 1$ que define o nível de atenuação de frequência do filtro a partir de sua origem.

A implementação da aplicação do filtro começa pela definição das coordenadas do centro da DCT. Sendo $C = (W/2, H/2)$ onde W e H são as dimensões da DCT-2D da imagem. Uma vez que as coordenadas do centro são conhecidas, é feita a construção do filtro a partir de uma iteração sobre as dimensões da DCT-2D onde que para cada valor de index $[k, l]$ é calculada a distância euclidiana $D = \sqrt{\text{pow}(cx - \text{abs}(1 - cx), 2) + \text{pow}(cy - \text{abs}(k - cy), 2)}$, em seguida os parâmetros são postos na função de transferência do filtro para cada posição de index $[k, l]$. Após a construção do filtro, é feita a multiplicação com a DCT-2D original para gerar a DCT-2D filtrada.

Resultados obtidos com algumas variações de frequência de corte e ordem do filtro:



Percebe-se que diminuir a distância de corte (f_c) causa um efeito bastante semelhante a um filtro médio aplicado no domínio espacial da imagem. Já a ordem de filtro (N) faz com que o efeito fique mais intenso. Também é possível perceber o fenômeno de Gibbs quando usado um valor de ordem de filtro alto.

2.0 FILTRO BUTTERWORTH PASSA-BAIXAS APLICADO EM ÁUDIO

Assim como na questão acima, o filtro Butterworth passa-baixas deve ser aplicado no domínio da frequência, mas dessa vez em um sinal de áudio em formato .wav. A função de transferência é dada por:

$$H(f) = \frac{1}{\sqrt{1 + \left(\frac{f}{f_c}\right)^{2n}}} \quad \text{Onde } f \text{ é a frequência em Hz, } f_c \text{ é a frequência de corte em Hz e } n \text{ é a ordem do filtro com } n > 1.$$

Ao ler um arquivo de áudio .wav com o módulo wavfile da biblioteca Scipy, é retornado um *array* contendo os sinais de áudio e a taxa de amostragem (quantidade de amostras de um sinal analógico coletadas em uma determinada unidade de tempo). A partir da informação da taxa de amostragem e o número de amostras do sinal, podemos calcular F1 (frequência do cosseno de índice 1 em Hz ao qual todas as outras frequências dos cossenos são múltiplas de F1.), com o valor conhecido de F1 podemos obter os coeficientes da DCT e os parâmetros do filtro medidos em Hz.

A aplicação do filtro Butterworth passa-baixas no sinal de áudio foi realizada no domínio da frequência, onde cada coeficiente da sua DCT é multiplicado pelo resultado da função de transferência do filtro com os parâmetros f_c e f medidos em Hz.

DIFICULDADES

Uma dificuldade encontrada foi em relação à performance das implementações da DCT e IDCT, pois são extremamente lentas e inviáveis na prática. Foi feita algumas tentativas de otimização, mas não foi alcançado nenhum ganho de performance muito relevante. Também foi utilizado a técnica de DCT-2D utilizando blocos 8x8, porém, houve problemas ao aplicar o filtro Butterworth passa-baixas no domínio da frequência com essa abordagem. Por fim, nenhuma otimização foi feita.