

# Projeto 2

Gustavo Antonio Souza de Barros  
18/0064487

Introdução ao processamento de imagens  
Engenharia da Computação  
Universidade de Brasília  
Brasília, Distrito Federal 70910-900  
Email: gustavoasb@gmail.com

**Abstract**—O projeto desenvolvido tem como objetivo a realização de três atividades. A primeira tem foco na segmentação de uma imagem, separando o texto do fundo, mediante filtros e operações morfológicas. A segunda aborda a aplicação de um filtro *rejeita-notch*, mais especificadamente o filtro *Butterworth*, a fim de tirar o padrão *moiré* da imagem. O terceiro discorre sobre a eliminação de uma *cookie* mordida em uma imagem, retirada a partir de algoritmos morfológicos.

## I. INTRODUCTION

O processamento de imagens é um tópico que tem alta aplicação nas tecnológicas mais atuais do mercado.

A manipulação de imagens se torna muito importante pela capacidade de fazer ajustes a fim de retirar informações dessas figuras. Auxiliando o ser humano a juntar dados ou aplicar operações encima desta.

A fim de entender melhor as tarefas desenvolvidas, é necessário descorrer sobre as chamadas transformações morfológicas.

O elemento estruturante é uma forma que tem como objetivo realizar procedimentos encima da imagem, a partir de um formato definido anteriormente e formatos geométricos da imagem, através de convoluções. Isso será esclarecido no descorrer do texto.

Quando se fala de código, o elemento estruturante é definido por matrizes binárias, como no exemplo abaixo:

$$\begin{bmatrix} 0 & 1 & 1 & 0 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 0 \end{bmatrix} \quad (1)$$

A forma acima lembra o que seria um círculo se ampliado, assim podemos formar várias figuras geométricas e utiliza-las para executar operações sobre as imagens.

As operações morfológicas anteriormente citadas são representadas por:

- Erosão
- Dilatação
- Abertura
- Fechamento
- Gradiente
- Top Hat
- Bottom Hat

A erosão promove o descarte de regiões que não se encaixam no elemento, diminuindo a espessura dessa região. A Dilatação faz o processo contrário, promovendo o aumento de espessura.

A abertura e fechamento compreendem a utilização dos dois já abordados em sequência. A abertura é uma erosão seguida por uma dilatação, podendo assim eliminar ruídos exteriores a região de interesse. O fechamento é uma dilatação seguida por uma erosão, eliminando buracos ou defeitos nas regiões dessa imagem.

O gradiente é a diferença de uma dilatação e uma erosão, dando o contorno da região de interesse.

O *top hat* é a diferença da imagem pela abertura da mesma, é usado para extrair pequenos detalhes de uma imagem, assim como para o realce de regiões claras em fundo escuros em imagens em tons de cinza.

O *bottom hat* é a diferença da imagem pelo seu fechamento, diminuindo o detalhamento e dando ênfase as regiões maiores.

Para a segunda atividade, conhecimento sobre a transformada de *fourier* é essencial.

A Transformada de *Fourier* é uma transformada que demonstra o sinal dado em funções de seno e cosseno. Quando utilizada em imagens ela se torna bidimensional, e se torna discreta, pois o computador não interpreta sinais contínuos, mas sim uma simplificação destes, virando uma transformada discreta de *fourier*.

Com a utilização da transformada, podemos utilizar filtros encima e obter resultados mais facilmente, pois ela separa as frequências da imagem em regiões, e assim pode-se atuar sobre estas.

## II. METODOLOGIA

### A. Questão 1

Para a primeira questão, realizou-se procedimentos encima da imagem dada, realizando operações para isolar as letras, assim como seu fundo.

Primeiramente, lê-se a imagem e aplica-se um filtro bilateral com auxílio do *OpenCV*.

Em seguida, faz-se um *bottom hat* na imagem, isolando assim as letras do fundo. Definido um limiar, binariza-se a imagem.

Para a segunda parte, aplica-se fechamento sobre a imagem, de forma a sumir com as letras, revelando o fundo.

Depois subtraiu-se o fundo da imagem original, deixando novamente só as letras. A fim de diminuir o ruído, aplicou-se uma abertura para tirar os pixels ruidosos.

### B. Questão 2

Para a segunda atividade, foi criado que some com o padrão moiré de uma imagem, utilizando o filtro passa-altas de Butterworth.

O primeiro procedimento adotado é aplicar a transformada de fourier na imagem, feito com auxílio do Numpy, também foi realizado o shift para centralizar a transformada. Depois foi criado o espectro de magnitude da transformada, e identificou-se quais as frequências dele que estavam ligados ao moiré.

Com a localização do padrão identificada, aplicou-se o filtro de Butterworth. Para aplicar esse filtro, é necessário calcular a distância, através da seguinte fórmula:

$$D_k(u, v) = [(u - P/2 - u_k)^2 + (v - Q/2 - v_k)^2]^{1/2} \quad (2)$$

Sendo  $P/2$  e  $Q/2$  os centros da imagem, e  $(u_k, v_k)$  os centros do par de filtro notch.

Após isso, usa-se a fórmula abaixo para calcular os valores da matrix do filtro:

$$Filter(u, v) = \frac{1}{1 + [D_0/D_k(u, v)]^{2n}} \quad (3)$$

Sendo  $D_0$  o raio do círculo de onde o filtro será aplicado e  $n$  a ordem do filtro, que no projeto foi fixado como 4.

É importante ressaltar que esse processo é feito duas vezes, pois o filtro tem que ser bidimensional, e depois multiplicou-se os valores encontrados pelas duas matrizes de filtragem.

Esse processo deixa falhas nas bordas, problema resolvido pelo padding, que foi aplicado ao redor da matriz de fourier e da de filtros.

Depois de todas essas etapas, multiplica-se a matriz de fourier pela do filtro, resultando na imagem filtrada. Agora é necessário fazer a transformada inversa de fourier para voltar á imagem original. Primeiro se desfaz o shift feito, e em seguida o fourier, descartando a parte imaginária da matriz.

### C. Questão 3

Na terceira questão, era necessário manipular a imagem, de forma a eliminar a cookie mordida em uma imagem em que há duas delas.

Inicialmente, binariza-se a imagem, de forma a separar as cookies de maneira mais cuidadosa possível do fundo.

Em sequência, faz-se a erosão na imagem com um elemento estruturante de raio aproximado ao do cookie, de forma a sumir com o cookie mordido e sobrar parte do inteiro.

Depois disso, dilata-se novamente o resultado da operação anterior, de forma a deixar do tamanho da original.

Percebe-se fazendo isso que há muita perda de detalhe, por uso de um elemento estruturante redondo, enquanto a esfera não é perfeitamente redonda. Para resolver isso, dilata-se mais um pouco a informação do cookie completo e multiplica-se pelo binário inicial, de forma a achar os pontos de interseção das duas, recuperando assim o cookie completo sem perda de detalhes.

Com o cookie completo separado, o mesmo foi apresentado de duas maneiras, a primeira é ele com o fundo preto, e a segunda é um fundo gerado aleatoriamente e baseado no fundo original, de forma a dar a impressão que o cookie mordido realmente foi tirado da imagem.

## III. RESULTADOS

Essa seção abordará sobre os resultados encontrados nos tópicos debatidos anteriormente.

### A. Questão 1

A imagem original é mostrada abaixo para efeito comparativo.

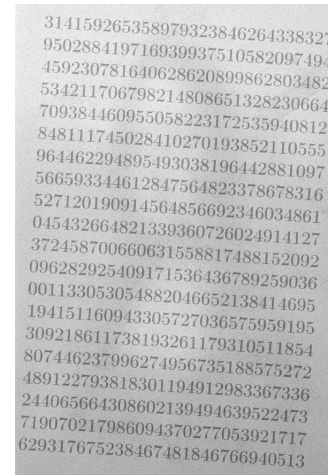


Fig. 1. Imagem original

Depois de aplicar o bottom hat e o filtro bilateral:



Fig. 2. Bottom hat aplicado

Ao definir-se um limiar bom, converte-se a imagem para binário.

Também aplica-se o fechamento para melhorar os contornos das letras, que muitas das vezes perdem detalhes no processos anteriores.

```

314159265358979323846264338327
950288419716939937510582097494
459230781640628620899862803482
534211706798214808651328230664
709384460955058223172535940812
848111745028410270193852110555
964462294895493038196442881097
566593344612847564823378678316
527120190914564856692346034861
045432664821339360726024914127
372458700660631558817488152092
096282925409171536436789259036
001133053054882046652138414695
194151160943305727036575959195
309218611738193261179310511854
807446237996274956735188575272
489122793818301194912983367336
244065664308602139494639522473
719070217986094370277053921717
629317675238467481846766940513

```

Fig. 3. Binarização

Outro processo pedido foi a separação do fundo, fazendo fechamento chegamos ao seguinte resultado:



Fig. 4. Fundo separado

## B. Questão 2

A saída desse exercício é a imagem original mostrada na Imagem 5 sem o padrão encontrado nela, processo feito por filtros de Butterworth.

Pra esse processo também é encontrada a transformada de Fourier da imagem, seu espectro de magnitude também é mostrada na Figura 6.

Como pode-se ver na Figura 7, o filtro correspondente ao pedido foi encontrado e aplicado, resultando na Figura 8.



Fig. 5. Image original com padrão moiré

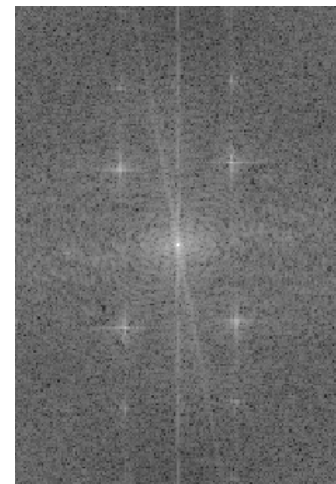


Fig. 6. Espectro de magnitude

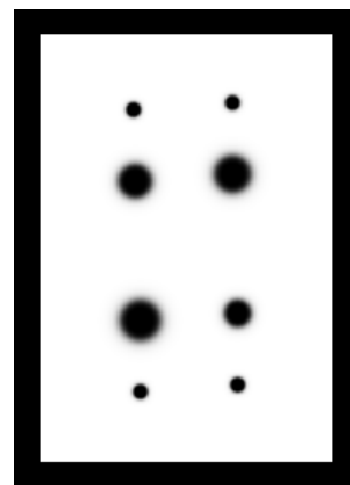


Fig. 7. Filtro utilizado

Abaixo, pode-se ver o resultado achado:



Fig. 8. Imagem Moiré Filtrada

### C. Questão 3



Fig. 9. Imagem Original dos Cookies

A terceira tarefa era baseada em eliminar a cookie mordida, trabalhando com transformações morfológicas.

A primeira etapa desse processo consiste na erosão da imagem deixando apenas parte da cookie completa, isso tudo depois de binarização, como pode-se ver na Figura 9

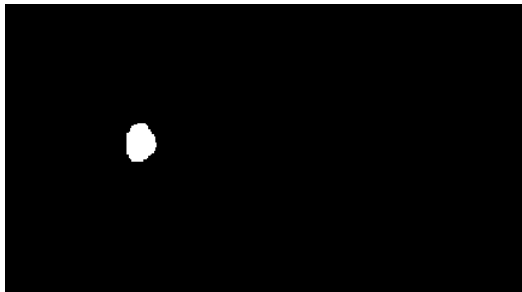


Fig. 10. Cookies erodidos

Depois disso dilata-se e se pega os pontos de interseção com o principal, chegando ao resultado da Figura 10

Em seguida, é hora de mostrar apenas a cookie completa, chegando a saída final presente na Figura 11

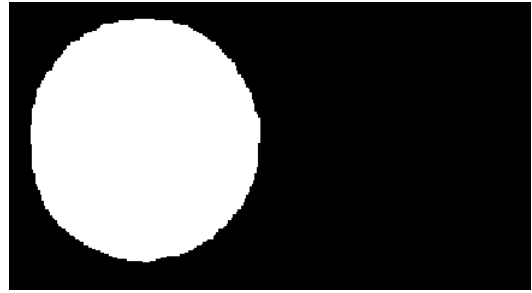


Fig. 11. Cookie completa recuperada

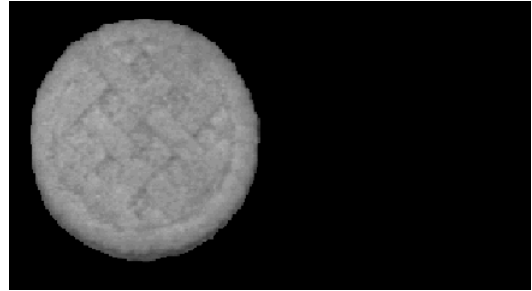


Fig. 12. Apenas cookie completa

## IV. CONCLUSÕES

Levando em conta toda a realização do projeto, considera-se que este foi executado com sucesso e com resultados aceitáveis. Apesar de ter sido concluído de forma decente, há espaço para melhorias.

A aplicação do filtro Butterworth foi feito através de loops, se funções do Numpy ou de outra biblioteca fossem utilizadas a velocidade de execução aumentaria consideravelmente.

Os resultados deram a perceber na prática como funcionam diversos dos conceitos desenvolvidos em sala de aula, além de uma maior percepção do assunto.

## REFERENCES

- [1] OpenCV Team, *Open Source Computer Vision Documentation*. [opencv.org](http://opencv.org).
- [2] OpenCV Team, *Morphological Transformations*. Accessed in 17, May, 2019.
- [3] SciPy Team, *NumPy User Guide*. [docs.scipy.org](https://docs.scipy.org).
- [4] Wikipedia, *Butterworth filter*. Accessed in 17, May, 2019.
- [5] Luiz Eduardo S. Oliveira, *Morfologia Matemática Binária*, Processamento de imagens, [www.inf.ufpr.br/lesoliveira/download/morfologia.pdf](http://www.inf.ufpr.br/lesoliveira/download/morfologia.pdf).