

# Trabalho 3 - MC920

Nome: Felipe Pessina RA: 171214

May 22, 2019

## 1 Introduction

O objetivo deste trabalho é realizar a identificação de linhas e palavras em uma imagem, com texto e figuras. Para isto, fizemos uso das operações de dilatação, erosão, fechamento, intersecção de imagens, detecção de regiões convexas e seus contornos. Para a seleção as regiões de interesse, calculamos algumas estatísticas da imagem, como densidade de pixels pretos, e número de transições e as usamos como referência para classificar as regiões de texto e não texto.

## 2 Implementação

Neste projeto fizemos uso da linguagem Python, e das seguintes bibliotecas para a leitura, tratamento e escrita das imagens: numpy, opencv.

A primeira etapa do projeto é realizar a leitura das imagens no formato PBM (Portable Bitmap), com o comando: `img = cv2.imread('nome-imagem', -1)`

Nesta leitura devemos ler com o parâmetro -1 pois devemos, lê-la sem alterá-la, já que a biblioteca do OpenCv tem 3 opções de leitura como descrito abaixo:

- <0 loads the image as is (including the alpha channel if present)
- 0 loads the image as an intensity one
- >0 loads the image in the BGR format

Após a leitura iremos aplicar as operações morfológicas, para permitir que nosso algoritmo de identificação de regiões e contorno tenha uma maior precisão

Mas antes de aplicar as operações, devemos tornar a imagem binária e inverter o valor dos bits, para que os objetos sejam corretamente identificados pelos métodos

As operações morfológicas utilizadas, estão todas disponíveis na biblioteca OpenCV. Para aplicar as funções, primeiramente devemos definir o elemento estruturante com o seguinte comando:

```
kernel = np.ones((kernelLines, kernelColumns), np.uint8)
```

Agora que possuímos nosso elemento estruturante, podemos aplicar ele na nossa imagem, de acordo com a operação que desejamos realizar. Neste projeto utilizamos 3 delas:

```
dilation = cv2.dilate(imagem,kernel,iterations = 1)
erosion = cv2.erode(imagem,kernel,iterations = 1)
closing = cv2.morphologyEx(imagem, cv2.MORPH_CLOSE, kernel)
```

Além destas operações também fizemos uso da intersecção entre duas imagens, que nada mais é que um bitwise AND, realizada pelo seguinte comando

```
imagem = cv2.bitwise_and(erosion1,erosion2)
```

Por fim, após tratarmos a imagem, e deixamos mais evidente as regiões de interesse, iremos utilizar um algoritmo que identifica as regiões conexas e desenha o seu contorno. Isto foi realizado com o auxílio do seguinte comando:

```
numLabels, labels, stats, centroids = cv2.connectedComponentsWithStats(image,
connectivity, cv2.CV_32S)
```

Deste retorno, nos interessamos por 2 elementos: numLabels e stats. O numLabels nos diz quantas regiões foram encontradas. O stats nos informa as coordenadas e dimensões de cada uma dessas regiões. Por fim utilizamos o stats para desenhar os contornos na imagem original, com o seguinte comando:

```
cv2.rectangle(image, (x, y), (x + width, y + height), 1, 1)
```

## 3 Resultados

### 3.1 Imagem Original

310

IEEE TRANSACTIONS ON ROBOTICS AND AUTOMATION, VOL. 7, NO. 3, JUNE 1991

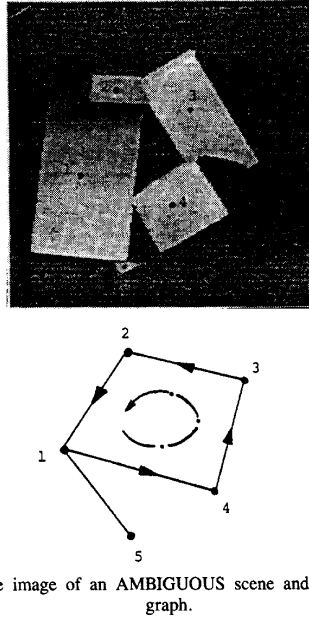


Fig. 4. Range image of an AMBIGUOUS scene and the corresponding graph.

sensory feedback is carried out in a local reflexive mode rather than in a planned mode with one exception, that is, when a pathological state is detected.

3) *States*: This is a finite set of states describing the environment of the Turing machine as perceived by the sensors. If new sensors are added, the set of states is partitioned to describe the scene as perceived by the additional sensors. For example, if a sensor capable of determining the "touch" relations of objects in the scene is added, then the set of five states, can be partitioned (a finer partition) to describe both the "touch" and "on-top-of" relations. The states of the machine are:

Empty	If there are no vertices in the diagraph, i.e., an empty diagraph.
Dispersed	If there no edges in the diagraph, i.e., a null diagraph (Fig. 2).
Overlapped	If there are at least two vertices connected with an edge (Fig. 3).
Ambiguous	If there is one or more directed cycles in the diagraph (Fig. 4).
Unstable	This category is not tested by the analysis of the graph but through analysis of the contact point/line of the object with the support plane. If this contact is a point or a line, it is classified as unstable. See Fig. 5.

Acima vemos a imagem original utilizada neste relatório a bitmap.pbm. Como podemos ver, ela possui regiões de texto e imagem. O que dificulta o processo de identificação de texto, pois temos de distinguir entre as regiões conexas quais são texto e quais são imagens. Para isto faremos o uso de algumas métricas das regiões de interesse

Para entendermos as imagens obtidas, antes precisamos compreender alguns conceitos das operações que utilizaremos

A aplicação de uma dilatação em uma imagem, tem como resultado a expansão de objetos, preenchimento de buracos e também conectar componentes próximos.

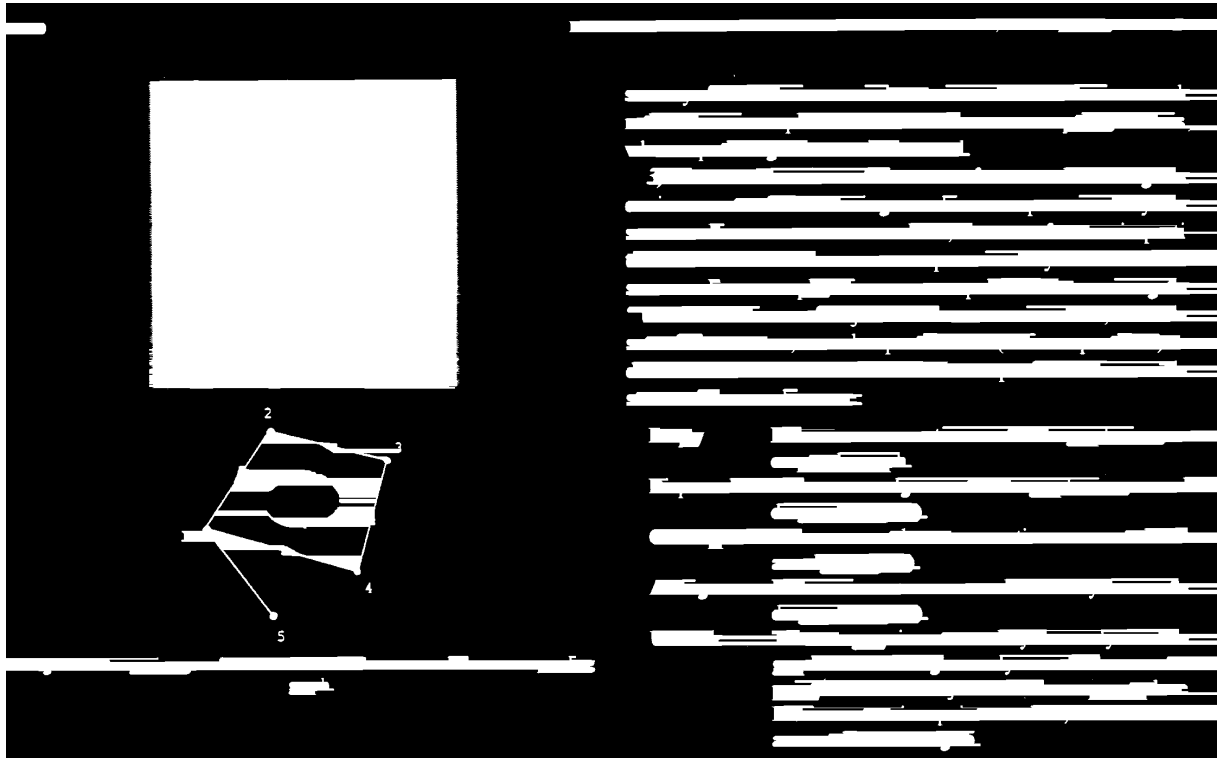
A aplicação da erosão em uma imagem tem como resultado diminuir objetos, remover partículas menores que o elemento estruturantes, aumentar buracos e também permitir a separação de componentes conectadas

O fechamento é a aplicação de uma dilatação seguida de uma erosão, que tem como resultado o fechamento de pequenos buracos e conexão de componentes na imagem

Vamos agora analisar as imagens resultantes das operações descritas acima

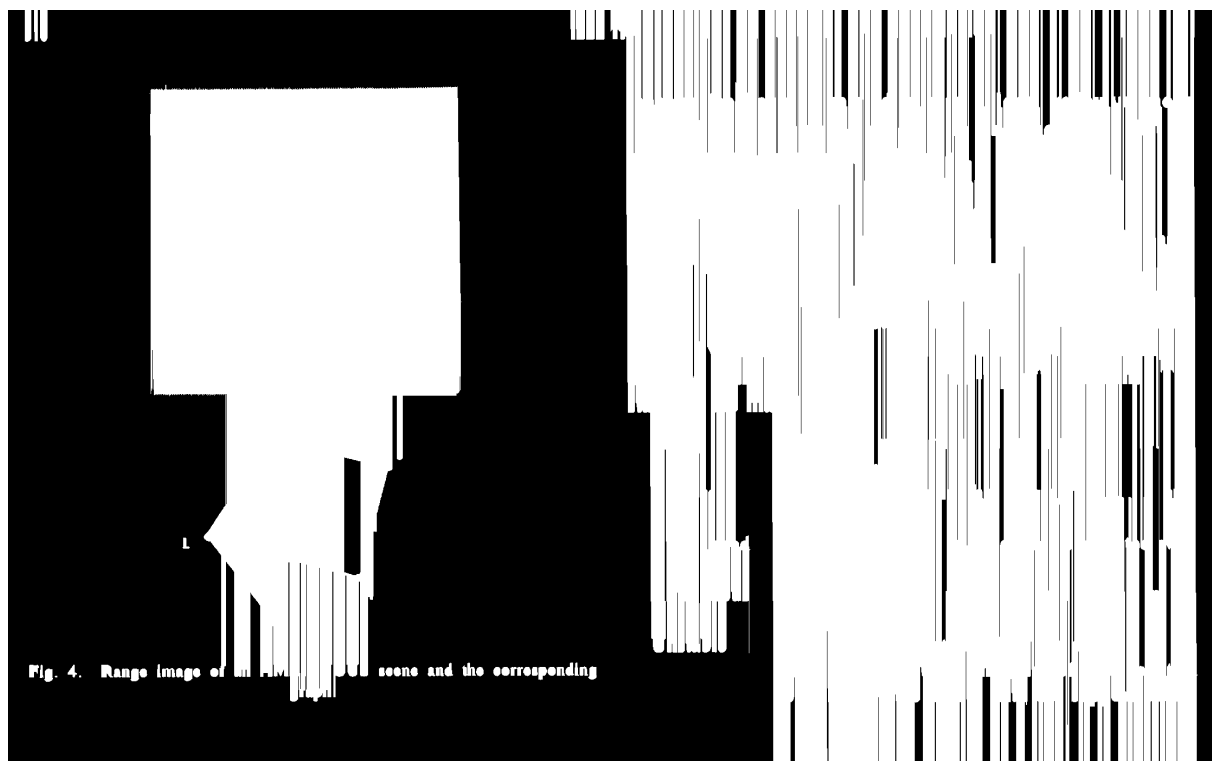
na imagem original

### 3.2 Primeiro Kernel



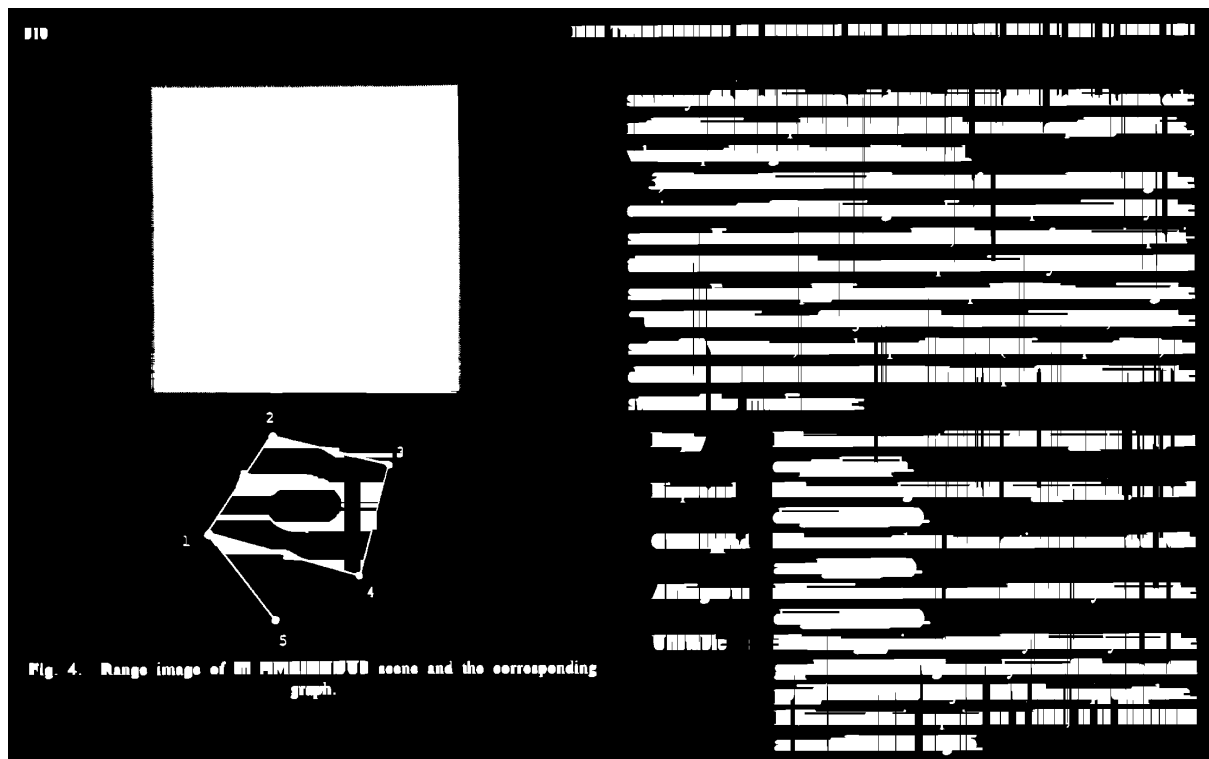
### 3.3 Segundo Kernel

A imagem acima foi obtida com a aplicação de uma dilatação seguida de uma erosão com um kernel de 1 pixel de altura e 100 de largura. Este kernel por ter uma largura muito maior que sua altura, tem como objetivo evidenciar as regiões conexas horizontalmente. Como vemos cada linha se tornou uma única região (em branco), e também as figuras de nossa imagem. Claro que temos algumas imprecisões, como a criação de linhas muito extensas. Por exemplo a união da palavra Dispersed com o texto a sua frente. Para solucionar isto utilizaremos em seguida um kernel com altura maior que a largura para identificar as uniões e separações no sentido vertical



A imagem acima foi obtida com a aplicação de uma dilatação seguida de uma erosão com um kernel de 200 pixels de altura e 1 de largura. Como comentado anteriormente, este kernel tem como objetivo evidenciar as regiões conexas verticalmente, já que sua altura é maior que sua largura. Uma informação importante a destacar aqui é o fato de que agora a palavra Dispersed esta separada da frase logo a sua frente, assim como na imagem original

### 3.4 Bitwise And



Esta imagem é o resultado da operação de bitwise\_and, entre a imagem resultante do kernel 1 com o a do kernel 2. Ou seja combinamos a nossa identificação de regiões conexas horizontais com verticais. Porém vemos que temos um ruído em nossas regiões, como por exemplo as linhas pretas verticais na região de texto, outra ruído interessante é que na frase que descreve a Fig 4, podemos ver as letras individualmente

### 3.4.1 Closing



O closing por fim é utilizado com um kernel muito menor que os outros, com 1 de altura e 30 de largura. O objetivo dele é remover os pequenos ruídos mencionados anteriormente, como remoção das pequenas linhas pretas horizontais, e tornar a frase da Fig 4 uma única componente conexa.

### 3.4.2 Identificação de regiões e contorno

Por fim após o tratamento da imagem, nós temos de utilizar um algoritmo para identificação das regiões conexas e seu contorno. O resultado obtido foi a imagem abaixo, em que vemos todas as nossas regiões conexas contornadas por um retângulo preto. Ao todo foram identificadas 54 regiões conexas, porém nem todas elas são textos, um exemplo são as 2 imagens que estão contornadas

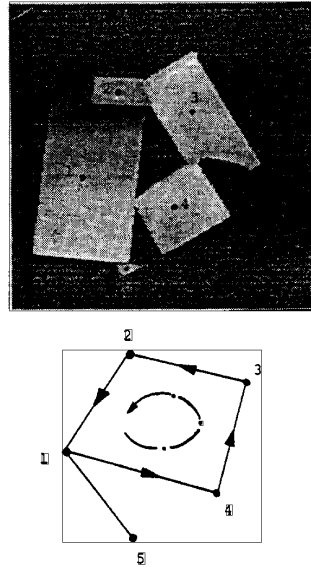


Fig. 4. Range image of an AMBIGUOUS scene and the corresponding graph.

sensory feedback is carried out in a local reflexive mode rather than in a planned mode with one exception, that is, when a pathological state is detected.

3) *States*: This is a finite set of states describing the environment of the Turing machine as perceived by the sensors. If new sensors are added, the set of states is partitioned to describe the scene as perceived by the additional sensors. For example, if a sensor capable of determining the "touch" relations of objects in the scene is added, then the set of five states, can be partitioned (a finer partition) to describe both the "touch" and "on-top-of" relations. The states of the machine are:

Empty	If there are no vertices in the diagraph, i.e., an empty diagraph.
Dispersed	If there no edges in the diagraph, i.e., a null diagraph (Fig. 2).
Overlapped	If there are at least two vertices connected with an edge (Fig. 3).
Ambiguous	If there is one or more directed cycles in the diagraph (Fig. 4).
Unstable	This category is not tested by the analysis of the graph but through analysis of the contact point/line of the object with the support plane. If this contact is a point or a line, it is classified as unstable. See Fig. 5.

O passo seguinte é determinar a densidade de pixels pretos em cada uma das regiões e o número de transições de branco para preto em relação ao total de pontos pretos e utilizar estas duas métricas para determinar quais regiões são texto e quais são imagens

Escolhendo todos as densidades de pixel entre 0.2 ; densidade ; 0.5 e a taxa de transição ; 0.25, obtemos a seguinte imagem



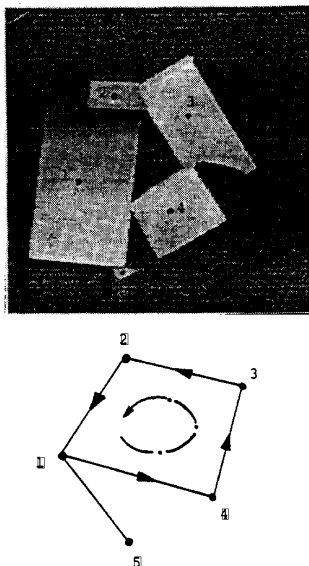


Fig. 4. Range image of an AMBIGUOUS scene and the corresponding graph.

sensory feedback is carried out in a local reflexive mode rather than in a planned mode with one exception, that is, when a pathological state is detected.

3) *States*: This is a finite set of states describing the environment of the Turing machine as perceived by the sensors. If new sensors are added, the set of states is partitioned to describe the scene as perceived by the additional sensors. For example, if a sensor capable of determining the "touch" relations of objects in the scene is added, then the set of five states, can be partitioned (a finer partition) to describe both the "touch" and "on-top-of" relations. The states of the machine are:

Empty	If there are no vertices in the diagraph, i.e., an empty diagraph.
Dispersed	If there no edges in the diagraph, i.e., a null diagraph (Fig. 2).
Overlapped	If there are at least two vertices connected with an edge (Fig. 3).
Ambiguous	If there is one or more directed cycles in the diagraph (Fig. 4).
Unstable	This category is not tested by the analysis of the graph but through analysis of the contact point/line of the object with the support plane. If this contact is a point or a line, it is classified as unstable. See Fig. 5.

Agora, como exercício do item 10, temos a tarefa de identificar as palavras individualmente, não apenas linhas. Para isto, eu repeti todas as etapas descritas anteriormente, mas aplicada a região conexa de cada linha.

Nesta etapa, tive um problema com os pingos nos Is, em que alguns deles eram identificados como regiões conexas distintas da letra I. Para solucionar este problema, eu utilizei as mesmas estatísticas do item 9, para distinguir o que era uma letra e o que era apenas um pingo.

Utilizando como limiar 0.1  $\rho$ densidade  $\rho$  0.7, obtemos a seguinte imagem:

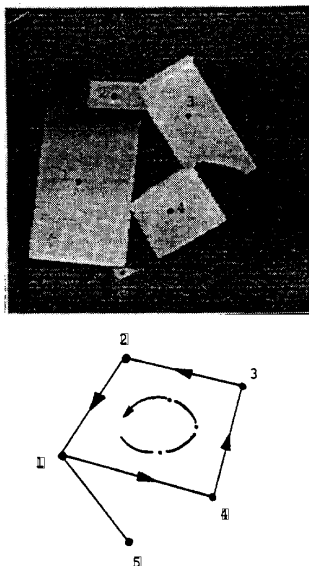


Fig. 4. Range image of an AMBIGUOUS scene and the corresponding graph.

sensory feedback is carried out in a local reflexive mode rather than in a planned mode with one exception, that is, when a pathological state is detected.

3) *States*: This is a finite set of states describing the environment of the Turing machine as perceived by the sensors. If new sensors are added, the set of states is partitioned to describe the scene as perceived by the additional sensors. For example, if a sensor capable of determining the "touch" relations of objects in the scene is added, then the set of five states, can be partitioned (a finer partition) to describe both the "touch" and "on-top-of" relations. The states of the machine are:

Empty	If there are no vertices in the diagraph, i.e., an empty diagraph.
Dispersed	If there no edges in the diagraph, i.e., a null diagraph (Fig. 2).
Overlapped	If there are at least two vertices connected with an edge (Fig. 3).
Ambiguous	If there is one or more directed cycles in the diagraph (Fig. 4).
Unstable	This category is not tested by the analysis of the graph but through analysis of the contact point/line of the object with the support plane. If this contact is a point or a line, it is classified as unstable. See Fig. 5.

Nesta última imagem, vemos que cada palavra foi selecionada individualmente.

### 3.5 Conclusão

O objetivo do projeto era relizar a identificação das linhas e palavras, com operadoes morfológicos. Nas duas ultimas imagens, vemos que conseguimos com sucesso realizar a identificação das regiões de interesse.

Uma das maiores dificuldades foi determinal quais seriam os kernels utilizados para preparar a imagem para o algoritmo de detecção de regiões conexas. Os primeiros kernels para identificação de linha foram dados. Porém para a segunda etapa de identificação de palavras foi mais difícil, e tive de testar alguns valores antes de separar corretamente as palavras

Outra parte que exigiu análise da imagem, foi definir os limiares para separar os pingos dos Is, e também separar o que eram regiões de texto e o que eram de imagem.

## References

- [1] Operadores morfológicos: <http://www.inf.ufpr.br/lesoliveira/download/morfologia.pdf>
- [2] Basics                      OpenCv:                      [https://docs.opencv.org/3.0-beta/doc/py\\_tutorials/py\\_gui/py\\_image\\_display/py\\_image\\_display.html](https://docs.opencv.org/3.0-beta/doc/py_tutorials/py_gui/py_image_display/py_image_display.html)