

Praktische Übung: Matlab für die medizinische und industrielle Bildinterpretation

Versuch 7: Morphologische Operationen

Aufgabe 1 - Bearbeitung von binären Masken

Binäre Masken, wie sie zum Beispiel bei der Segmentierung entstehen, benötigen oftmals weitere Verarbeitungsschritte, um Störungen zu entfernen. Hierbei sind vor allem Kombinationen der grundlegenden morphologischen Operationen Dilatation ($M \oplus S$) und Erosion ($M \ominus S$) einer Maske M mit einem Strukturelement S hilfreich.

- a) Machen Sie sich mit den Operationen Dilatation und Erosion vertraut. Hierzu können die Masken einiger Formen aus der Datei `shapes.mat` geladen werden. Sie können die MATLAB-eigenen Funktionen `imdilate` und `imerode` verwenden. Strukturelemente können mit dem Befehl `strel` automatisch erstellt werden oder per Hand als Matrix $S \in \{0, 1\}^{n \times m}$ definiert werden.

- b) Closing ($M \bullet S = (M \oplus S) \ominus S$) kann verwendet werden, um Löcher in Masken zu schließen.

Nutzen Sie diese Operation um die Löcher in dem Bild `silhouette_holes.jpg` zu schließen. Wandeln Sie das Bild dazu zunächst mit dem Befehl `imbinarize` in eine (binäre) Maske um.

- c) Opening ($M \odot S = (M \ominus S) \oplus S$) wird verwendet um unerwünschte Strukturen im Bild zu entfernen.

Nutzen Sie verschiedene Strukturelemente, um in dem Bild `lines.jpg` Linien mit gleicher Ausrichtung zu extrahieren.

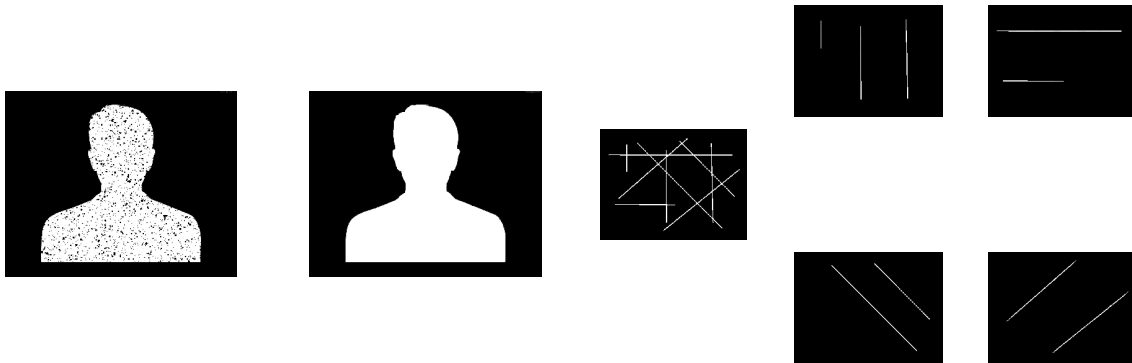


Abbildung 1: Beispielhafte Ergebnisse von Aufgabenteil b) und c)

- d) Kombinieren Sie die obigen Ansätze, um das Bild `silhouette_corrupted.jpg` zu rekonstruieren. Vergleichen Sie ihr Ergebnis mit dem Original.

Hinweis: Es ist nicht gefordert eine perfekte Rekonstruktion der Maske zu erstellen, ein grobes Entfernen der Fehler genügt.



Abbildung 2: Beispielhafte Rekonstruktionsschritte

Aufgabe 2 - Morphologie und Nachbarschaften

Dilatation und Erosion ermöglichen einen einfachen Zugriff auf angrenzende Bereiche einer Maske, wobei verschiedene Nachbarschaften (4er-, 8er-Nachbarschaft, ...) über das Strukturelement definiert werden können.

- a) Nutzen Sie Erosion, um den Rand δM einer Maske M zu bestimmen und Dilatation, um die Nachbarn $N(M)$ der Maske zu bestimmen.

Es gilt

$$\delta M = M - (M \ominus S)$$

und

$$N(M) = (M \oplus S) - M.$$

- b) Schreiben Sie eine Funktion

`M = regionGrowingMorphology(I,xStart,yStart,threshold),`

welche die Steppenbrandvariante des Region Growing Algorithmus umsetzt, wobei Sie Dilatation für eine einfache Bestimmung der nächsten Nachbarn benutzen. Setzen Sie die folgenden Schritte um:

1. Initialisierung der (Segmentierungs-)Maske M in der Größe des Bildes I
Die Maske enthält bis auf den Saatpunkt (x_{start}, y_{start}) nur Nullen. Casten Sie ihre Maske falls nötig auf `logical`. Sie können dann direkt mit der Maske auf alle segmentierten Punkte im Bild zugreifen, indem Sie $I(M)$ verwenden.
2. Bestimmung des Mittelwertes μ in der segmentierten Region M
3. Bestimmung der nächsten Nachbarn $N(M)$
4. Aufnahme der ähnlichen Nachbarn (Abstand unterhalb des Thresholds) in die segmentierte Region:

$$M = M \cup (N \cap \{|I - \mu| < threshold\})$$

Hinweis: Dieser Schritt lässt sich in MATLAB in einer Zeile umsetzen.

5. Wiederholung von 2.-4. solange neue Punkte zur Segmentierung hinzugefügt werden

Testen Sie Ihre Funktion und vergleichen Sie auch die Laufzeit mit Ihrer Originalversion aus dem 3. Versuch. (MATLAB: `tic`, `toc`).

Aufgabe 3 - Distanztransformation

Über die iterative Bestimmung des Randes beziehungsweise der Nachbarn einer Region, kann eine vorzeichenbehaftete Distanztransformation bestimmt werden.

- a) Schreiben Sie eine Funktion `D = distanceTransform(M)`, welche die Distanztransformation bestimmt. D soll hierbei den jeweils (kleinsten) Abstand zum Rand der Maske $M = M_0$ enthalten. Das Vorzeichen gibt an, ob ein Punkt innerhalb oder außerhalb der Maske liegt. Für Punkte innerhalb der Maske gilt demnach

$$D(\delta M_i) = i \quad \text{mit} \quad M_i = M_{i+1} \ominus S, \quad i < 0$$

und für Punkte außerhalb

$$D(N(M_j)) = j \quad \text{mit} \quad M_j = M_{j-1} \oplus S, \quad j > 0$$

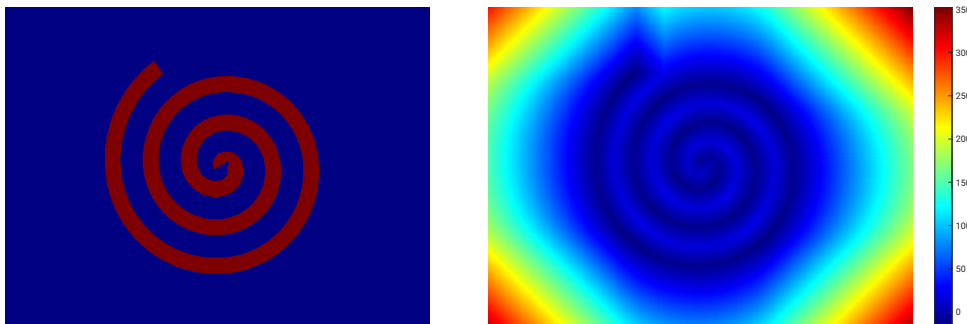


Abbildung 3: Anwendung der Distanztransformation auf die Spirale

- b) Die Distanztransformation kann z.B. dazu verwendet werden zwei Masken linear zu interpolieren. Vervollständigen Sie die Funktion `morphing(A, B, n)`, die in n Schritten die Maske A in die Maske B überführt. Die Zwischenschritte C_i lassen sich dabei wie folgt bestimmen:

$$D_i = \left(1 - \frac{i}{n}\right) D_A + \frac{i}{n} D_B$$

$$C_i(x, y) = 1 \Leftrightarrow D_i(x, y) \leq 0$$

wobei D_A und D_B die Distanztransformationen von A und B sind. Die Funktion gibt ein Video des Morphingvorgangs zurück. Sie können dieses erneut mit dem Befehl `movie` abspielen.



Abbildung 4: Morphingvorgang zwischen zwei Shapes

Kontrollfragen

- Welche Bereiche sind mit morphologischen Operationen besonders schwierig zu rekonstruieren (siehe Aufgabe 1)?
- Angenommen Sie haben ein Bild mit Erbsen und Stöckern (siehe Vorlesung). Welche Eigenschaften müssen erfüllt sein, damit die Erbsen im Bild über morphologische Operationen gefunden werden können?
- Die Distanztransformation aus Aufgabe 3 verwendet keine euklidischen Abstände. Welche Norm wird stattdessen bei 4er- bzw. 8er-Nachbarschaft verwendet?