

Trabalho 3 - MC920

GUSTAVO HENRIQUE STORTI SALIBI

RA: 174135 / Ciência da Computação - Graduação

E-mail: gustavohstorti@gmail.com

I. INTRODUÇÃO

A morfologia matemática consiste em uma metodologia para análise de imagens que permite a construção de operadores úteis para a descrição de objetos em imagens.

Os operadores morfológicos podem ser utilizados em um grande número de aplicações em processamento e análise de imagens, tal como a extração de componentes conexos.

O objetivo deste trabalho é aplicar operadores morfológicos para segmentar regiões compreendendo texto e não texto em uma imagem de entrada.

A seção seguinte, II, explica de forma geral o problema a ser tratado neste trabalho. A seção III fala sobre como o programa lida com as entradas de dados e como ele é construído. A seção IV, por sua vez, mostra os resultados obtidos e os explica. A seção V, por fim, explica como e onde as saídas são salvas.

II. ESPECIFICAÇÃO DO PROBLEMA

Para atingir a solução do problema, 10 etapas deverão ser concluídas:

- 1) Dilatação da imagem original com um elemento estruturante de 1 pixel de altura e 100 pixels de largura;
- 2) Erosão da imagem resultante com o mesmo elemento estruturante do passo (1);
- 3) Dilatação da imagem original com um elemento estruturante de 200 pixels de altura e 1 pixel de largura;
- 4) Erosão da imagem resultante com o mesmo elemento estruturante do passo (3);
- 5) Aplicação da intersecção (AND) dos resultados dos passos (2) e (4);
- 6) Fechamento do resultado obtido no passo (5) com um elemento estruturante de 1 pixel de altura e 30 pixels de largura;
- 7) Aplicação de algoritmo para identificação de componentes conexos sobre o resultado do passo (6);
- 8) Para cada retângulo envolvendo um objeto, calcular:
 - a. razão entre o número de pixels pretos e o número total de pixels (altura * largura);
 - b. razão entre o número de transições verticais e horizontais branco para preto e o número total de pixels pretos.

- 9) Criação de uma regra para classificar cada componente conexo, de acordo com as medidas obtidas no passo (8), como texto e não texto;
- 10) Aplicação de operadores morfológicos apropriados para segmentar cada linha do texto em blocos de palavras. Coloque um retângulo envolvendo cada palavra na imagem original. Calcule o número total de linhas de texto e de blocos de palavras na imagem.

III. ENTRADA DE DADOS

O programa foi desenvolvido na linguagem Python e testado com o interpretador Python 3.7. Foram também utilizadas as seguintes bibliotecas extras: OpenCV 3.2, Numpy 1.12.1 e scikit-image 0.14.2.

Para o correto funcionamento, a imagem de entrada deve ser colocada na pasta "imagens/entrada" com o título "imagem_entrada.pbm" (já há uma imagem de exemplo).

Para executar o programa, basta utilizar o seguinte comando:

```
python3 programa.py
```

Após isso, o programa processará os dados exibindo as saídas na tela e salvando as imagens de saída no diretório "imagens/saída".

A. Código

O código começa por importar as bibliotecas necessárias. Depois disso, são criadas as seguintes funções:

- 1) `conta_transicoes`: Retorna quantidade de transições dentro de um intervalo informado por parâmetro para uma imagem.
- 2) `cria_mascara`: Dadas informações sobre os componentes conexos encontrados, retorna uma máscara binária como uma matriz com posições referentes ao que deve ser contornado com o valor 1.
- 3) `desenha_saida`: Recebe uma imagem, desenha sobre ela as marcações necessárias e retorna a imagem desenhada.

Após isso, o programa executa ações para cada passo pedido no enunciado. Há no código um comentário para cada parte do enunciado que está sendo implementada, o dividindo assim em 10 seções.

IV. SOLUÇÃO

Tomaremos como base para comparação a seguinte imagem original:

310

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

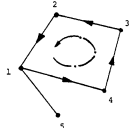
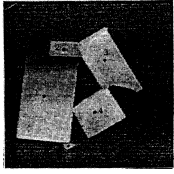


Fig. 4. Range image of an AMBIGUOUS scene and the corresponding graph.

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 diagram, i.e., an empty diagram.
Dispersed	If there are no edges in the diagram, i.e., a null diagram (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 diagram (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.

Figura 1. Bitmap. Imagem original.

A. Cálculo das razões

Tanto a razão entre o número de pixels pretos e o número total de pixels quanto a razão entre o número de transições verticais e horizontais preto para branco e o número total de pixels pretos são impressas na tela para cada retângulo encontrado durante a execução do programa. Nesse caso, a saída foi:

```

1: pretos/total=0.8407 | trans./pretos=0.0387
2: pretos/total=0.9694 | trans./pretos=0.0058
3: pretos/total=0.7500 | trans./pretos=0.3333
4: pretos/total=1.0000 | trans./pretos=0.0000
5: pretos/total=0.9756 | trans./pretos=0.0058
6: pretos/total=0.6217 | trans./pretos=0.0501
7: pretos/total=0.6604 | trans./pretos=0.0383
8: pretos/total=0.6612 | trans./pretos=0.0433
9: pretos/total=0.6892 | trans./pretos=0.0379
10: pretos/total=0.5962 | trans./pretos=0.0445
11: pretos/total=0.9000 | trans./pretos=0.0556
12: pretos/total=0.5864 | trans./pretos=0.0542
13: pretos/total=0.8500 | trans./pretos=0.1765
14: pretos/total=0.6608 | trans./pretos=0.0398
15: pretos/total=0.5999 | trans./pretos=0.0439
16: pretos/total=0.6667 | trans./pretos=0.0436
17: pretos/total=0.6306 | trans./pretos=0.0438
18: pretos/total=0.7249 | trans./pretos=0.0369
19: pretos/total=1.0000 | trans./pretos=0.0000
20: pretos/total=0.7409 | trans./pretos=0.0494
21: pretos/total=0.5083 | trans./pretos=0.3115
22: pretos/total=0.6615 | trans./pretos=0.0432
23: pretos/total=0.1709 | trans./pretos=0.0951
24: pretos/total=0.7149 | trans./pretos=0.0453
25: pretos/total=0.7407 | trans./pretos=0.0414
26: pretos/total=1.0000 | trans./pretos=0.0000
27: pretos/total=0.6605 | trans./pretos=0.0563
28: pretos/total=0.5835 | trans./pretos=0.0627
29: pretos/total=0.9107 | trans./pretos=0.0784
30: pretos/total=0.8462 | trans./pretos=0.0212
31: pretos/total=0.7467 | trans./pretos=0.0454

```

```

32: pretos/total=0.8000 | trans./pretos=0.2500
33: pretos/total=0.7640 | trans./pretos=0.0419
34: pretos/total=0.3615 | trans./pretos=0.3191
35: pretos/total=0.7832 | trans./pretos=0.0308
36: pretos/total=0.6304 | trans./pretos=0.0489
37: pretos/total=0.6161 | trans./pretos=0.0523
38: pretos/total=0.5219 | trans./pretos=0.2353
39: pretos/total=0.8431 | trans./pretos=0.0192
40: pretos/total=0.8000 | trans./pretos=0.3000
41: pretos/total=0.6299 | trans./pretos=0.0519
42: pretos/total=0.4881 | trans./pretos=0.3659
43: pretos/total=0.8814 | trans./pretos=0.0501
44: pretos/total=1.0000 | trans./pretos=0.0000
45: pretos/total=1.0000 | trans./pretos=0.0000
46: pretos/total=1.0000 | trans./pretos=0.0000
47: pretos/total=0.6369 | trans./pretos=0.0486
48: pretos/total=0.5702 | trans./pretos=0.0559
49: pretos/total=0.4892 | trans./pretos=0.0941
50: pretos/total=0.9375 | trans./pretos=0.1333
51: pretos/total=0.5822 | trans./pretos=0.0778
52: pretos/total=0.6747 | trans./pretos=0.0431
53: pretos/total=1.0000 | trans./pretos=0.0000
54: pretos/total=0.7500 | trans./pretos=0.3333
55: pretos/total=0.6440 | trans./pretos=0.0381
56: pretos/total=0.6789 | trans./pretos=0.0357

```

As razões impressas se referem aos retângulos enumerados salvos na saída "saida_8.png":

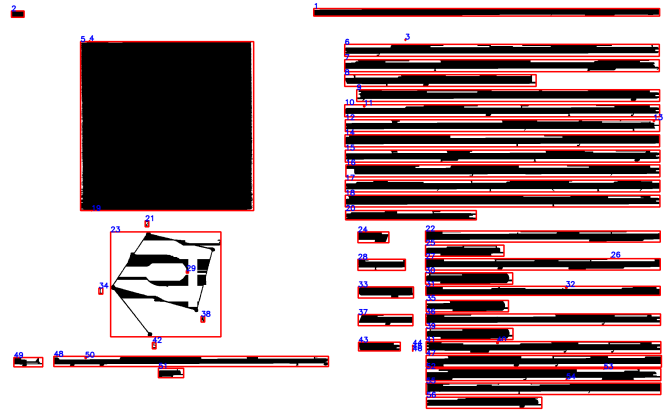


Figura 2. saida_8.png

B. Classificação de componentes conexos

A próxima imagem de saída ilustra os componentes conexos, de acordo com as medidas obtidas no passo (8), como texto (com retângulo em torno) e não texto:

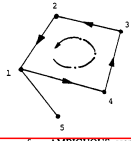
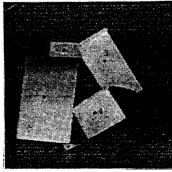


Fig. 4. Range image of an AMBIGUOUS scene and the corresponding graph.

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 diagram, i.e., an empty diagram.
Dispersed	If there are no edges in the diagram, i.e., a null diagram (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 diagram (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.

Figura 3. saída_9.png

C. Segmentação em blocos de palavras

A imagem de saída seguinte ilustra a segmentação de todas as linhas do texto em blocos conexos de palavras:

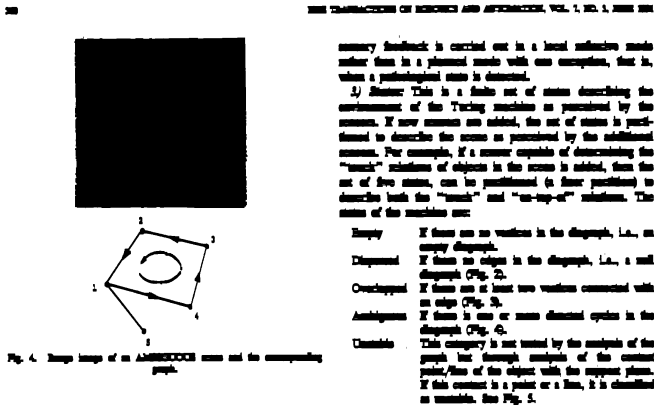


Figura 4. saída_10-blocos-palavras.png

D. Retângulos envolvendo palavras

A próxima imagem de saída mostra um retângulo envolvendo cada palavra na imagem original:

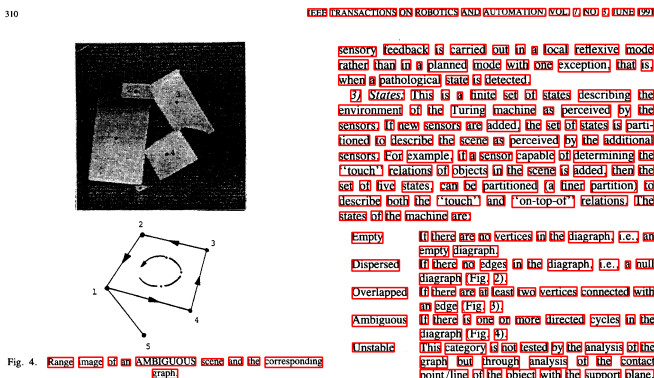


Fig. 4. Range image of an AMBIGUOUS scene and the corresponding graph.

Figura 5. saída_10-palavras-com-retangulo.png

E. Número de linhas e de palavras

Por fim, a imagem a seguir mostra a saída impressa no terminal contendo o número total de linhas e de palavras encontradas para a imagem original:

```

retangulo 00: pretos/total=0.6789 | transicoes/pretos=0.0357
Total de linhas: 33
Total de blocos de palavras: 234

```

Figura 6. Saída final do terminal.

V. SAÍDA DE DADOS

Além da saída em texto no terminal, o programa gera quatro imagens em PNG, referentes a cada uma das técnicas descritas. Elas são salvas no diretório imagem/saída com os nomes respectivos às técnicas.