Introdução ao Processamento Digital de Imagens

Emmanuella Faustino Albuquerque
20170002239

Módulo 2 do Trabalho Prático

8 de junho de 2022

VISÃO GERAL

Qualquer sequência numérica pode ser vista como um sinal digital, assim como qualquer matriz matemática pode ser vista como uma imagem digital. Transformar para domínio da frequência faz com que alguns problemas sejam resolvidos de forma mais fácil, o objetivo deste trabalho é familiarizar-se com o tema Transformada Cosseno Discreta direta (DCT) e inversa (IDCT) em 1D (uma dimensão) e em 2D (duas dimensões).

Dessa forma, neste trabalho, foram implementadas as seguintes funcionalidades: a conversão para o domínio da frequência com a (DCT) e sua inversa (IDCT). Assim como, a aplicação em duas dimensões utilizando o conceito de separabilidade. Além disso, foi implementado o filtro Butterworth passa baixas para aplicação em imagens e áudios.

ATIVIDADES DESENVOLVIDAS

Nesta seção, serão apresentadas as atividades desenvolvidas, isto é, a descrição das funcionalidade implementadas e os resultados obtidos.

1. Transformada Cosseno Discreta (DCT e IDCT)

Fundamentação Teórica

DCT é uma técnica de transformação amplamente utilizada no processamento de sinais e compressão de dados. Ela expressa uma sequência finita (sinal) em termos de uma soma de funções de cosseno, tal técnica é utilizada para compressão de imagens (ex.: PNG), vídeo e áudio (ex.: MP3).

Estratégia

➢ DCT 2D

Considerando dois eixos de frequências (horizontal, vertical) (sinal bidimensional), podemos utilizar a operação separável para trabalharmos com a transformada em imagens.

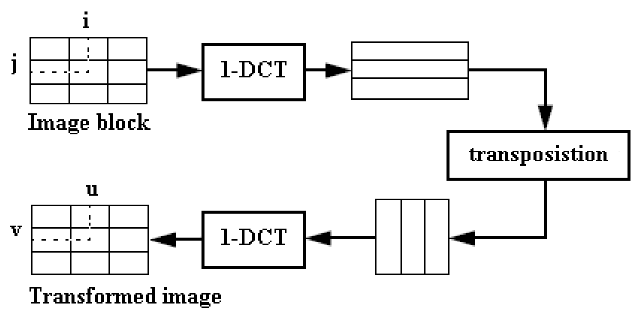
DCT 1D de x[n] (obs.: no geral a natureza nos dá x[n])

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

IDCT 1D de X[k]

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

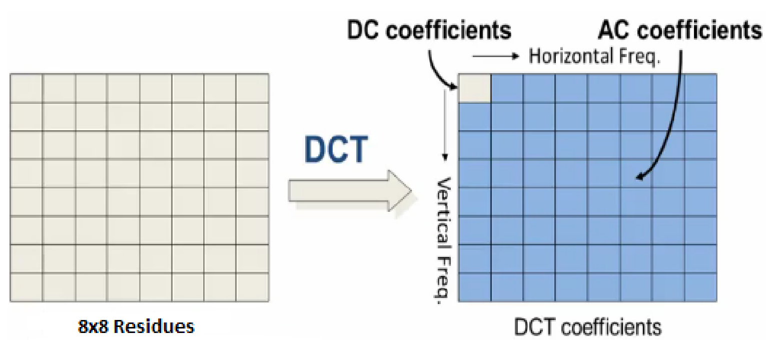
Dessa forma, para realizar essa operação (DCT e IDCT 2D) podemos aplicamos a DCT 1D nas linhas e depois nas colunas, conforme mostra a figura abaixo:



➤ **Aproximação de I (Imagem)**

A importância dos cossenos (dado pela Amplitude) tende a decrescer à medida que a frequência dos cossenos aumenta, assim, concentra-se maximamente a importância dos cossenos nas mais baixas frequências.

Sabendo disso, jogar fora os cossenos que tem amplitude mais baixa produz a melhor compressão possível para uma certa distorção, ou a menor distorção possível para uma certa compressão. Dessa forma, pequenos componentes de alta frequência podem ser descartados.

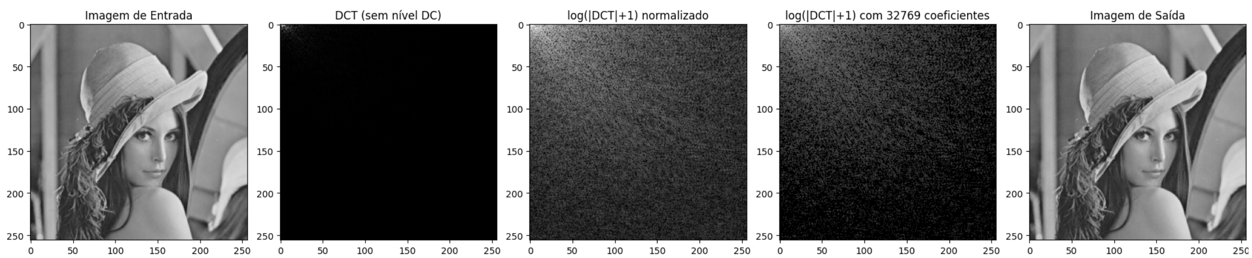


Resultados

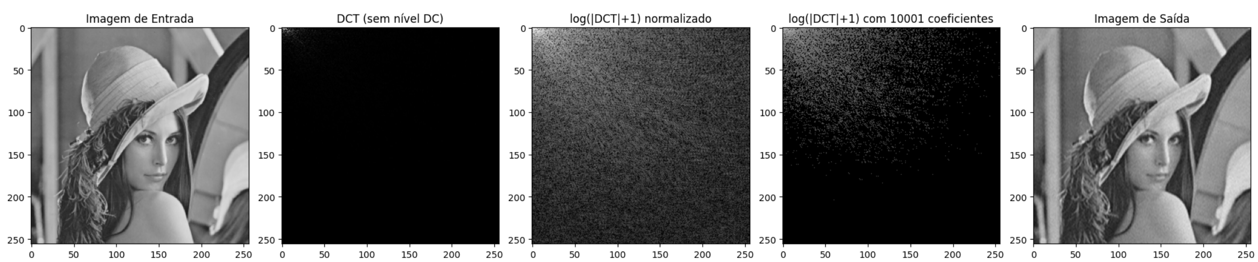
Foram aplicadas algumas alterações de visualização (log(|DCT| + 1) normalizado) para visualizar melhor a retirada de alguns coeficientes AC nas imagens.

Considerando a imagem da Lena com RxC = 65536, temos que, com apenas a metade dos coeficientes mais importantes da imagem, ainda obtemos uma imagem de saída "idêntica aos olhos humanos” da imagem de entrada. Com isso, podemos perceber a grande utilidade da DCT na compressão de imagens.

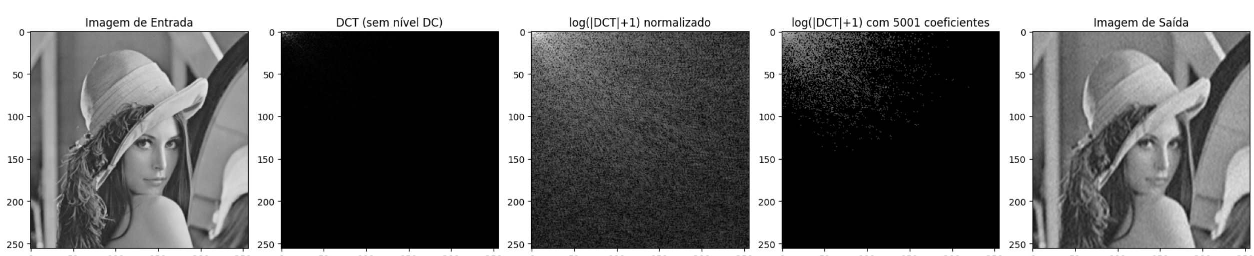
➢ **Com metade dos coeficientes mais importantes + coeficiente de DC**



➢ **Com os 10.000 coeficientes mais importantes + coeficiente de DC**

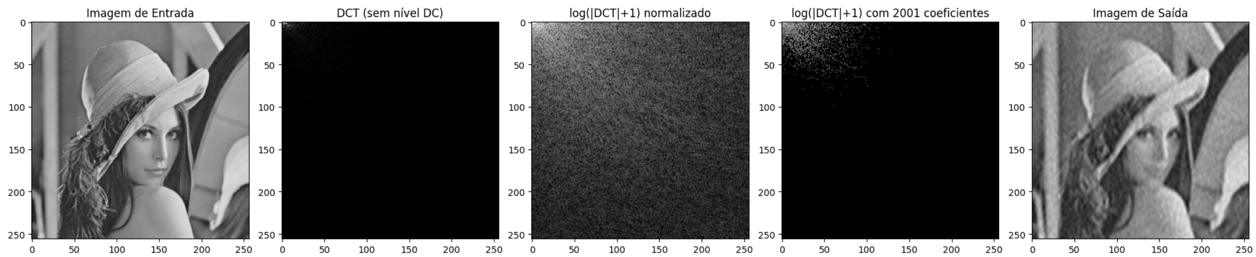


➢ **Com os 5.000 coeficientes mais importantes + coeficiente de DC**

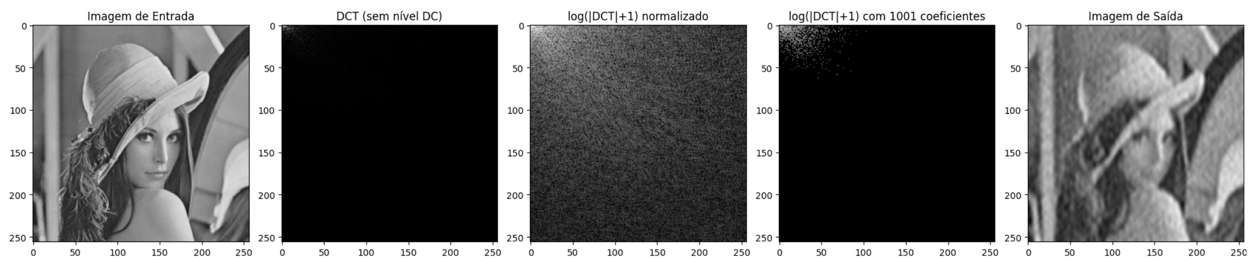


Conforme reduzimos o número de coeficientes utilizados (n), podemos percebemos a redução da qualidade da imagem (caso muitos coeficientes importantes sejam excluídos). Além disso, é possível observar o fenômeno de Gibbs, isto é, uma oscilação indesejada no resultado.

➢ **Com os 2.000 coeficientes mais importantes + coeficiente de DC**



➢ **Com os 1.000 coeficientes mais importantes + coeficiente de DC**



Como é possível observar nas imagens acima, podemos obter um nível de compressão muito bom dependendo do quanto queremos em qualidade, assim, mesmo usando menos da metade dos coeficientes mais importantes ainda conseguimos obter um bom resultado em qualidade e compressão da imagem.

2. Filtro de Butterworth passa-baixas (Imagem)

Fundamentação Teórica

O filtro de Butterworth é um tipo de filtro de processamento de sinal maximamente plano não-ideal. Um de seus modelos é o filtro passa-baixas. O filtro passa-baixas ou corta-altas é um filtro que passa sinais com uma frequência inferior à frequência de corte selecionada e corta frequências superiores à frequência de corte selecionada.

Em relação ao utilizado neste projeto para imagens temos, o H (que é a função de transferência do filtro), o d(k,l) (que é a distância euclidiana do coeficiente (k,l) até a origem, o fc (que é a distância de corte até a origem - input do usuário) e o n (que é a ordem do filtro - input do usuário).

Estratégia

Primeiramente, precisamos calcular a distância euclidiana de cada coeficiente até a origem, podemos fazer isso facilmente com a fórmula abaixo:

$$D(u,v) = \sqrt{(u^2 + v^2)}$$

Com isso, após a aplicação da transformada DCT e dado os valores de fc e n inseridos, podemos calcular a função de transferência do filtro através da fórmula abaixo:

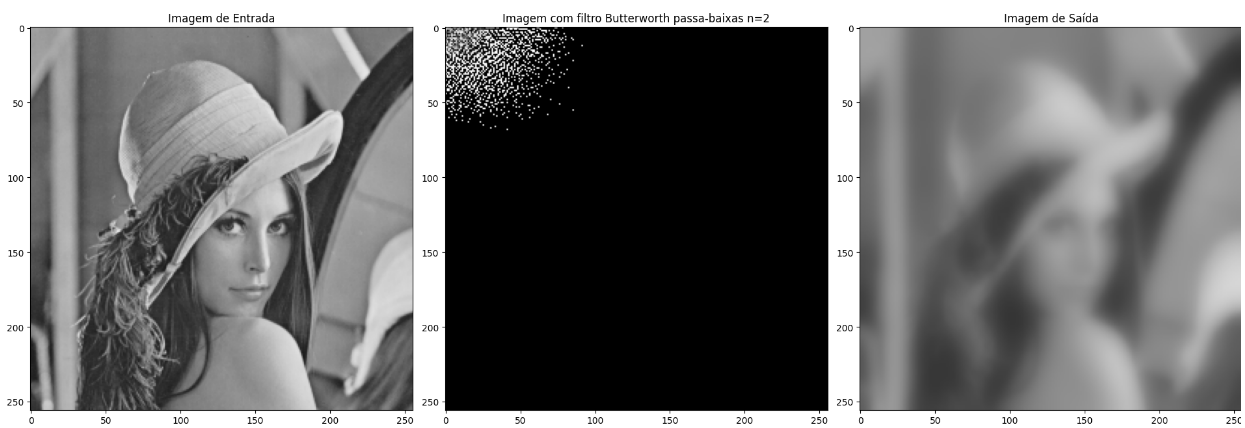
$$H(d(k,l)) = \frac{1}{\sqrt{1+\left(\frac{d(k,l)}{fc}\right)^{2n}}}$$

E então obter o resultado do filtro de butterworth passa-baixas aplicando a função de transferência H a cada coeficiente X[k][l] da imagem.

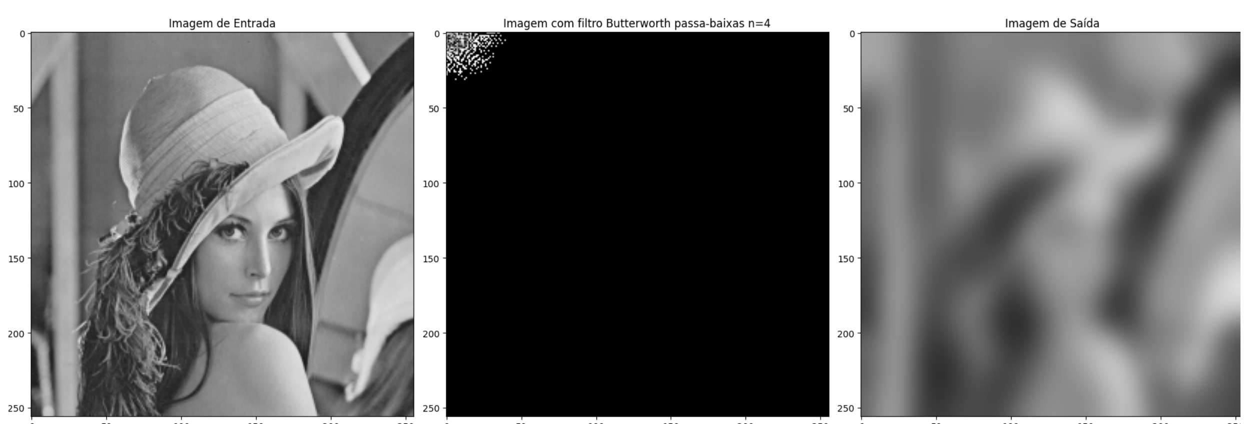
Resultados

Como é possível observar nas imagens abaixo, o filtro de Butterworth passa-baixas suaviza a imagem, isso acontece pois como o nome já diz, são preservados os cossenos de baixa frequência, dessa forma, eliminando os cossenos de altas frequências, perdemos as transições rápidas da imagem e ela se torna suavizada.

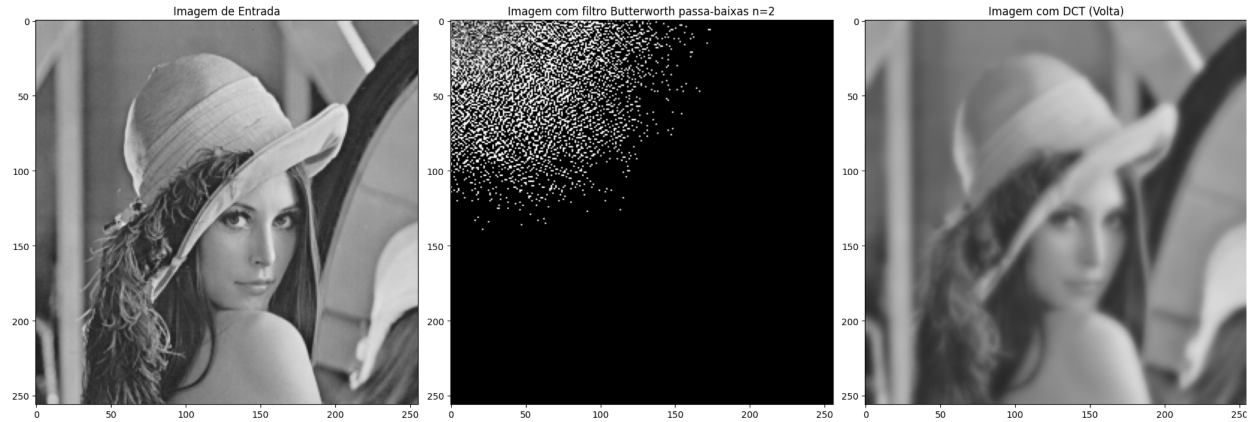
➢ **Com fc = 10 e n = 2**



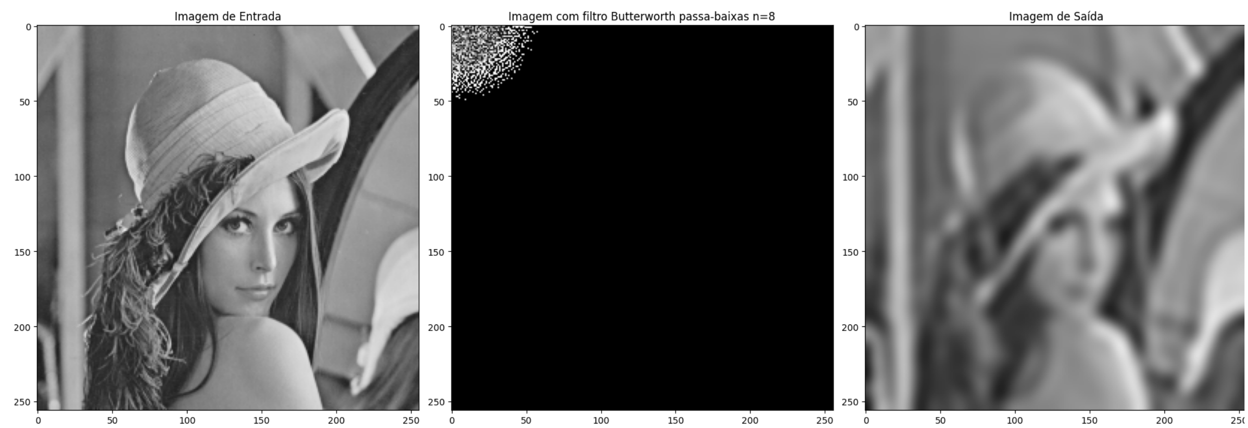
➢ **Com fc = 10 e n = 4**



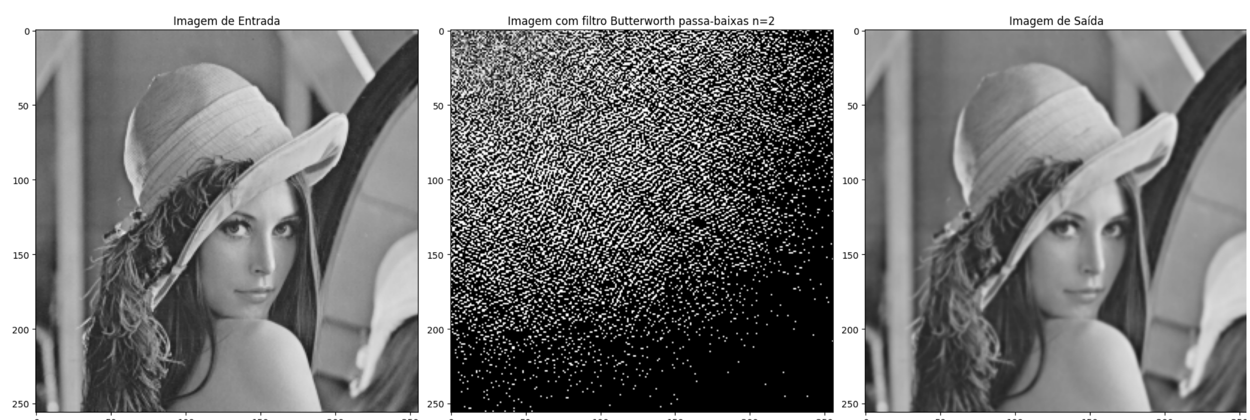
➤ **Com fc = 30 e n = 2**



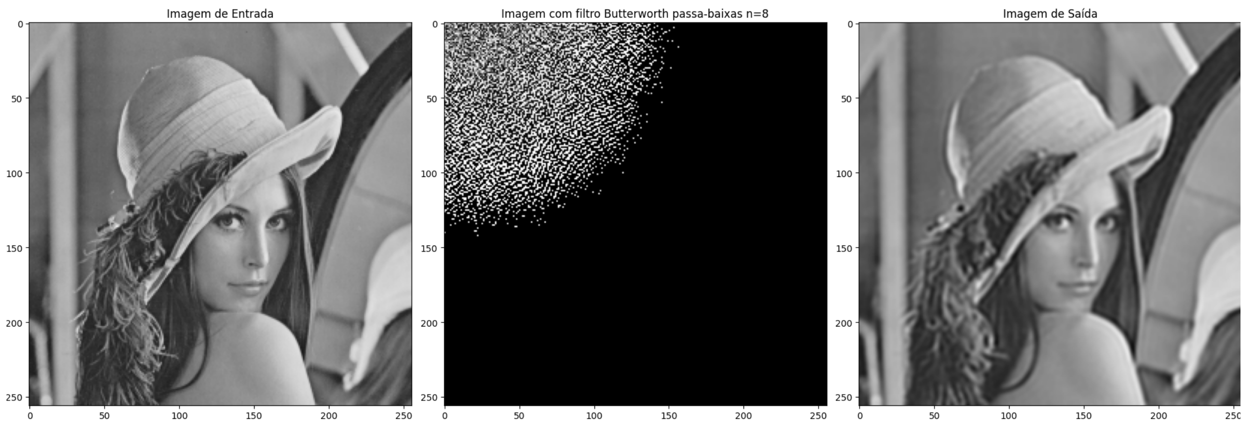
➤ **Com fc = 30 e n = 8**



➤ **Com fc = 100 e n = 2**



➢ **Com fc = 100 e n = 8**



Como é possível observar nas imagens acima, quanto menor o valor do fc, mais a imagem vai suavizar.

3. Filtro de Butterworth passa-baixas (Áudio)

Fundamentação Teórica

Podemos considerar a mesma fundamentação teórica do filtro butterworth passa-baixas, só que agora em relação aos áudios. Neste projeto para áudios temos, o H (que é a função de transferência do filtro), o f (que é a frequência em Hz), o fc (que é a frequência de corte em Hz) e o n (que é a ordem do filtro).

Estratégia

Precisamos calcular o array de frequências em Hz, para isso, primeiramente temos que calcular a frequência fundamental (f1) utilizando a fórmula abaixo:

$$f_1 = \frac{f_a}{2(N-1)} Hz$$

Com a frequência fundamental (f1), agora podemos calcular os valores de (f) para cada coeficiente, com a fórmula abaixo:

$$f_k = k \cdot f1Hz$$

Com isso, após a aplicação da transformada DCT e dado os valores de fc e n inseridos, além das frequências calculadas, podemos calcular a função de transferência do filtro.

$$H(f) = \frac{1}{\sqrt{1+\left(\frac{f}{fc}\right)^{2n}}}$$

Por fim, obtemos o resultado do filtro de butterworth passa-baixas aplicando a função de transferência H a cada coeficiente X[k] do áudio.

Resultados

O resultado percebido na aplicação do filtro butterworth passa-baixas foi um efeito de um som mais "abafado" à medida que se aumenta o valor da ordem do filtro (n) e utilizando uma frequência de corte (fc) mais baixa (fc = 500), isso acontece pois as frequências mais altas são perdidas.

Observação: os áudios de resultados encontram-se na pasta src\assets\audios

CONCLUSÃO

Com o desenvolvimento deste trabalho, foi possível entender melhor como a Transformada cosseno discreta (DCT) é utilizada na prática para processamento digital de imagens e compressão de dados. Além disso, que através do filtro Butterworth Passa-baixas (Low Pass) também é possível realizar a suavização da imagem no domínio da frequência.

Em relação aos resultados obtidos, as principais dificuldades foram encontrar valores de frequência de corte (fc) e ordem do filtro (n) para realizar os testes e perceber as diferenças.

REFERÊNCIAS BIBLIOGRÁFICAS

[1] Discrete cosine transform. Disponível em: https://en.wikipedia.org/wiki/Discrete_cosine_transform. Acesso em: 06 de junho de 2022.

[2] DC & AC Coefficients. Disponível em: https://sh-tsang.medium.com/review-cnnac-cnn-based-arithmetic-coding-for-dc-coefficients-hevc-intra-coding-d8c92462d769. Acesso em: 06 de junho de 2022.

[3] Computation of 2-D DCT using separability property. Disponível em: https://www.researchgate.net/figure/Computation-of-2-D-DCT-using-separability-property_fig2_224585702. Acesso em: 07 de junho de 2022.

[4] Butterworth filter. Disponível em: https://en.wikipedia.org/wiki/Butterworth_filter. Acesso em: 07 de junho de 2022.