

Reconhecimento de caracteres e palavras em imagens

Aline Araujo e Jéssica C. Ruel

Resumo—Este artigo está abordado as principais técnicas utilizadas no processamento de imagens para reconhecimento textual, assim como os resultados e as considerações finais sobre o tema.

I. INTRODUÇÃO

Processamento de imagem é um conjunto de técnicas computacionais aplicadas à uma imagem a fim de tratá-la ou extrair alguma informação.

Dado essas técnicas em conjunto com outras ferramentas, nos permite tratar e analisar caracteres e textos inseridos em imagens. A partir disso, é possível trabalhar com diversas imagens e quadros de vídeo de uma maneira fácil, ao se aplicar os métodos corretos para cada caso.

Considerando esse contexto, este trabalho apresenta as técnicas utilizadas para o processamento de imagens, com o intuito de extrair textos das mesmas. Este artigo está organizado da seguinte forma: na Seção II é apresentado as técnicas utilizadas, os resultados são apresentados na Seção III e as considerações finais na Seção IV.

II. TÉCNICAS UTILIZADAS

Nesta seção, detalharemos todas as técnicas utilizadas neste trabalho. Sendo elas: Thresholding, Suavização, Transformações Morfológicas, Detecção de bordas e Histogramas.

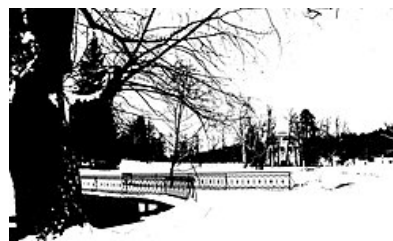
A. Thresholding

No processamento de imagens, os métodos de segmentação são usados para particionar uma imagem, de forma que seus pixels se dividem em grupos. O thresholding ou limiarização é a forma mais simples de

segmentação, sendo habitualmente o primeiro passo desse processo.



(a) imagem original.



(b) imagem binária resultante na limiarização.

Fig. 1. Exemplo de thresholding. [1]

Nesse trabalho, abordamos os seguintes tipos de thresholding:

1) Thresholding Simples

Na limiarização simples é aplicado o mesmo valor limite para todos os pixels. Sendo assim, quando o valor é menor que o limite, ele é definido como 0 e quando não é, ele é definido como um valor máximo. Esse é o thresholding binário, já o thresholding binário inverso funciona com os valores invertidos da sua forma normal.

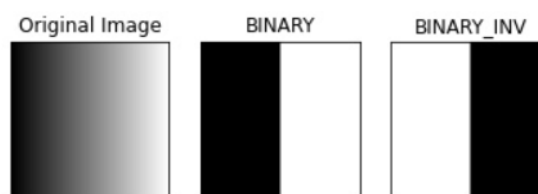


Fig. 2. Exemplo de thresholding binário e thresholding binário inverso. [2]

2) Thresholding Adaptativo

A limiarização simples pode não ser boa para todos os casos, já que uma imagem pode ter, por exemplo, diferentes iluminações em diferentes áreas. Sendo assim, o thresholding adaptativo pode ser útil. Nesse algoritmo é determinado o limite do pixel levando em consideração a região ao seu redor.

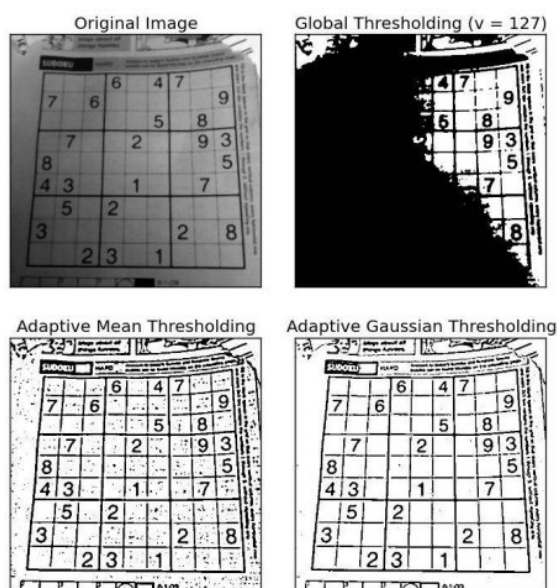


Fig. 3. Exemplo de thresholding adaptativo. [2]

B. Suavização

A suavização de imagem é um método comum de processamento de imagens digitais que remove o conteúdo de alta frequência, como os ruídos e bordas de uma imagem. Nesse trabalho, abordamos os seguintes tipos de suavização:

1) Suavização Média

Essa técnica utiliza um filtro de caixa normalizado para convoluir uma imagem, simplesmente, obtendo a média de todos os pixels na área do kernel especificado e substituindo o elemento central.

2) Suavização Gaussiana

Essa técnica é similar à Suavização Média, tendo como diferença a utilização de um

kernel gaussiano em vez de um filtro de caixa e determinado a suavização pelo desvio padrão gaussiano.

3) Suavização Mediana

Essa técnica é similar às outras já citadas, se diferenciando ao obter a mediana de todos os pixels da área do kernel e substituindo o valor central pelo valor mediano.

C. Transformações Morfológicas

Após à segmentação, pode ser necessário o uso de transformações morfológicas para remover ou minimizar as imperfeições geradas.

Foram abordadas as seguintes transformações morfológicas:

1) Erosão

A erosão funciona de uma maneira muito simples, o kernel passe pela imagem e o pixel será considerado 1 apenas se todos os pixels da área do kernel forem 1, caso contrário, ele se torna 0, erodindo.

2) Dilatação

Essa é o oposto da erosão, com a dilatação o elemento será 1 se existir algum pixel 1 na área do kernel.

3) Abertura

A abertura é apenas o processo de erosão seguida pela dilatação. É bastante utilizada para remoção de ruídos na imagem.

4) Fechamento

Já o fechamento, é o oposto da abertura. Sendo o processo de dilatação seguindo pelo de erosão. É útil para fechar orifícios ou pequenos pontos.



(a) erosão (b) dilatação

Fig. 4. Exemplo de erosão e dilatação. [4]

D. Detecção de bordas

Dependendo do processamento que está sendo feito, pode ser interessante métodos para detecção de contornos. Trabalhamos com as seguintes técnicas para extrair contornos de imagem:

1) Filtro Sobel

Esse filtro calcula diferenças finitas em cada ponto da imagem, captando a direção da maior variação de claro para escuro. Obtendo assim, noção da variação de luminosidade de cada ponto.

2) Filtro Laplaciano

Já o filtro laplaciano implementa uma derivada de segunda ordem da imagem, detectando áreas de alta variação de cor.

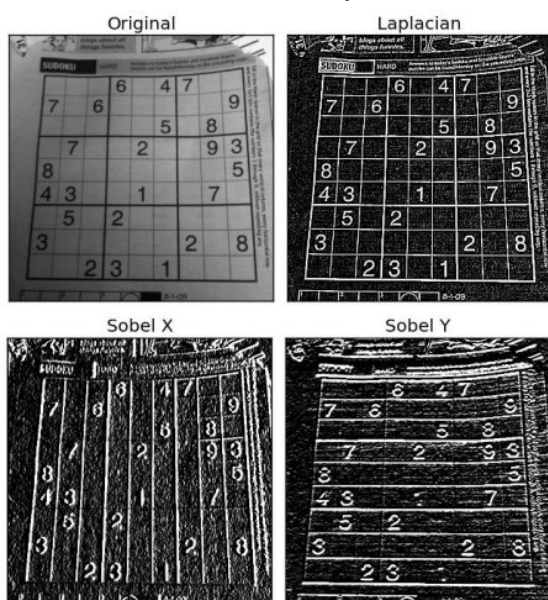


Fig. 5. Exemplo de filtro sobel e laplaciano. [7]

3) Canny

Esse é um algoritmo muito utilizado na detecção de bordas. Sendo seus processos a redução de ruídos, obter o gradiente de intensidade da imagem utilizando um kernel sobel, remoção de pixels indesejados e não bordas, por fim, são definidas as bordas dado a intensidade dos pixels.

E. Histogramas

O histograma é um gráfico que lhe dá informações gerais sobre a distribuição de intensidade de uma imagem. Podemos obter

informações sobre uma imagem como brilho, distribuição de intensidade, contraste e etc, apenas visualizando o histograma da mesma.

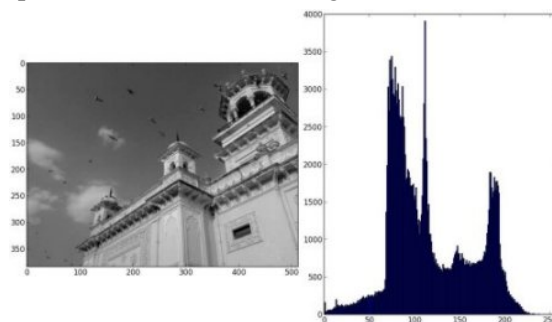


Fig. 6. Exemplo de histograma. [8]

III. RESULTADOS

Nesta sessão, será apresentado os detalhes e resultados dos experimentos realizados no presente trabalho.

Experimento 1:

No experimento 1, foi realizado o reconhecimento de texto em uma imagem com escala de cinza e ruído.

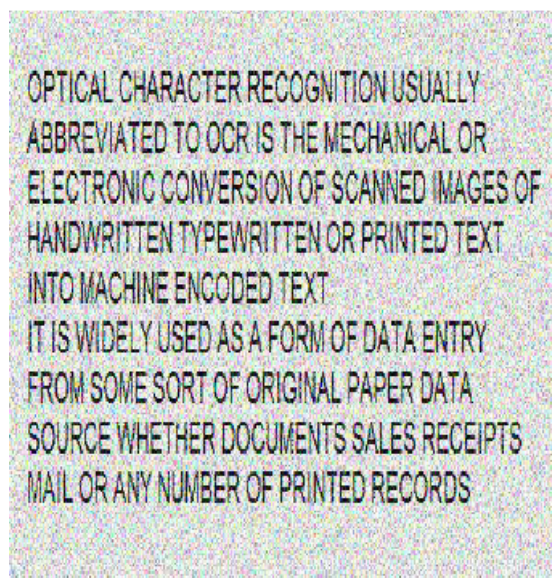


Fig. 7. Imagem com escala de cinza e ruído.

Foram utilizadas as técnica de thresholding binário, blur mediano e fechamento para ser possível aplicar a função de reconhecimento de texto na imagem. Após a aplicação destas técnicas, foi obtida a seguinte imagem e texto:

OPTICAL CHARACTER RECOGNITION USUALLY
 ABBREVIATED TO OCR IS THE MECHANICAL OR
 ELECTRONIC CONVERSION OF SCANNED IMAGES OF
 HANDWRITTEN TYPEWRITTEN OR PRINTED TEXT
 INTO MACHINE ENCODED TEXT
 IT IS WIDELY USED AS A FORM OF DATA ENTRY
 FROM SOME SORT OF ORIGINAL PAPER DATA
 SOURCE WHETHER DOCUMENTS SALES RECEIPTS
 MAIL OR ANY NUMBER OF PRINTED RECORDS

Fig. 8. Imagem com escala de cinza e ruído
 pós tratamento.

“OPTICAL CHARACTER RECOGNITION
 USUALLY
 ABBREVIATED TO OCR IS THE
 MECHANICAL OR
 ELECTRONIC CONVERSION OF
 SCANNED IMAGES OF
 HANDWRITTEN TYPEWRITTEN OR
 PRINTED TEXT
 INTO MACHINE ENCODED TEXT

IT IS WIDELY USED AS A FORM OF
 DATA ENTRY
 FROM SOME SORT OF ORIGINAL PAPER
 DATA

SOURCE WHETHER DOCUMENTS SALES
 RECEIPTS
 MAIL OR ANY NUMBER OF PRINTED
 RECORDS”

Experimento 2:

No experimento 2, foi utilizada uma
 imagem colorida com boa resolução.



Fig. 9. Imagem colorida.

Foram utilizadas as técnica de thresholding
 binário invertido, blur mediano e fechamento.
 Após a aplicação das técnicas, foi possível
 aplicar a função para reconhecer o texto e
 capturar os seguintes resultados:

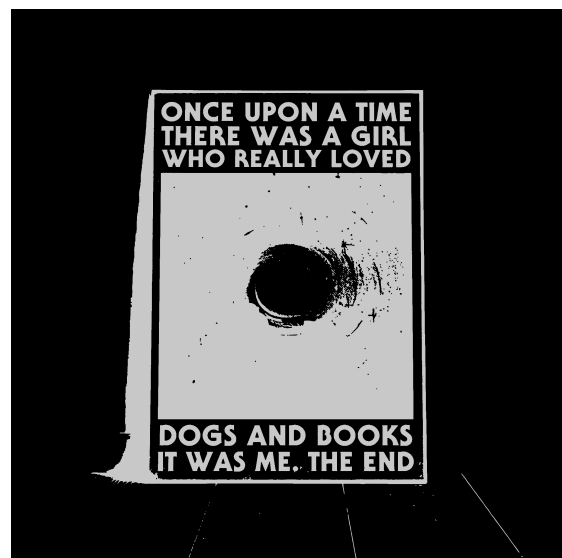


Fig. 10. Imagem colorida pós tratamento.

“ONCE UPON A TIME

THERE WAS A GIRL
 WHO REALLY LOVED

DOGS AND BOOKS
IT WAS ME. THE END.”

Experimento 3:

No experimento 3, foi utilizada uma imagem de raio-x em escala de cinza de com ruído. A imagem possui informações do paciente, como nome, idade, horário da realização do exame, etc.

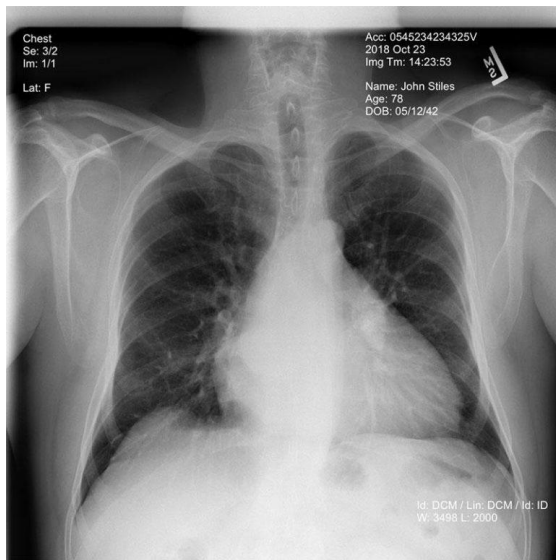


Fig. 11. Imagem Raio-x.

Neste caso, foi utilizada somente a técnica de thresholding binário. Após sua aplicação, foi obtido os seguintes resultados:



Fig. 12. Imagem Raio-x pós tratamento.

“Chest

Se: 3:2
Im: 1/1

Lat: F

Acc: 0545234234325V
2018 Oct 23
Img Tm: 14:23:53 ra

»

Name. John Stiles
Age: 78
DOB: 05/12/42

BCM Id: 1D”

Experimento 4:

Neste último experimento, foi utiliza uma página de livro amarelada.

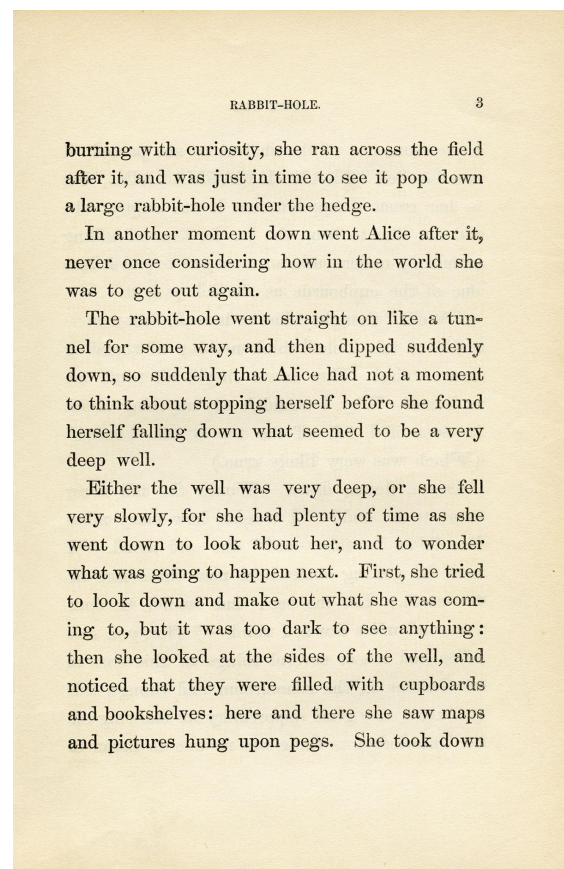


Fig. 13. Imagem da página de um livro amarelada.

Foram utilizadas as técnica de thresholding binário invertido e blur mediano. Os seguintes resultados foram obtidos:

“RABBIT-HOLE. 3

burning with curiosity, she ran across the field after it, and was just in time to see it pop down a large rabbit-hole under the hedge.

In another moment down went Alice after it, never once considering how in the world she was to get out again.

The rabbit-hole went straight on like a tunnel for some way, and then dipped suddenly down, so suddenly that Alice had not a moment to think about stopping herself before she found

herself falling down what seemed to be a very deep well.

Hither the well was very deep, or she fell very slowly, for she had plenty of time as she went down to look about her, and to wonder what was going to happen next. First, she tried to look down and make out what she was coming to, but it was too dark to see anything: then she looked at the sides of the well, and noticed that they were filled with cupboards and bookshelves: here and there she saw maps

and pictures hung upon pegs. She took down”

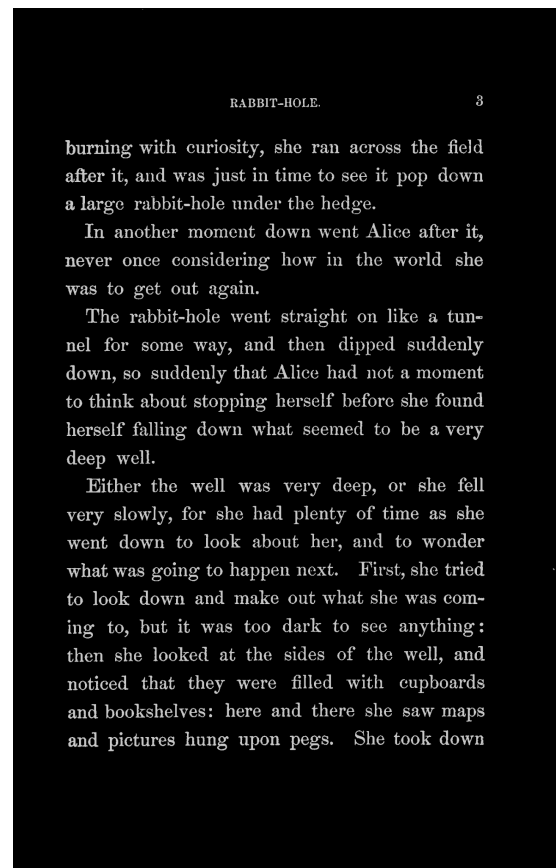


Fig. 14. Imagem da página de um livro amarelada pós tratamento.

IV. CONCLUSÃO

Apesar da utilização de algumas técnicas se repetir no decorrer dos experimentos, é possível notar que nem sempre o reconhecimento é perfeitamente concluído. A dificuldade em processar o raio-x, por exemplo, fez com que alguns ruídos saíssem como caracteres no texto, porém ainda assim foi obtido um resultado surpreendente.

REFERENCES

- [1] Wikipedia “Thresholding (image processing)” “Disponível em: <[https://en.wikipedia.org/wiki/Thresholding_\(image_processing\)](https://en.wikipedia.org/wiki/Thresholding_(image_processing))>
- [2] OpenCV Documentation “Image Thresholding”. Disponível em: <https://docs.opencv.org/4.x/d7/d4d/tutorial_py_thresholding.html>
- [3] OpenCV Documentation “Smoothing Images”. Disponível em: <https://docs.opencv.org/4.x/d4/d13/tutorial_py_filtering.html>
- [4] OpenCV Documentation “Morphological Transformations”. Disponível em: <https://docs.opencv.org/4.x/d9/d61/tutorial_py_morphological_ops.html>
- [5] Wikipedia “Filtro Sobel” “Disponível em: <https://pt.wikipedia.org/wiki/Filtro_Sobel>
- [6] OpenCV Documentation “Image Gradients”. Disponível em: <https://docs.opencv.org/4.x/d5/d0f/tutorial_py_gradients.html>
- [7] OpenCV Documentation “Histograms - 1 : Find, Plot, Analyze !!!”. Disponível em: <https://docs.opencv.org/4.x/d1/db7/tutorial_py_histogram_begins.html>