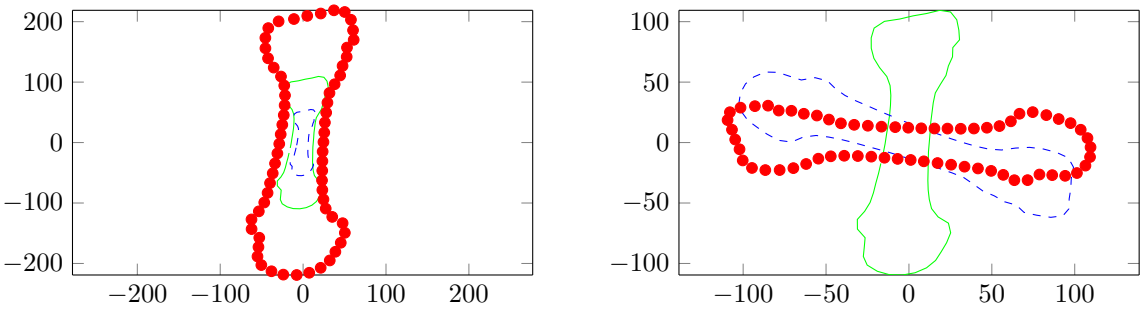
Gruppennummer 16

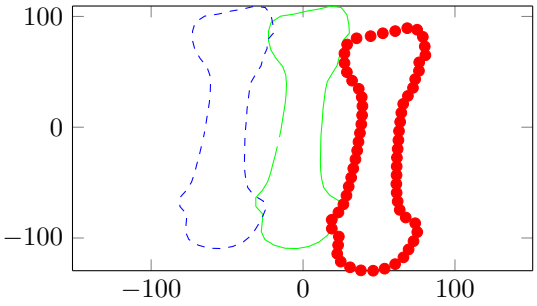
Andreas Cremer (0926918)
Hanna Huber (0925230)
Lena Trautmann (1526567)

June 24, 2016

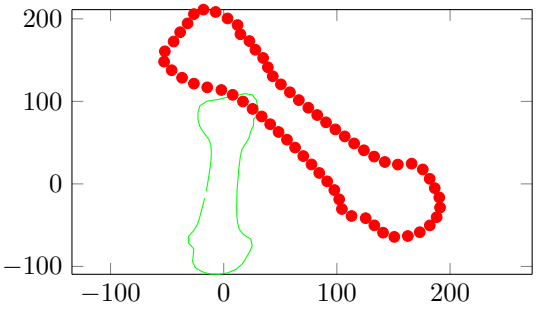


(a) $s = 2$ (rot) und $s = 0.5$ (blau)

(b) $r = 90°$ (rot) und $r = 250°$ (blau)



(c) $t = (50, -20)$ (rot) und $t = (-50, 0)$ (blau)



(d) $s = 1.5$, $r = 45°$ und $t = (70, 70)$ (rot)

Figure 1: Vergleich zwischen transformierten Shapes mit Vergrößerungsfaktor s, Drehwinkel r und Translationsvektor $t = (t_x, t_y)$ und Original-Shape (grün, $s = 1$, $r = 0$, $t = (0, 0)$) : Skalierung (1a), Rotation (1b), Translation (1c) und beliebige Transformation (1d).

1. Shape-Modell

   Abbildung 1 zeigt von generateShape.m generierte Shapes und vergleicht die Original-Shape mit verschiedenen Transformationen.

2. Featureberechnung

   In Abbildung 2 sind die für Bild 1 des Datensatzes *handdata* berechneten Features graphisch dargestellt. In 2a sind die unveränderten Grauwerte zu sehen. 2b und 2c stellen die daraus hergeleiteten Gradienten in x- und y-Richtung dar. Mit Hilfe dieser beiden Werte lässt sich die Stärke des Gradienten 2d berechnen.
   Von den Haar-like wird nur das erste Feature, berechnet mit den Grauwerten 2e und mit der Gradientenstärke 2f, dargestellt. Ein Pixel dieses Features ergibt sich folgendermaßen: Um das entsprechende Pixel im Originalbild wird ein Rechteck gelegt. Die Werte in der linken Hälfte des Rechtecks werden aufaddiert und davon die aufaddierten Werte der rechten Hälfte abgezogen. Der daraus resultierende Wert ist der neue Pixelwert. Den Unterschied zwischen der Berechnung aus den Grauwerten und der Berechnung aus der Gradientenstärke kann man am mittleren Knochen auf Höhe 200 erkennen. Im Grauwertbild hat der gesamte Knochen im Vergleich zur Umgebung erhöhte und halbwegs Werte, wodurch der Haar-like-Wert am linken Rand des Knochens hoch ist und sonst nirgends. Die Gradientenstärke ist an beiden Knochenrändern hoch und in der Mitte niedrig, wodurch der entsprechende Haar-like-Wert sowohl am linken als auch am rechten Knochenrand steigt und dann absinkt.
   Die letzten beiden Plots stellen jeweils die x- und die y-Koordinate des jeweiligen Pixels dar und sind damit bis auf die Bildgröße nicht vom jeweiligen Bild beeinflusst.

3. Klassifikation und Feature-Selection

   (b) Abbildung 3 zeigt den Klassifikationsfehler für Random-Forests mit unterschiedlich vielen Bäumen. Generell erhöht sich mit einer größeren Anzahl an Bäumen auch die Genauigkeit der Klassifikation. Jedoch mit immer geringer werdendem Unterschied. Ab 40 Bäumen ist keine wesentliche Verbesserung mehr erkennbar, wenn die Anzahl der Bäume weiter erhöht wird.

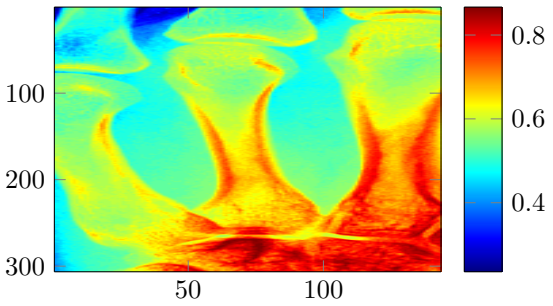
   (c) Abbildung 4 zeigt den Einfluss der einzelnen Features auf den Klassifikationsfehler für verschiedene Random-Forests. Dieser variiert bei manchen Features - z.B. manchen Haar-Features - stark mit der Anzahl an Bäumen. Features mit insgesamt stärkstem Einfluss sind die

Stärke des Gradienten (gradMag) und die Pixelkoordinaten (x,y), wobei die y-Koordinate stärkeren Einfluss hat als die x-Koordinate. Keinen Einfluss haben das 18.-20. Haar-Feature (grayHaar18-20, gradHaar18-20).

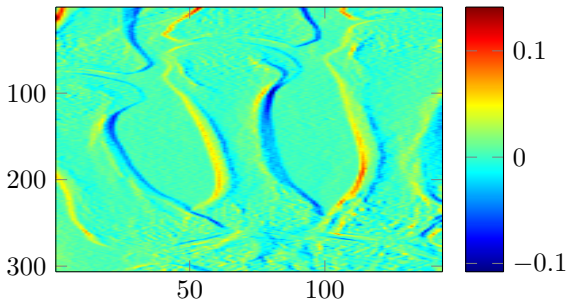
4. Shape Particle Filter

(d) Aufgrund der Ergebnisse von Aufgabe 3b und dem Beispielaufruf aus der Angabe, wurde die Anzahl der Bäume pro Random Forest auf 32 festgelegt. Würden wir mehr Bäume verwenden, könnte die Segmentiergenauigkeit der Methode verbessert werden, da die Vorhersage der Masks verbessert würde. Um die Rechenzeit gering zu halten, haben wir davon abgesehen, eine größere Zahl an Bäumen zu verwenden.
Bei der Optimierungsfunktion musste der Parameterbereich der Optimierung mit Minimum und Maximum beschränkt werden. Wird der Bereich ungeschickt gewählt, so kann es passieren, dass die Optimierungsfunktion nur ein lokales Minimum findet. Wird zBsp der Bereich für die Rotation von minimal 0° bis maximal 359° festgelegt, werden häufig Rotationen von 180° als Optimum gefunden (siehe Abbildung 5a). Wird der Bereich von $-180°$ bis $+180°$ definiert, ist dies nicht mehr der Fall.
Außerdem kann bei der Rotation angenommen werden, dass die Knochen nicht um 90° gedreht im Bild liegen. Beschränkt man die Rotation auf Werte von bis zu 50° in beide Richtungen, kommt es nicht zu Fehloptimierungen wie in Abbildung 5b.
Zusätzlich muss auch der Bereich für die Skalierung gut angepasst sein. Wird ein zu großer Bereich zugelassen, können zu kleine Knochen, siehe Abbildung 5c, als Optimum bestimmt werden.
Beim Shape-Modell wurden zwei verschiedene Varianten getestet. Eines mit neun Eigenvektoren, das 99,02% der Gesamtvarianz abdeckt, und eines mit vier Eigenvektoren, das 96,34% der Gesamtvarianz abdeckt. Der Unterschied im Fehler war hierbei zwar relativ gering, aber das gröbere Modell hatte einen geringeren Fehler, siehe Abbildung 6 (3)&(4). Dies kann daher kommen, dass für das genauere Modell die Shape exakter an die fehlerbehaftete Klassifizierung angepasst wird und daher einen größerer Fehler zur originalen Mask entsteht.
Als Kostenfunktion haben wir die Summe der euklidischen Distanzen aller Landmarks zu ihrem nächstgelegenen Punkt auf der Maske verwendet. Da der nächstgelegene Punkt mittels knnSearch ermittelt wurde, betrug die Rechenzeiten der Optimierung pro Bild zwischen 30 und 75 Sekunden. Im Vergleich dazu wurde sowohl die quadrierte Summe der Abweichungen verwendet, als auch eine einfache Kostenfunktion, bei der für jeden Landmark überprüft wird, ob er direkt auf einem Konturpixel liegt oder nicht. Die Berechnung der Kostenfunktion war zwar deutlich schneller (ungefähr 1 sek pro Bild), jedoch war die Bedingung zu strikt und das Ergebnis daher nicht zu verwenden.

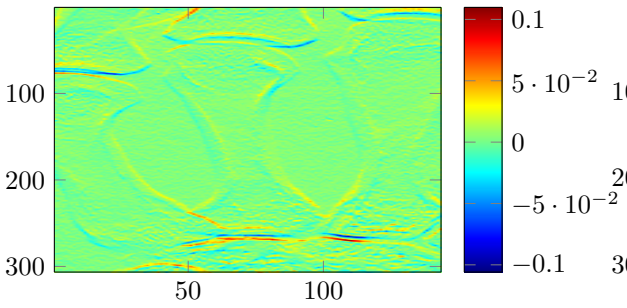
Generell können wir für unser Modell mit vier Eigenvektoren und geeigneter Einschränkung des Optimierungsbereichs ganz gute Ergebnisse erzielen (Beispiele siehe Abbildung). Der Median der Fehlerdistanz liegt zwischen 0.75 bis 1.2 Pixel (siehe Abbildung 6). Die besten Ergebnisse wurden für einen Optimierungsbereich zwischen ± 3 Standardvarianzen für die Shape-Parameter, eine Rotation von $\pm 50°$ und eine Skalierung zwischen 0.7 und 1.2 erreicht (Abbildung 6 (3)). Für festgesetzte Werte der Shape-Modell-Parameter zwischen ± 3 erhöhte sich der Median der Fehlerdistanz auf 1 Pixel (Abbildung 6 (2)). Wurden die Parameter weiter halbiert, stieg der Fehler, wie erwartet nochmal an (Abbildung 6 (1)). Ebenso wurden auch die Ausreißer deutlich größer.
Somit wurde zwar für den willkürlich auf 3 festgelegten Parameterbereich der Fehler im Median erhöht, die Rechenzeit war dagegen geringer (im Schnitt 37s anstatt 55s). Sowohl die weitere Halbierung des Optimierungsbereichs, als auch die Änderung der Anzahl an Eigenvektoren brachte keinen Zeitgewinn.
Zuletzt wurde noch der Einfluss der Kostenfunktion mit quadrierter Summer der Abweichungen untersucht (Abbildung 6 (5)). Dabei stieg der Fehler erneut und es kamen deutlich größere Ausreißer dazu, da bei manchen Bildern, z.Bsp. Bild 48, die Klassifikation sehr schlecht war (Abbildung 8).



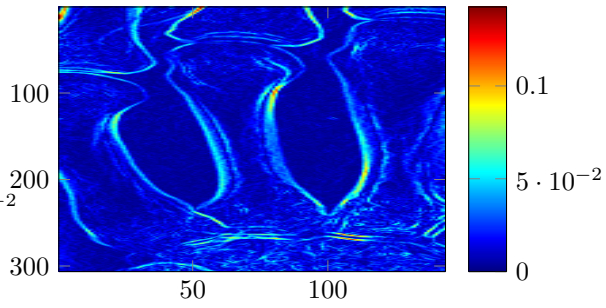
(a) Grauwert



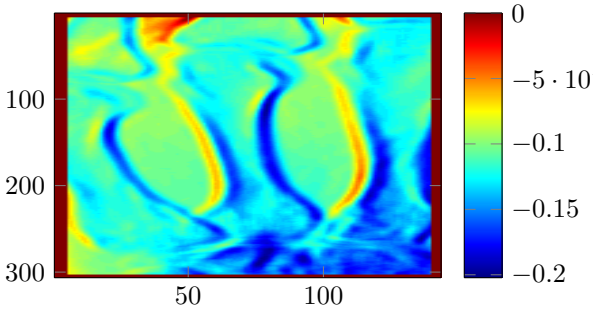
(b) Gradient in X-Richtung



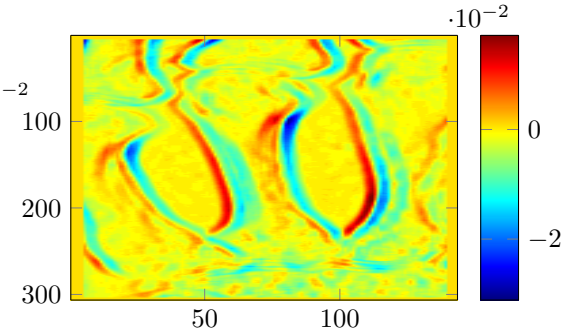
(c) Gradient in Y-Richtung



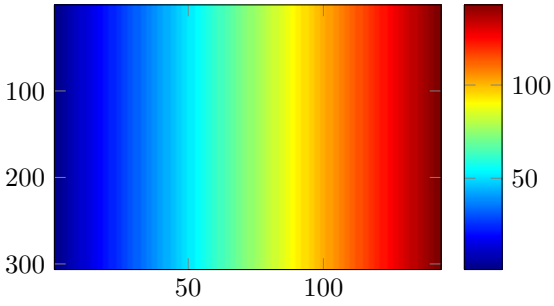
(d) Stärke des Gradienten



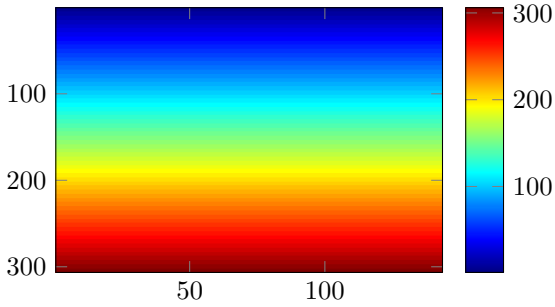
(e) Haar-like Features, berechnet auf dem Grauwertbild



(f) Haar-like Features, berechnet auf der Gradientenstärke



(g) x-Koordinate des Pixels



(h) y-Koordinate des Pixels

Figure 2: Ausgewählte Features für Bild 1 des Datensatzes *handdata*

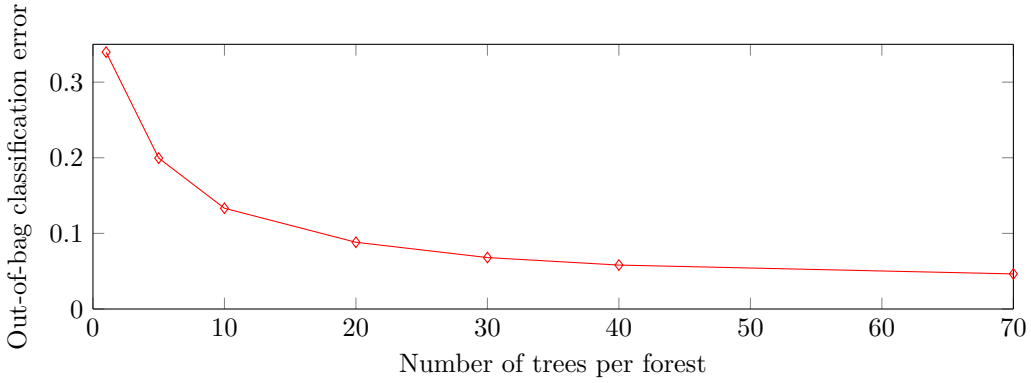


Figure 3: Klassifikationsfehler in Abhängigkeit von der Anzahl an Bäumen in einem Random-Forest

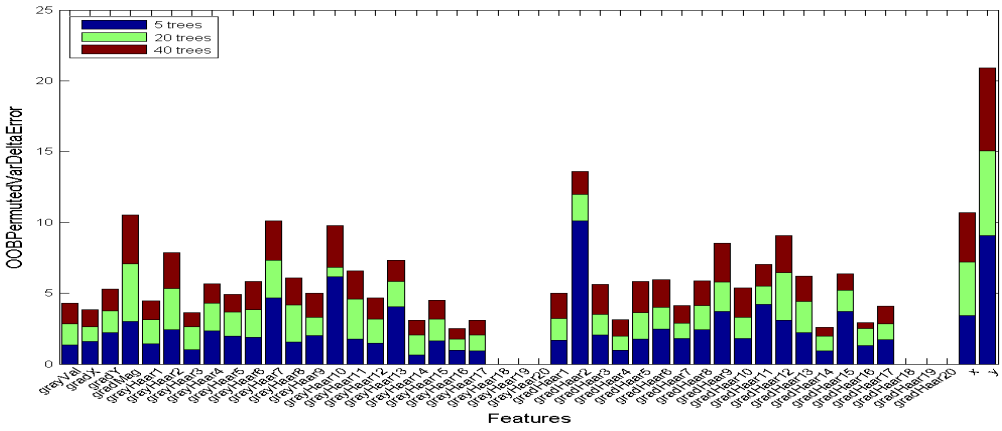
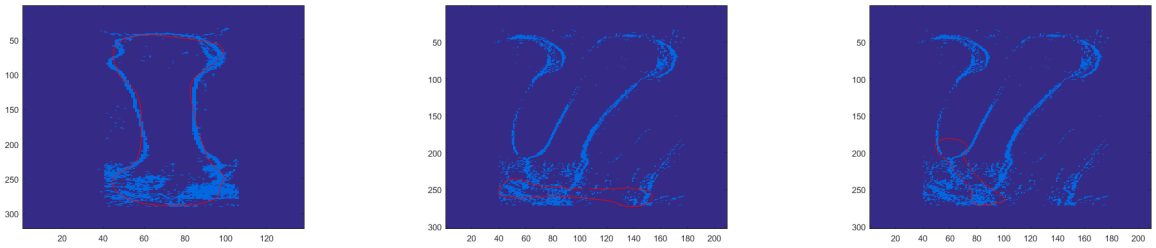


Figure 4: Einfluss der einzelnen Features auf den Klassifikationsfehler

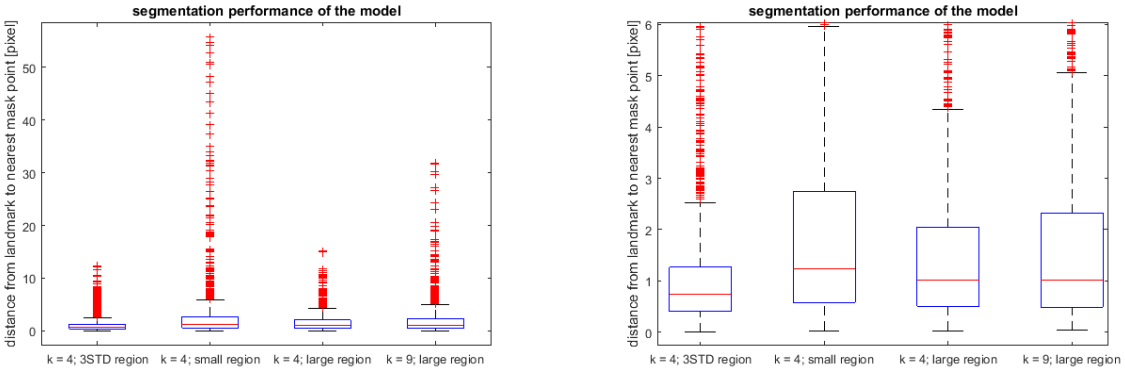


(a) Als Optimum gefundenes Shape Modell für eine falsche Rotationsbegrenzung von 0° bis 359°.

(b) Als Optimum gefundenes Shape Modell für eine falsche Rotationsbegrenzung von 0° bis 359° und einen zu großen Skalierungsbereich von 0,5 bis 2.

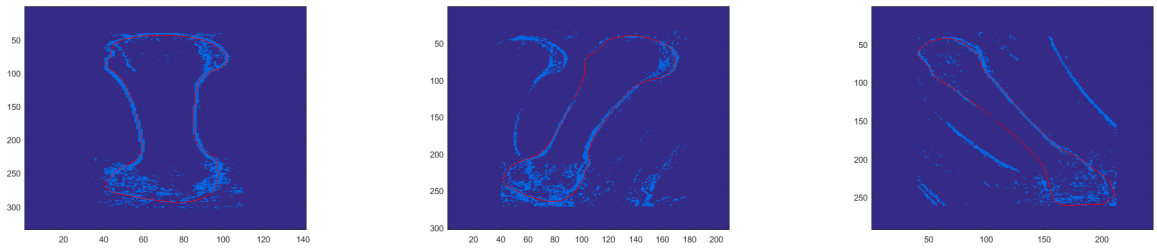
(c) Als optimal gefundenes Shape Modell für einen zu großen Skalierungsbereich von 0,5 bis 2.

Figure 5: Fehler im Optimum des Shape Modells, die bei einem schlecht gewählten Optimierungsbereich auftreten können.



(a) overview (b) zoomed view

Figure 6: Box-plot der Fehler zwischen Shape-Modell und Klassifikatorergebnis. (1): Das Modell deckt 95% der Varianz ab und für die Optimierung wurde ein kleinerer Bereich verwendet. (2): Das Modell deckt 95% der Varianz ab und für die Optimierung wurde ein größerer Bereich verwendet. (3): Das Modell deckt 95% der Varianz ab und für die Optimierung wurde für das Shape-Modell ein Bereich von ±3 Standardvarianzen verwendet. (4): wie (3), aber das Modell deckt 99% der Varianz ab. (5): wie (3), aber für die Kostenfunktion wurde die quadrierte Summe der Abweichung verwendet.



(a) Bild 34 (b) Bild 38 (c) Bild 48

Figure 7: Optimierungen bei vier Eigenvektoren für das Shape-Modell, der Summe der Abweichungen als Kostenfunktion und einem Optimierungsbereich von ±3 Standardabweichungen für die Shape-Modell-Parameter (Werte von r,s, tx und ty sind im Text gegeben).

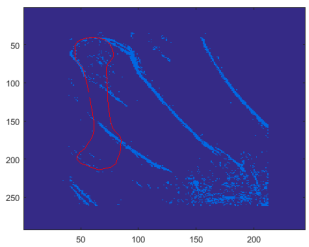


Figure 8: Fehler im Optimum des Shape Modells (Bild 48), der bei quadrierten Distanzen als Kostenfunktion und schlechter Klassifizierung auftrat (vgl mit Abbildung 7c).