
Bildverarbeitung SoSe 2012

Übung 8

Max Michels, Sebastian Kürten

1 Aufgabe 1

siehe letztes Blatt

2 Aufgabe 2

2.1 Teil a

Die Hauptdatei von Aufgabe 2a führt die Operationen Erosion, Dilation, Opening und Closing auf dem Binärbild „rhino-part-bin.png“ aus und speichert die Ergebnisse in Dateien ab. Es wird jeweils ein kreuzförmiges 3x3-Strukturelement verwendet. Da dieses Strukturelement symmetrisch ist, musste es nicht transponiert werden.

Listing 1: Hauptdatei a2a.m

```
1 I = imread("rhino-part-bin.png");
2
3 S = [0 1 0; 1 1 1; 0 1 0];
4
5 Id = dilation(I, S);
6 Ie = erosion(I, S);
7 Ic = closing(I, S);
8 Io = opening(I, S);
9
10 imwrite(Id, "rhino-part-dilation.png");
11 imwrite(Ie, "rhino-part-erosion.png");
12 imwrite(Ic, "rhino-part-closing.png");
13 imwrite(Io, "rhino-part-opening.png");
```

Erosion und Dilation wurden wie folgt implementiert.

Listing 2: Erosion

```
1 function O = erosion(I, S)
2     [h, w] = size(I);
3     [sh, sw] = size(S);
4     oh = h - sh;
5     ow = w - sw;
6     O = zeros(oh, ow);
7     for i = [1:oh]
8         for j = [1:ow]
9             P = I(i:i+sh-1, j:j+sw-1) & S;
10            O(i, j) = max(P(:));
11        end;
12    end;
13 end;
```

Listing 3: Dilation

```

1 function O = dilation(I, S)
2     [h, w] = size(I);
3     [sh, sw] = size(S);
4     oh = h - sh;
5     ow = w - sw;
6     O = zeros(oh, ow);
7     for i = [1:oh]
8         for j = [1:ow]
9             P = (I(i:i+sh-1, j:j+sw-1) & S) == S;
10            O(i, j) = min(P(:));
11        end;
12    end;
13 end;

```

Opening und Closing sind eine einfache hintereinanderausführung von Erosion und Dilation bzw. umgekehrt.

Listing 4: Opening

```

1 function O = opening(I, S)
2     O = dilation(erosion(I, S), S);
3 end;

```

Listing 5: Closing

```

1 function O = closing(I, S)
2     O = erosion(dilation(I, S), S);
3 end;

```

2.2 Teile b und c

Für die Aufgabenteile b und c wurden die unten stehenden Bilder erzeugt und dann mit dem folgenden Programm wiederhergestellt.

Listing 6: Hauptdatei a2bc.m

```

1 Iw = imread("text-with-white-lines.png");
2 Ib = imread("text-with-black-lines.png");
3
4 S = [0 1 0; 1 1 1; 0 1 0];
5
6 Ic = closing(Iw, S);
7 Io = opening(Ib, S);
8
9 imwrite(Ic, "text-white-restored.png");
10 imwrite(Io, "text-black-restored.png");

```

Es wurde wiederum das 3x3 Kreuz als Strukturelement verwendet und die Linien wurden mittels Closing bzw. Opening entfernt.

This is current driver circuit.
Phil:
22-9-71

(a) mit weißen Linien

This is current driver circuit.
Phil:
22-9-71

(b) mit schwarzen Linien

Abbildung 1: modifizierte Bilder (die weißen Linien sind an den gleichen Stellen wie die schwarzen)

This is current driver circuit.
Phil:
22-9-71

(a) mit weißen Linien

This is current driver circuit.
Phil:
22-9-71

(b) mit schwarzen Linien

Abbildung 2: wiederhergestellte Bilder

3 Aufgabe 3

3.1 Teil a

Alle Operationen wurden hier analog zu Aufgabe 2 implementiert. Lediglich die Zeilen 9 und 10 mussten in der Erosion bzw. Dilation angepasst werden. In der Hauptdatei werden die vier Operationen auf „lena-bw.png“ ausgeführt und abgespeichert. Die Ergebnisse sind weiter unten zu sehen.

Listing 7: Hauptdatei a3.m

```
1 I = double(imread("lena-bw.png"));  
2  
3 S = [NaN NaN 0 NaN NaN; NaN 0 2 0 NaN; 0 2 4 2 0; NaN 0 2 0 NaN; NaN NaN 0 ...  
      NaN NaN];  
4  
5 Id = gs_dilation(I, S);  
6 Ie = gs_erosion(I, S);  
7 Ic = gs_closing(I, S);
```

```

8 Io = gs_opening(I, S);
9
10 imwrite(uint8(Id), "lena-bw-dilation.png");
11 imwrite(uint8(Ie), "lena-bw-erosion.png");
12 imwrite(uint8(Ic), "lena-bw-closing.png");
13 imwrite(uint8(Io), "lena-bw-opening.png");

```

Listing 8: Grauwert Erosion

```

1 function O = gs_erosion(I, S)
2     [h, w] = size(I);
3     [sh, sw] = size(S);
4     oh = h - sh;
5     ow = w - sw;
6     O = zeros(oh, ow);
7     for i = [1:oh]
8         for j = [1:ow]
9             P = I(i:i+sh-1, j:j+sw-1) - S;
10            O(i, j) = min(P(:));
11        end;
12    end;
13 end;

```

Listing 9: Grauwert Dilation

```

1 function O = gs_dilation(I, S)
2     [h, w] = size(I);
3     [sh, sw] = size(S);
4     oh = h - sh;
5     ow = w - sw;
6     O = zeros(oh, ow);
7     for i = [1:oh]
8         for j = [1:ow]
9             P = I(i:i+sh-1, j:j+sw-1) + S;
10            O(i, j) = max(P(:));
11        end;
12    end;
13 end;

```

Opening und Closing sind eine einfache hintereinanderausführung von Erosion und Dilation bzw. umgekehrt.

Listing 10: Grauwert Opening

```

1 function O = gs_opening(I, S)
2     O = gs_dilation(gs_erosion(I, S), S);
3 end;

```

Listing 11: Grauwert Closing

```

1 function O = gs_closing(I, S)
2     O = gs_erosion(gs_dilation(I, S), S);
3 end;

```



(a) Erosion



(b) Dilation

Abbildung 3: Grauwert-Bilderoperationen



(a) Opening



(b) Closing

Abbildung 4: Grauwert-Bilderoperationen