

Relatório da Prática 4: Convolução Discreta

Matrícula:

Aluno: Antônio Teixeira Santana Neto

82888

Resumo Este relatório apresenta as informações obtidas e discutidas com relação à quarta prática do Laboratório de Sinais e Sistemas: Convolução Discreta. Aqui, exercitaremos o conceito e aplicação da convolução discreta através da construção de funções que simulem essa operação, assim como seu uso como filtros no processamento digital de imagens.

1 Introdução

No seguinte relatório serão trabalhados conceitos que servirão como base para teórica para a realização da prática. O objetivo deste é fazer um estudo mais aprofundado sobre a convolução e sua aplicação em uma imagem. Isto é, aqui estaremos dando os primeiros passos no que se refere ao processamento digital de imagens, no qual, por meio de uma série de manipulações, podemos extrair melhores informações contidas nas imagens de modo a facilitar a identificação e sua interpretação, ou até mesmo, enfatizar aspectos estéticos. Essa área possui diversas aplicações em vista do crescente desenvolvimento da tecnologia e aumento da sua acessibilidade, sendo assim, é de extrema importância concentrarmos esforços para alcançar resultados cada vez mais agradáveis. Para atingir esse objetivo, introduziremos um novo conceito, a convolução 2D, na qual nossa entrada deixará de ser um vetor, mas sim uma matriz, dado que nosso objeto de estudo consistirá em uma imagem previamente selecionada. Para tanto, utilizaremos mais uma vez o software MATLAB, a qual consiste numa poderosa ferramenta computacional, para realizarmos o nosso estudo a cerca da convolução 2D e sua aplicação no processamento digital de imagens.

2 Objetivos

Exercitar, no MATLAB, os conceitos de convolução discreta a partir de uma equação de diferença e, posteriormente, introduzir os passos iniciais para o processamento digital de imagens a partir da construção de um *script* que realize a operação de convolução sobre dois vetores dados, extrapolando esse ponto para o desenvolvimento de um outro *script* que realize a convolução 2D, aplicando-o sobre uma imagem previamente escolhida.

3 Metodologia

Como base para a execução dos experimentos será tomado como o roteiro da aula prática, dessa forma os dados foram obtidos a partir da resolução dos exercícios abordados no mesmo. Para auxiliar no desenvolvimento faremos o uso do software MATLAB, o qual é uma poderosa ferramenta de computação. Inicialmente, obtemos a resposta ao impulso de um sistema dado pela seguinte equação de diferença:

$$y[n] - 0,6y[n - 1] - 0,16y[n - 2] = 5x[n]$$

Essa resposta pode ser obtida com o uso da função *filter* que possui como argumentos os vetores *b* e *a*, correspondentes aos coeficientes do numerador e denominador da função de transferência, respectivamente, e o vetor de entrada, o impulso unitário. O mesmo procedimento também foi utilizado para obter a resposta do sistema ao degrau unitário, $u[n]$ e para $\frac{\sin[n]}{n}$. Os resultados obtidos serão mostrados posteriormente.

Em um segundo momento desta prática, para fixar nossos conceitos de convolução, criamos um *script* com o objetivo de realizar a operação de convolução entre dois vetores de entrada. Aqui, o código faz com que um dos vetores de entrada seja invertido utilizando o comando *fliplr* para depois, então, "deslizar" sobre o outro vetor de entrada. À medida que isso ocorre, os valores correspondentes de cada vetor são multiplicados e somados, retratando dessa forma a convolução. Os resultados serão comparados com aqueles obtidos pelo uso da função *conv*, presente no MATLAB, a qual também realiza essa operação.

Finalmente, no terceiro exercício, implementamos a convolução discreta bidimensional, a qual recebemos uma imagem e aplicamos uma série de processos sobre ela. Aqui o objeto de estudo, tratado como a entrada do sistema, é considerado como uma matriz de pixels com *I* linhas e *J* colunas, cada elemento da matriz pode representar diversas variáveis como: profundidade e cor. Utilizamos a filtragem como processo para realçar, corrigir (remoção de aspectos indesejáveis) ou suavizar as características da imagem. Esse procedimento é realizado pixel a pixel, em que o novo nível de cinza de um ponto *P* qualquer, por exemplo, depende do seu nível de cinza original e do de outros pontos considerados como vizinhos de *P*. Assim, aplicamos um filtro o qual funciona como uma máscara que desliza por toda a imagem, cada elemento do filtro age como um peso ou coeficiente. Dessa forma, cada valor correspondente ao pixel da imagem é substituído através de um processo no qual cada peso da máscara é multiplicado pelos elementos vizinhos da figura e por fim somados resultando no novo valor do pixel. Vale ressaltar que, os filtros utilizados, por questões de simetria, são do tipo $N \times N$, onde *N* é ímpar. Para realizar essa operação criamos um *script* que recebe o nome de *convolucao2D*, o qual recebe a matriz que corresponde à imagem e àquela referente ao filtro. Os comandos iniciais são de inversão da imagem a ser tratada e adição de duas linhas e duas colunas nulas às extremidades desta matriz, de modo que, ao se aplicar o filtro nos pixels da primeira coluna e linha da imagem todos os coeficientes da máscara sejam multiplicados e somados, evitando possíveis erros de operação. A partir daqui inicia-se o processo de convolução que corresponde a deslizar o filtro através da imagem como dito anteriormente. Os resultados obtidos serão abordados posteriormente.

4 Resultados

No primeiro exercício, trabalhamos com uma equação de diferença a fim de obter a resposta ao impulso através da aplicação de um filtro (função *filter* do MATLAB). Em seguida, o mesmo procedimento foi abordado para encontrar a resposta do sistema dado às entradas: $u[n]$ e $\frac{\sin[n]}{n}$. O resultado pode ser observado na *Figura 1*.

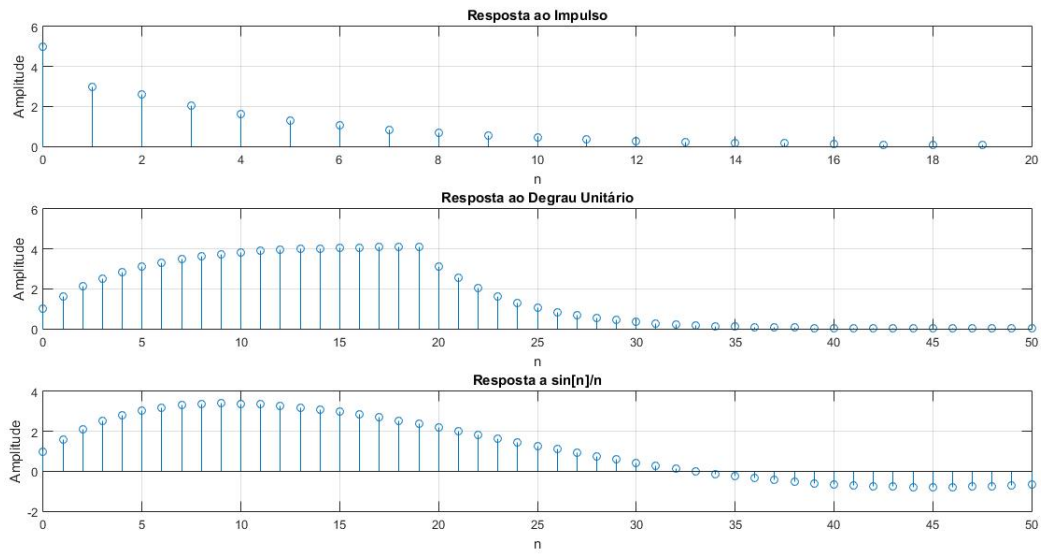


Figura 1: Resposta às entradas do sistema.

No segundo exercício, o conceito da convolução foi trabalhado profundamente, de modo que o objetivo dessa seção foi a elaboração de um *script* que realizasse a convolução entre dois vetores de entrada, os quais foram:

$$a = [\text{ones}(1,30)] \quad e \quad b = [1 \quad 1 \quad 1 \quad 1 \quad 1 \quad \text{zeros}(1,15)]$$

O resultado da função criada bem como a utilização da função *conv* presente no MATLAB são mostrados na *Figura 3* para fins comparativos.



Figura 2: Resposta das convoluções.

Percebe-se que a função criada consegue realizar a operação de convolução com uma boa aproximação daquela presente no MATLAB, visto que os gráficos são bastante parecidos. Ademais, a função criada demandou de um tempo correspondente a 9ms, aproximadamente, contra 50ms da função *conv*, o que demonstra que nosso *script* não demanda de muita capacidade de processamento, cerca de 5 vezes mais rápida.

Finalmente, abordaremos agora os resultados obtidos acerca da convolução 2D realizada. Para isso, começamos pela construção de um *script* o qual seria responsável por fazer a convolução da nossa imagem. Isto é, após transformar a imagem, obtida através do *download* do arquivo "lua" do PVA-net, numa matriz na qual cada elemento corresponde a um pixel da figura, um filtro selecionado irá percorrer toda a imagem realizando a sua convolução. A seguir, podemos observar as mudanças na imagem após a utilização do filtro F1:

$$F1 = \begin{bmatrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{bmatrix}$$



Figura 3: Imagem da Lua sem filtro (esquerda) e com filtro (direita).

Podemos notar na *Figura 3* que a imagem foi realçada após o uso do filtro. Isso se deve ao fato de ter sido utilizado um Filtro de Laplace, ou um filtro passa-altas, o qual atenua (ou elimina) baixas frequências, enfatizando altas frequências e tornando mais nítida a transição entre diferentes regiões. A seguir aplicamos os filtros F2 e F1 sobre uma outra imagem a fim de discutir os resultados obtidos.

$$F2 = \frac{1}{2} * \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix} \quad e \quad F3 = \begin{bmatrix} -3 & 2 & 1 \\ -0,5 & 10 & -0,5 \\ -0,5 & -1,5 & -0,5 \end{bmatrix}$$



Figura 4: Imagem original (esquerda); imagem após aplicação do filtro F3 (centro); imagem após aplicação do filtro F2 (direita).

Percebe-se que, com a aplicação do filtro passa-baixas, F2, na *Figura 4*, obtemos uma imagem mais "suavizada", porém mais clara que a original e atenuada, uma vez que uma característica deste filtro é atenuar frequências que estão relacionadas aos detalhes da imagem, diminuindo o "ruído". Ao aplicar o filtro F3 (imagem do centro), por sua vez, percebe-se a presença de imperfeições, como se a imagem tivesse mais "ruído". Este resultado pode ser dado ao

filtro utilizado, o qual não apresenta nenhum critério em si (não caracteriza um filtro passa-baixa ou passa-alta, por exemplo).

Finalmente, obtemos o seguinte resultado da aplicação de um filtro passa-baixa, F4, sobre a imagem de um texto:

$$F2 = \frac{1}{9} * \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

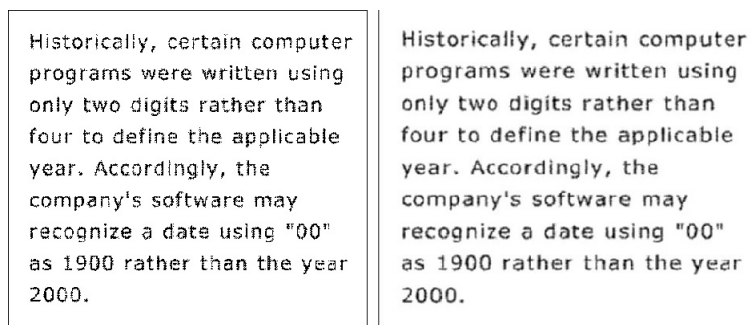


Figura 5: Imagem do texto sem o filtro (esquerda); Texto após aplicação do filtro (direita).

Notamos que a utilização do filtro tornou o texto mais agradável à leitura, uma vez que remove os "ruídos" presente na imagem. Além disso, podemos notar que o fundo branco ficou mais intenso e a imagem, em si, mais "suave".

5 Conclusão

A partir do que foi decorrido acima, ficou fixado o conceito da convolução e sua aplicação, assim como o uso de filtros para obter a resposta ao impulso de um sistema dado uma equação de diferença. Ademais, podemos observar a aplicação dessa operação no processamento digital de imagens, uma vez que uma figura qualquer pode ser tratada como uma matriz em que cada elemento corresponde a um pixel e, a partir daí, podemos submeter o objeto a uma série de tratamentos por meio de filtros especiais. Com isso, podemos, então, escolher quais características da imagem devem ser realçadas ou atenuadas, de modo que facilite a interpretação dos dados ou até mesmo a extração de informações que antes estavam difíceis de serem visualizadas.

6 Referências

OPPENHEIM, Alan V., WILLSKY, Alan S., com NAWAB, S. Hamid. Sinais e Sitemas. 2ª edição. São Paulo: Pearson, 2010.

Processamento de Imagens - Tratamento da Imagem Disponível em

;<http://equipe.nce.ufvj.br/thome/pgad/nnimg/transp/c4filtros.pdf> > acesso em : 10/09/2017