

Aufgabe 1

Die Berechnung eines Histogramms eines Graubildes wurde erfolgreich implementiert. Für die gegebenen Bilder 'lena1.bmp '1, 'lena2.bmp '2, 'lena3.bmp '3 wurde ein Histogramm erstellt und mit der octave-Funktion bar veranschaulicht, siehe Abbildung 4 bis 6. Anschließend wurde ein Histogrammausgleich ausgeführt. Die Ergebnisse sind in den Abbildungen 7 bis 9 dargestellt. Zusätzlich wurden die Histogramme dieser Bilder errechnet und in den Abbildungen 10 bis 11 gezeigt. Alle Histogramme zeigen ein recht kontrastarmes Bild, mit verschiedenen Helligkeiten. Nach dem Histogrammausgleich zeigen die Bilder einen guten Kontrast, was auch in den Histogrammen zu diesen deutlich wird.

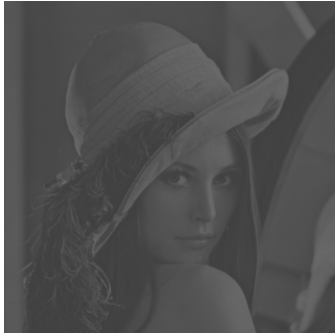


Abbildung 1: lena1.bmp

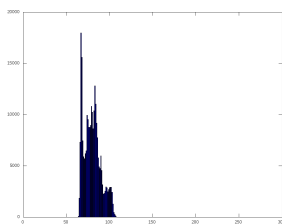


Abbildung 4: Histogramm von Abb. 1

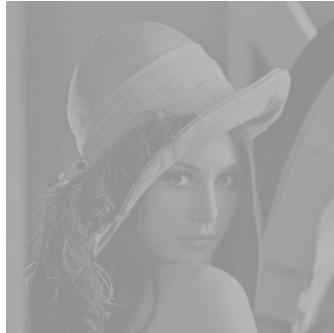


Abbildung 2: lena2.bmp

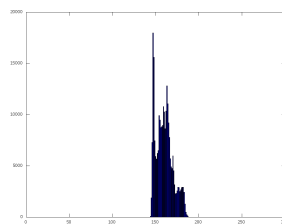


Abbildung 5: Histogramm von Abb. 2

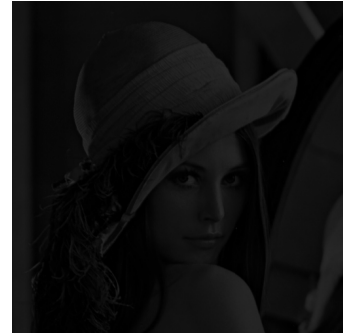


Abbildung 3: lena3.bmp

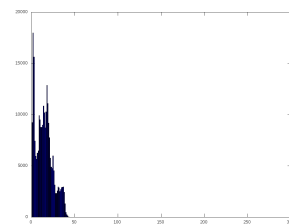


Abbildung 6: Histogramm von Abb. 3



Abbildung 7: Ausgleich von Abb. 1



Abbildung 8: Ausgleich von Abb. 2



Abbildung 9: Ausgleich von Abb. 3

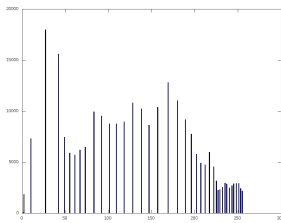


Abbildung 10: Histogramm
von 7

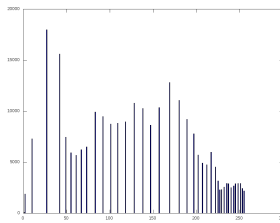


Abbildung 11: Histogramm
von 8

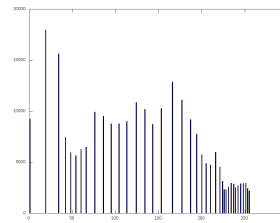


Abbildung 12: Histogramm
von 9

Aufgabe 2

Der Canny-Edge-Detektor wurde erfolgreich implementiert und getestet. Um zu zeigen, dass der Filter in allen Richtungen richtig arbeitet wurde er zunächst auf einen Kreis angewendet, der in Abbildung 13 gezeigt ist. Das Ergebnis 14 ist eine durchgängige dünne Kreislinie, die zeigt, dass der Algorithmus an jeder Stelle im Bild richtig funktioniert.

Mit diesem Wissen kann der Algorithmus auf das Bild 'falten.bmp' 15 angewendet werden (`Canny(Bild, 3, 2, 60, 10)`). Zuerst wurde es mit einem Gaussfilter der Größe 3 und $\sigma = 2$ geglättet. Das Ergebnis hiervon ist in Abbildung 16 gezeigt. Anschließend wurde der Sobel- und Sobely-Operator auf das geglättete Bild angewendet, was in den Abbildungen 17 und 18 zu sehen ist. Die folgenden beiden Bilder 19 und 20 zeigen den absolutbetrag der Sobeloperatoren

$$R = \sqrt{sx^2 + sy^2}$$

und die Richtung

$$T = \arctan\left(\frac{sy}{sx}\right),$$

wobei wir dies mit der Funktion `atan2(sy, sx)` umgesetzt haben, um uns nicht um Null-division etc. kümmern zu müssen. Mit diesen Beiden Bildern konnte nun die Non-Maxima-Suppression durchgeführt werden, die eine dünne Linie des Maximas hinterlässt, was in Bild 21 gezeigt ist. Nun wurde das Bild mit $T1 = 60$ geschwellwertet. Ausgehend von diesen gefundenen Punkten wurde eine Art 'Floodfill' durchgeführt, der die Linien von 21 entlangfährt, solange der Schwellwert $T2 = 10$ überschritten ist. Das Endergebnis ist das gesuchte Bild mit den Kanten, das in Abbildung 23 dargestellt ist. Vergleicht man die letzten beiden Bilder erkennt deutlich, dass kurze Kanten richtig verlängert werden (vor allem im Kinnbereich).

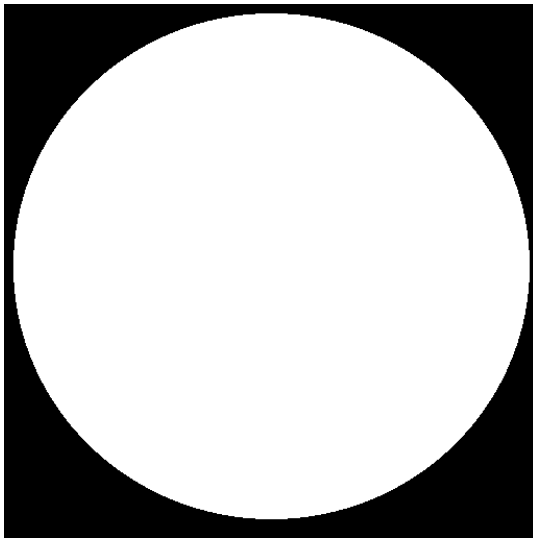


Abbildung 13: Originalbild, 'Kreis.png'.

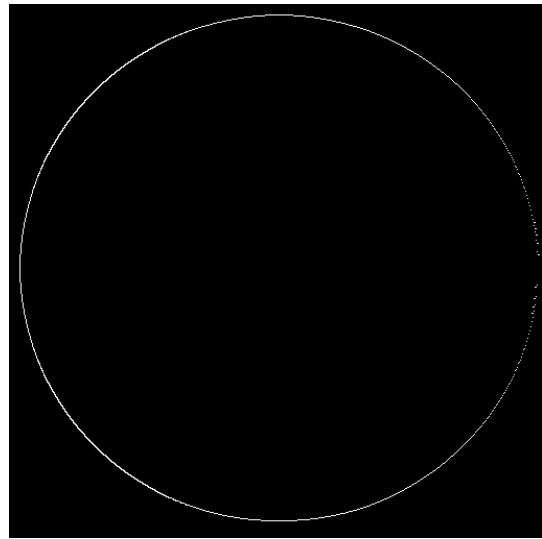


Abbildung 14: Canny angewendet auf 13



Abbildung 15: Originalbild, 'falten.bmp'.



Abbildung 16: Gausfilter auf Abb. 15



Abbildung 17: Sobelxfilter auf Abb. 16

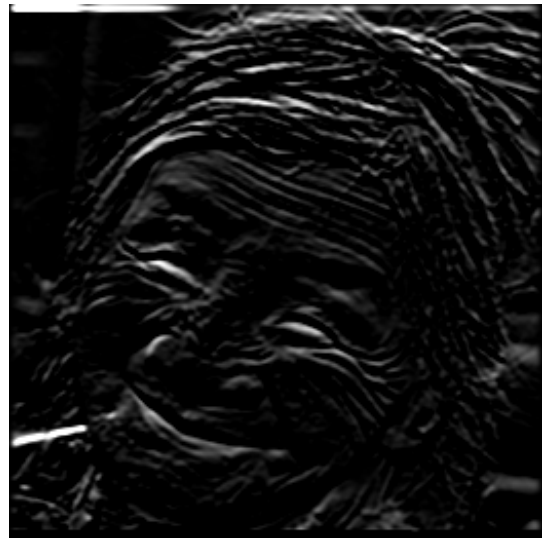


Abbildung 18: Sobelyfilter auf Abb. 16



Abbildung 19: Absolutbetrag von s_x und s_y



Abbildung 20: $\text{ArcTan}(s_y/s_x)$



Abbildung 21: Non-Maximum-Supression



Abbildung 22: Tresholdfilter von 21



Abbildung 23: Histeresefilter von 22

Aufgabe 3

In dieser Aufgabe sind die morphologischen Operatoren "dilate", "erode", "open" und "close" zu implementieren und auf das Ergebnis der Kantenerkennung aus Aufgabe 2 (Abbildung 23) anzuwenden.

Der Operator "dilate" ist so definiert, dass man all die Pixel im Zielbild belässt, wo mindestens ein Pixel von dem Kernel des morphologischen Operators und dem aktuell betrachteten Bildbereich übereinstimmt. Dies führt dazu, dass das Bild verstärkt wird, also die vorhandenen Bereiche vergrößert (und zwar um die Form des Kernels). Die Abbildung 24 zeigt das

Resultat.

Der Operator "erode" führt das Gegenteil durch und verkleinert die vorhandenen Bereiche. Hier wird im Zielbild nur der Pixel gesetzt, wo der Kernel komplett auf den Bildausschnitt passt. (Bild 25).

Als "close" bezeichnet man dann die Hintereinanderausführung von Erode und Dilate, d.h. wir wenden auf das Ergebnis von "dilate" "erode" an. (Bild 26) Bei "open" wird erst "erode" und dann "dilate" ausgeführt. Dies ist zu sehen in Abbildung 27.

Die Resultate sehen dann wie folgt aus (jeweils mit einem 3×3 -Kernel in Kreisform):



Abbildung 24: Dilation

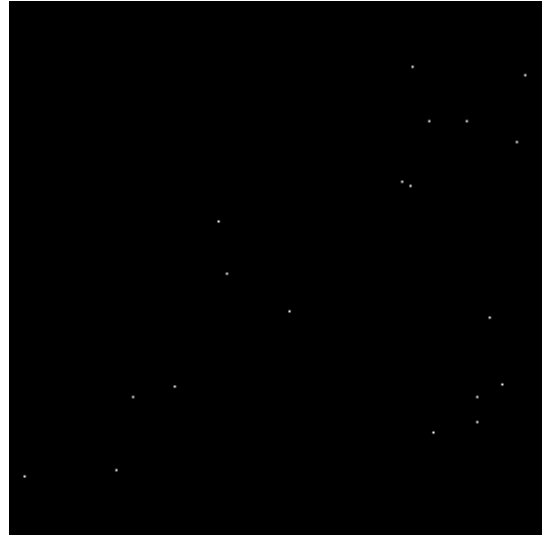


Abbildung 25: Eroding



Abbildung 26: Closing



Abbildung 27: Opening

Das man beim "open" und beim "erode" nichts sieht liegt daran, dass man keine Stelle findet, wo Kernel und Bild passen.