

# **Exposee**

## **Automatisches Schätzen der Haaranzahl in ausgefallenen Haarbüscheln**

Janelle Pfeifer

Delpstraße 28

53359 Rheinbach

janelle.pfeifer@smail.inf.h-brs.de

Hochschule Bonn-Rhein-Sieg

Institute of Visual Computing

Fachbereich Informatik

Studiengang: Informatik (B.SC.)

Rheinbach, 15.02.2020

# Inhaltsverzeichnis

Exposee .....	1
1 Einleitung .....	3
2 Grundlagen .....	3
2.1 Computervision .....	3
2.2 matchTemplate .....	4
2.3 Canny Edge Detection .....	4
2.4 Dilatation .....	4
2.5 Inversion eines Graustufenbildes .....	4
2.6 watershed Algorithmus .....	5
2.7 Connected Components .....	5
2.8 Skelettierung .....	5
2.9 outlier removal ? .....	5
2.10 fitted curve, lineare regression, spline .....	5
3 Design und Implementation .....	5
3.1 Verarbeitung der Haar-Bilder .....	6
3.2 Kalibrieren .....	13
3.3 Relation Daten zu Haar Menge .....	13
3.4 Schätzen eines Haar-Bildes .....	13
3.5 Tests .....	17
3.6 Test: Knielange, dunkelrote Haare .....	17
3.7 Test: Hüftlange, feine, blonde Haare .....	18
3.8 Test: kurze Hellbraune Haare .....	19
3.9 benutzung? .....	19
4 Fazit .....	19
A Schätzungs Modelle .....	21
B Anhang blondes Haar .....	21

**Zusammenfassung.** am ende hier ein abstract hin

## 1 Einleitung

Das Ausfallen von Haarsträhnen ist ein natürlicher Teil des Haarwachstum Zyklus. Durchschnittlich verliert der Mensch 50 bis 100 Haarsträhnen am Tag. Vermehrter Haarverlust kann ein Anzeichen von Krankheiten oder Mängelscheinungen sein. Daher ist es von Vorteil das früher darauf reagieren zu können.

Haarausfall ist sehr variabel und wird durch viele Faktoren beeinflusst. Er kann sich abhängig von dem Haartyp, der Haarlänge und der Haarpflege-Methode jeden Tag ändern und Muster aufweisen. Jeder Mensch hat ein eigenes Haarausfall-Muster. Eine langzeituntersuchung gibt die möglichkeit diese Muster zu erkennen und Abweichungen herauszustellen. [1, 2, 3]

In dem Paper „The Hair shedding visual scale: A quick tool to assess hair loss in Women“ wird eine Methode beschrieben, in der Frauen anhand von Bildern den Umfang ihres täglichen Haarausfalls bestimmen können. Dabei werden den Frauen Bilder von abgezählten, ausgefallenen Haarbüscheln gezeigt, die ihrer eigenen Haarlänge entsprechen. Die Frauen wählen das Foto aus, welches ihrem persönlichen täglichen Haarausfall entspricht. Es wurde eine Korrelation festgestellt zwischen Frauen, die klinisch bestätigt Haarverlust erfahren und den Bildern, die sie ausgewählt haben. Der Haarausfall wird visuell, anhand von Bildern festgestellt.[4]

In dieser Arbeit wird ein Verfahren vorgestellt, mit dem der tägliche Haarausfall mithilfe von Methoden der Bildverarbeitung und statistischer Auswertung geschätzt werden kann. So wird eine Langzeitüberwachung möglichst effizient durchführbar.

## 2 Grundlagen

### 2.1 Computervision

In dem Bereich der Bildverarbeitung gibt es viele Algorithmen, die es ermöglichen Objekte in einem Bild erkennbar zu machen und zu verarbeiten.

## **2.2 matchTemplate**

Template matching findet bereiche eines bildes, die einem gegebenen Vorbild (template) ähnlich sind.

Dazu wird ein Source Image und ein template Image angegeben. Um die gesuchten Bereiche zu finden, wird das Template Image über das Source Image Pixelweise darüber geschoben. An jeder stelle wird das template mit dem source Image verglichen und eine Ähnlichkeit zwischen den beiden berechnet. Als Ergebnis erhält man eine Matrix die für jeden Pixel des Bildes eine Wahrscheinlichkeit angibt, das sich dort das template befindet.

## **2.3 Canny Edge Detection**

Canny Edge Detection ist ein Algorithmus der Kanten in einem Bild erkennt. Es handelt sich um einen Hochpassfilter. Hochpassfilter erhalten hochfrequente Bildanteile, während Niederfrequente verschwinden oder abgeschwächt werden. Hochfrequent sind Bereich mit schnellen Farb- und Helligkeitsveränderungen, was bei Linien und Kanten der Fall ist.

## **2.4 Dilatation**

Die Dilatation ist eine Morphologische Operation. Morphologische Operationen verändern ein Bild basierend auf Formen. Sie wenden ein Skturktu element

Die Dilatation arbeitet auf binären Bildern. Es gibt nur schwarz und weiß. Die Dilatation fügt weiße Pixel an die Ränder von weißen Objekten hinzu. Bei der Dilatation handelt es sich um ein Morphologischen Operator. Diese arbieten das Bild Pixelweise durch. Der Wert jedes Pixels basiert auf einem vergleich zischen dem Pixel und dessen Nachbar-Pixeln.

## **2.5 Inversion eines Graustufenbildes**

schwarz wind weiß und weiß wird schwarz. und alles daziwschen entsprechend. Helle Graustufen werden dunekl und umgekehrt. Die Graustufen eines bildes werden in 8 bit dargestellt. Somit gibt es

255 graustufen. 0 ist schwarz und 255 ist weiß. Somit gilt bei der inversion für jedes Pixel mit dem graustufenwert  $g$ :  $g_{\text{neu}} = 255 - g_{\text{alt}}$

## 2.6 watershed Algorithmus

**Region Growth** ist ein Verfahren, bei dem ein Bild in Regionen unterteilt wird. So werden zum Beispiel Sektionen mit ähnlicher Farbe herausgestellt.

## 2.7 Connected Components

## 2.8 Skelettierung

reduziert Objekte in einem Bild auf eine Pixelbreite. So können Linien auf eine Pixelbreite geschrumpft werden.

## 2.9 outlier removal ?

## 2.10 fitted curve, lineare regression, spline

# 3 Design und Implementation

Das Ziel ist es die Einschätzung eines Haarbüschels, mithilfe von Computervision, zu automatisieren. Zunächst wird ein Kalibriervorgang durchgeführt, in dem der Nutzer abgezählte Haarbüschelbilder eingibt. Dann kann die Menge der Haare in nicht abgezählten Haarbüscheln automatisch geschätzt werden.

Die Schätzungen werden zugehörig zum Datum gespeichert. Um einen Zeitlichen verlauf festzuhalten. Zusätzlich zu den Schätzungen kann für jedes Datum eine Marker angegeben werden. So können regelmäßige Wiederholungen der Haarpflege, wie zum Beispiel eine Haarwäsche, notiert werden. Des Weiteren können gesundheitliche Veränderungen oder Einnahme von Medikamenten vermerkt werden.

Das Programm wurde in Python implementiert und nutzt unter anderem das Plugin Open-cv. Der Nutzer steuert das Programm über die Kommandozeile.

Da die Haare jeder Person unterschiedlich sind, muss für jeden Nutzer eine eigene Kalibrierung durchgeführt werden. Um Vermischung

von Daten unterschiedlicher Personen und häufige Neukalibrierungen zu vermeiden, kann das Programm beliebig viele Nutzer anlegen und verwalten.

### 3.1 Verarbeitung der Haar-Bilder

Sowohl für die Kalibrierung als auch für die Schätzung von Haar-Mengen müssen die eingegebenen Bilder zunächst verarbeitet werden und Daten, mit denen auf die Haar-Menge geschlossen werden kann herausgestellt werden. Dazu werden Methoden der Bildverarbeitung verwendet.

Die Methode detect implementiert die Verarbeitung eines Bildes. Als Eingabe wird von einem Haarbüschel auf einem einfarbige Hintergrund ausgegangen. Dabei sollte ein möglichst guter Kontrast zu der Haarfarbe gewählt worden sein. Beispielsweise ein weißer Hintergrund für dunkle haare oder ein schwarzer Hintergrund für blondes oder hellbraunes Haar. Die Ecken des Hintergrundes sind durch ein Symbol markiert. Vergleich Abbildung 1.

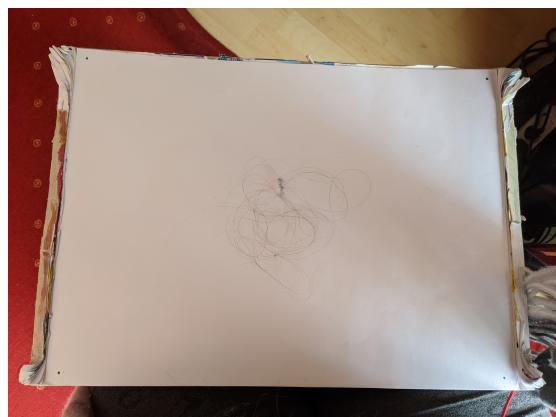


Abb. 1: Input

Im ersten Schritt der Bildverarbeitung wird das Bild auf die Markierungen in den Ecken zugeschnitten. Das ist in der Methode cropDots implementiert. Über einen Aufruf der Open cv methode matchTemplate wird nach vorkommen der Markierung gesucht. Das Resultat

ist ein Karte an Wahrscheinlichkeiten, die für jedes Pixel angibt, zu welcher Wahrscheinlichkeit sich dort die Markierung befindet. Die 4 Stellen an denen die Wahrscheinlichkeit am höchsten ist, werden als Eckpunkte angesehen. Siehe Abbildung 2.

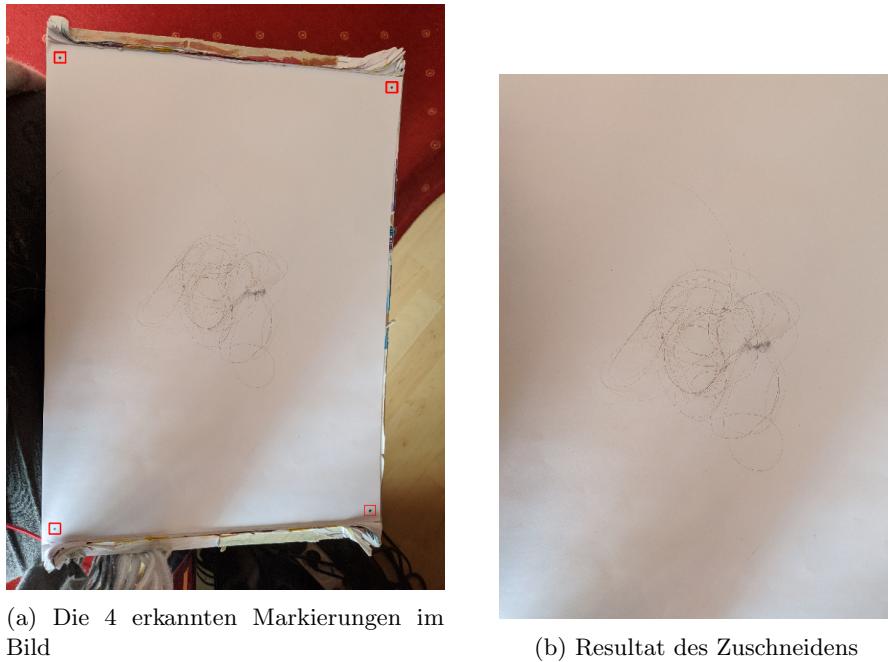
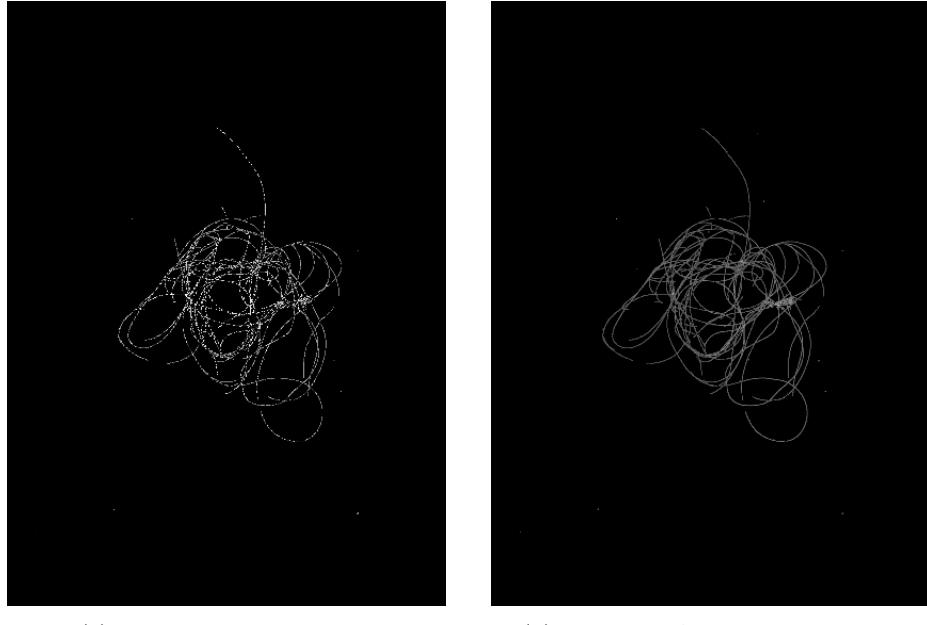


Abb. 2: Der Prozess des Zuschneidens auf Markierungen in den Ecken

Durch das Zuschneiden der Bilder wird sichergestellt, dass die Haare immer etwa einen gleichgroßen Hintergrund haben. So werden die Daten, die aus den Bildern gezogen werden, nicht verfälscht dadurch, dass die Haare eventuell von unterschiedlichen Abständen fotografiert wurden.

Nach dem Zuschneiden werden die Haare von dem Hintergrund getrennt. Das ist in der Methode edgeProcess implementiert. Mithilfe von Canny Edge detection werden die Haare grob erkannt. Weiße Pixel gelten als Haare, schwarze als Hintergrund. Siehe Abbildung 3a.



(a) Canny Kantendetektion

(b) Intensität-Abbildung der Haare

Abb. 3: Beispiel input von knielangen dunkelroten Haare

In der Methode hairPixelIntensity wird die Detektion verfeinert. Auf das Resultat der Katedetektion wird eine Dilatation ausgeführt. Dadurch gelten mehr Pixel als Haare und eventuelle Löcher wachsen zusammen.

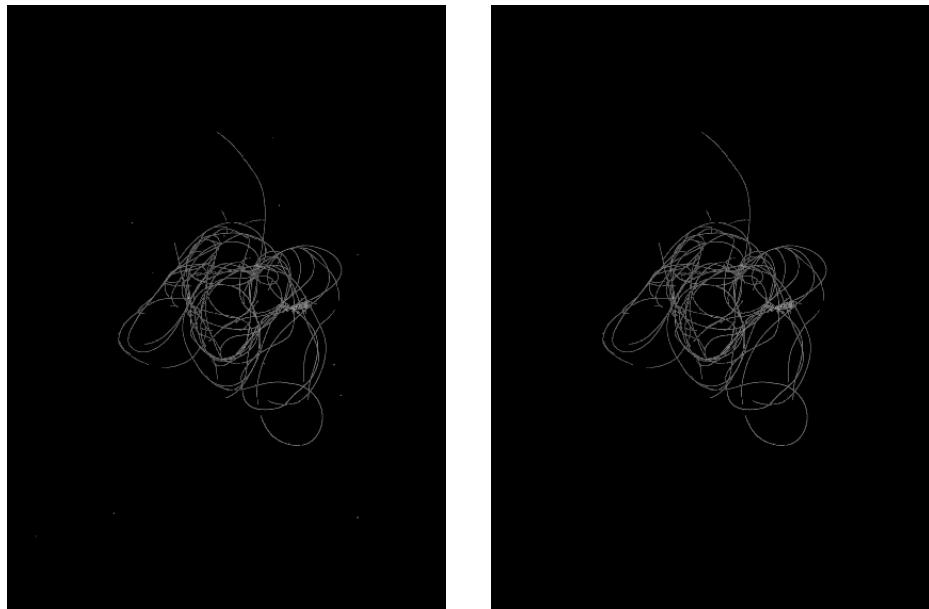
Da Haare halb-transparent sind, wird ihre Farbe intensiver, wenn sie sich überlagern. Sie weisen eine höhere Intensität auf. Um die Haar-Intensität des Bildes zu extrahieren, wird ein Bild gebaut, in dem Hintergrund Pixel weiß sind, und Haarpixel ihren Graustufenwert aus dem Original Bild erhalten. Siehe Abbildung ??.

Wenn die Haare heller sind als der Hintergrund, es sich beispielsweise um blondes Haar auf schwarzem Hintergrund handelt haben die Haare eine Höhere intensität, wenn sich heller sind. Bei dunklem Haar auf hellem Hintergrund ist es gegenteilig. Die Intensität ist höher, wenn das Haar dunkler ist. Damit beide Konstellationen zu einem ähnlichen Ergebnis kommen, werden die graustufenwerte invertiert, wenn die Haare hell sind. So ist auch bei ursprünglich hellem Haar, die Intensität höher, wenn die Graustufen dunkler sind.

Die Inversion dieses Bildes ergibt ein Bild mit schwarzem Hintergrund und Haaren, die Heller sind an Stellen wo sie sich Überlagern. Siehe Abbildung 4a.

Es werden kleine vereinzelte Gebiete des Hintergrundes in der Kanten-detection erkannt und gelten somit als Haare. Um diese Gebiete zu erkennen und zu entfernen, wird der Watershed Algorithmus angewendet. Dabei werden Marker in alle verbundenen Gebiete gesetzt, die nicht schwarz sind. Der Watershed Algorithmus lässt die Regionen wachsen. Dann werden die Regionen entfernt, die im Vergleich auf die Größe des Bildes sehr klein sind. Siehe Abbildung ??.

Die Summe aller Graustufenwerte werden als Intensität gespeichert. Dabei wird der Hintergrund außer acht gelassen, weil es schwarz ist, und somit den Graustufenwert 0 hat.



(a) Intensität-Abbildung der Haare

(b) Kleine Regionen Entfernt

Abb. 4: Entfernen von kleinen Regionen, vorher nachher

In der Methode hairPixelPercentage werden die Pixel gezählt die als Haare gelten. Zusätzlich wird der Prozentsatz an Haar Pixeln und die Intensität des Bildes im Vergleich zur Bildgröße gespeichert.

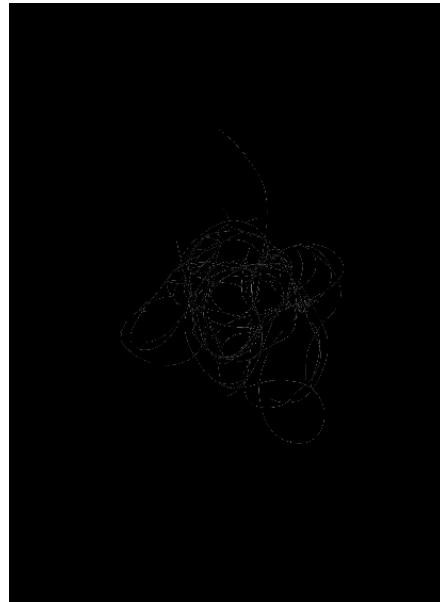
Dann wird die Dichte der Haare Untersucht. In der Methode Dafür werden die Hintergrund Regionen bestimmt, die durch Haare getrennt werden. Dafür wird der Watershed Algorithmus angewendet. Die Marker werden in alle zusammenhängenden Hintergrundgebiete gesetzt. Dabei wird die Größe der Region, die die Haare umschließt gespeichert. Abbildung 5b. Zusätzlich wird die Anzahl der Regionen die von Haaren umschlossen wird und deren Durchschnittliche Größe gespeichert.



Abb. 5: Anwendung von watershed

Als nächstes werden die Gebiete des Bildes untersucht, die eine erhöhte Haar-Dichte aufweisen. Dafür wird das Intesität-Bild ske-

lettiert. Siehe Abbildung 6b. Die Hintergrund Regionen werden wie zuvor untersucht.



(a) Skelett des Intensität-Bildes



(b) Äußerte Region des skelettierten Bildes

Abb. 6: Dichtere Regionen untersuchen

In der Methode `denseAndLoosePerc` wird die Haar-Intensität in den dichteren Regionen und in den weniger dichten Regionen berechnet und abgespeichert.

Für das Bild werden 28 Datenpunkte extrahiert und abgespeichert:

- `intensitySum` - Summe der Graustufenwerte des Intensität-Bildes
- `intensityShare` - Intensität pro HaarPixel
- `hairpixels` - Anzahl der Pixel, die Haare darstellen
- `imagepixels` - Gesamtanzahl der Pixel im Bild
- `hairpixel percentage` - Prozentsatz der Haarpixel im Bild

- number of sections - Anzahl an Hintergrund Regionen, die Watershed findet.
- number of background sections - Anzahl an Regionen, die den Hintergrund darstelle. (2 weniger als number of section. Watershed zählt Haare und Übergänge zwischen Regionen mit)
- outerSectionSum - Summe an Pixeln, die sich in der Region befinden, die die Haare umschließt
- outerSectionPercentage - outerSectionSum/imagepixels
- innerSectionSum - Summe an Pixeln, die sich in Regionen befinden, die von Haaren umschlossen werden
- innerSectionAvgSize - Durchschnittliche Pixelanzahl, die sich in den umschlossenen Regionen befinden
- innerSectionAvgSize Percentage - innerSectionAvgSize/imagepixels
- innerSectionSizeVariance - statistische Varianz der Pixelmengen, die sich in umschlossenen Regionen befinden
- innerSection standard deviation - Standardabweichung der Pixelmengen, die sich in umschlossenen Regionen befinden
- number of sections only considering dense hair - Anzahl der Regionen, die Watershed findet nachdem der input Skelettiert wurde. Regionen werden von dichten Haaren unterteilt.
- number of background sections inclosed by dense hair - Anzahl an Regionen, die den Hintergrund zwischen dichten Haaren darstellen
- dense hair outerSectionSum - Summe an Pixeln, die die Regionen umschließen, die von dichten Haaren umschlossen werden
- dense outerSectionPercentage - dense hair outerSectionSum / imagepixels
- dense innerSectionSum - Summe an Pixeln, die sich in den Regionen befinden, die von dichten Haaren umschlossen werden

- dense hair innserSectionAvgSize - Durchschnittliche Anzahl der Pixel, die sich in Regionen befinden, die von dichten Haaren umschlossen werden
- dense hair innerSectionAvgSize Percentage -  $\text{dense hair innerSectionAvgSize} / \text{imagepixels}$
- dense innerSectionSizeVariance - statistische Varianz der Pixelmengen
- dense innerSection Size standard deviation - Standardabweichung der Pixelmengen
- denseHairSum - Summe an Pixeln, die als Dichte Haare angesehen werden
- looseHairSum - Summe an Pixeln, die als Haare angesehen werden, jedoch nicht als Dicht gelten
- IntensitySum in Loose Section - Summe der Graustufenwerte in nicht-dichten Regionen
- IntensitySum in Dense Section - Summe der Graustufenwerte in dichten Regionen

### **3.2 Kalibrieren**

Die Kalibrierung ist in der Methode calibration implementiert. Für die Kalibrierung werden mehrere Bilder als Input angegeben. Für jedes der Bilder wird die Menge der Haare in dem Titel des Bildes angegeben. Die Bilder werden alle mit detect untersucht. Die Daten aus den Untersuchungen, sowie die Haar-Mengen werden abgespeichert. ganzen Ordner kalibrieren. einzelne Bilder hinzufügen.

### **3.3 Relation Daten zu Haar Menge**

### **3.4 Schätzen eines Haar-Bildes**

Nachdem eine Kalibrierung durchgeführt wurde. Kann die Haar-Menge auf beliebigen Bildern geschätzt werden. Das Schätzen ist in der Methode readFilesAndGuess implementiert.

Zuerst wird das zu schätzende Bild mithilfe von detect Untersucht. Dann werden die Daten aus der Kalibrierung geladen. Die Daten werden teilweise miteinander verrechnet und dann auf die dazugehörigen Haar-Mengen abgebildet. So entstehen eine Menge an Graphen, bei denen die Haar-Menge jeweils auf der y Achse liegt. Siehe Abbildung 7.

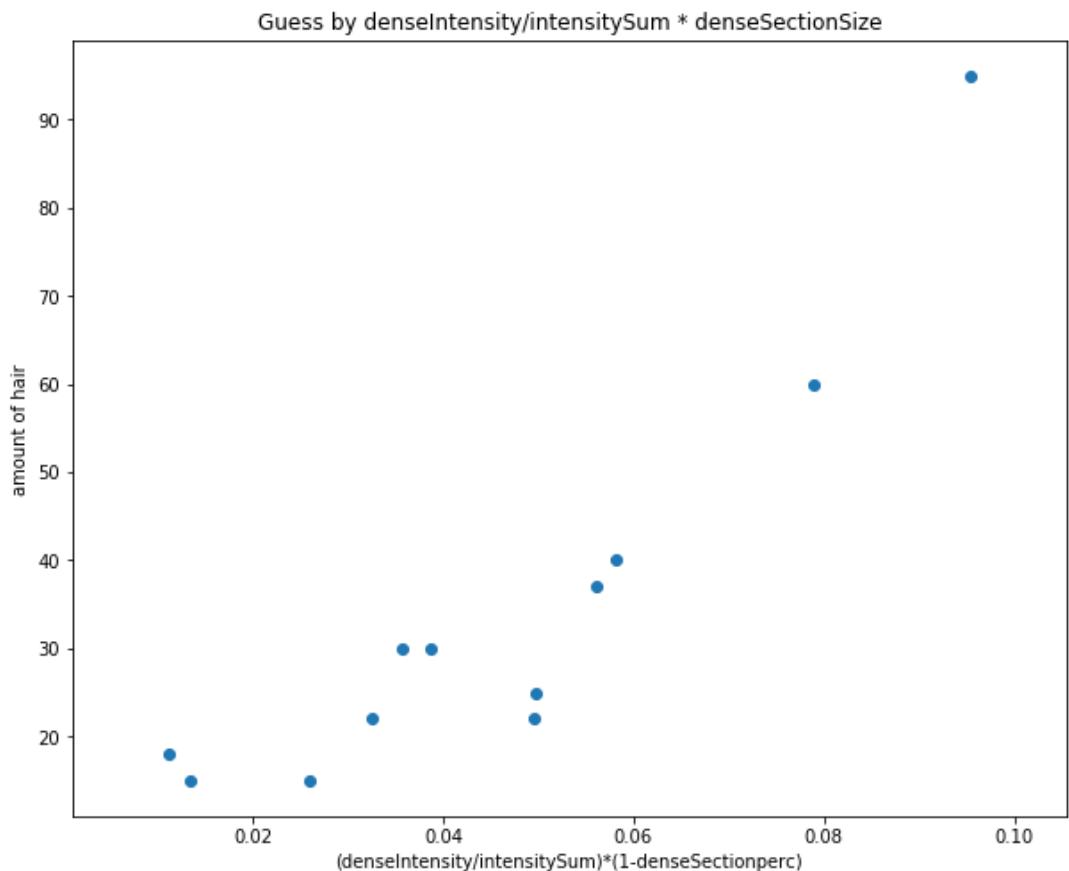


Abb. 7: Haar-Menge in Relation zur Intensität in dichten Regionen normalisiert durch gesamte Intensität und Region-Größe

Die Punkte in den Graphen werden durch Funktionen Approximiert. Die Funktionen können dann genutzt werden um Haar-Mengen zu schätzen. Für jeden Graph werden 3 Funktionen berechnet. Eine

Lineare Regression, eine Spline-Interpolation, und eine benutzerdefinierte Funktion. Der Benutzer kann aus linear, logarithmisch und exponentiell aussuchen. Siehe Abbildung 8.

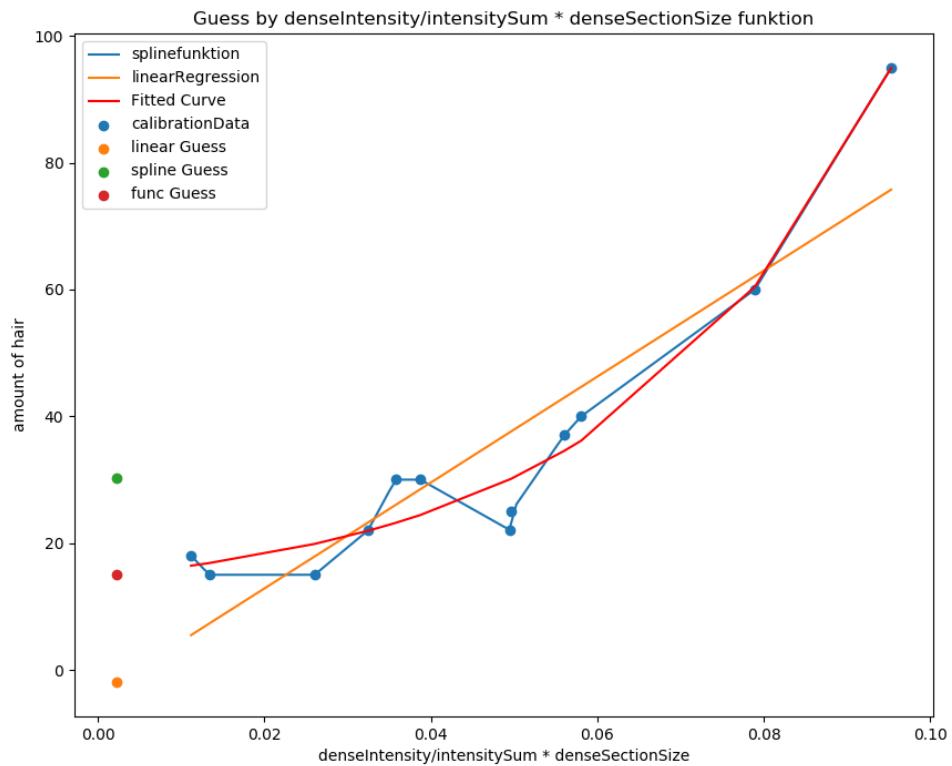


Abb. 8: Funktionen, die Datenpunkte Approximieren. Spline-Interpolation, lineare Regression und exponentielle Funktion.

Von den 28 Datenpunkten die aus den Bildern ausgewertet werden, werden 17 Genutzt um 12 Modelle zu erstellt. Diese Modelle wurden als aussagekräftig gefunden, um die Haarmenge schätzen zu können. Die folgenden x Werte werden verwendet:

- hairpercent

- Prozentsatz der Haarpixel im Bild
  - $(\text{hairpixels} / \text{hairSectionSize}) * (1 - \text{outerSectionPerc})$  HaarPixel dichte in der Region, in der sich Haare Befinden. Gibt dasselbe Ergebnis wie hairpercent.
  - $(\text{hairpixels}/\text{hairSectionSize}) * (1-\text{outerSectionPerc}) * \text{hairPerc}$  Dichte
  - $((\text{denseHairSum} / \text{denseInnerSectionSize}) * (1 - \text{denseSectionperc})) = \text{denseDensity}$
  - $\text{denseDensity} / \text{looseDensity}$
  - $\text{backgroundSectionNum} * (1-\text{outerSectionPerc})$
  - $\text{denseSectionNum} * (1-\text{outerSectionPerc})$
  - $\text{intensitySum} * \text{hairpixels} * (1-\text{outerSectionPerc})$
  - $\text{denseSectionAVGSize} / (1-\text{denseSectionperc})$
  - $(\text{denseSectionAVGSize}/\text{looseSectionAVGSize}) / (1-\text{outerSectionPerc})$
  - $(\text{denseIntensity}/\text{intensitySum}) * (1-\text{denseSectionperc})$
  - $(\text{denseIntensity}/\text{denseHairSum}) * (1-\text{denseSectionperc})$

Siehe den Anhang für die graphische Darstellung der Modelle.

Für jedes der Modelle werden 3 Funktionen approximiert. Mit den Funktionen werden jeweils 3 Schätzungen für das zu schätzende Bild berechnet.

Ausreißer werden entfernt. Dazu werden zunächst alle Schätzungen von weniger als 0 Haaren entfernt. Zusätzlich werden statistische Ausreißer mithilfe eines Algorithmus basierend auf dem tschebyschef theorem entfernt.

Der Mittelwert der übrig gebliebenen Schätzungen ist das Endergebnis. Siehe Abbildung 10 für den Verlauf einer Schätzung auf der Kommandozeile.

```

D:\Eigene Dateien\Dokumente\GitHub\HairEstimation\HairEstimationPython>python Estimation.py guess Users/Mummel/estimationInput/IMG_20200406_153354.jpg -d -r
active user: Mummel

guessing the amount of hair in the picture
image path Users/Mummel/estimationInput/IMG_20200406_153354.jpg
cropping image to TemplateDot using PatternMatching
Users/Mummel
detecting Hair via edge detection
backgroundcolor 77.54571671611444
removing smaller Regions...
done
processing background regions...
innerSectionNum 203
done
processing background regions...
innerSectionNum 69
done
Guess by hairpercent [ spl 17.10918697133671 ;func 16.068781620734256 ;linguess 8.8175636009952198 ]
Guess by density * hairsection size [ spl 17.109186971336708 ;func 16.068779751624007 ;linguess 8.8175636009952205 ]
Guess by density per sectionsize per hairperc [ spl 17.024852282387712 ;func 16.08739465599372 ;linguess 12.660421865313491 ]
Guess by density of dense section in relation to section size [ spl -5.032616520694617 ;func 12.65011382719596 ;linguess -0.7966692059873983 ]
Guess by dense Density to looseDensity ratio [ spl 21.878453935854328 ;func 25.152241123554553 ;linguess -6.1125370025128184 ]
Guess by density by background sections peeking through [ spl 17.841441686572296 ;func 12.290643639856775 ;linguess 12.290643505525784 ]
Guess by dense Density by background sections peeking through [ spl 14.153844827316952 ;func 10.218164223287738 ;linguess 10.21816431054834 ]
Guess by intensity in relation to hairPixels and hairsection size [ spl 16.646633135931893 ;func 18.824654298332096 ]
Guess by average background section sizes in dense section [ spl -19.15505299724856 ;func -1.5712803441519831 ]
Guess by dense average background section sizes compared to looseaveragebackground sections size [ spl 23.017721516155298 ;func 26.624021893042254 ]
Guess by denseIntensity/intensitySum * denseSectionSize [ spl 30.189213701279854 ;func 15.04653160017627 ;linguess -1.9621895800223128 ]
Guess by denseIntensity/denseHairSum* denseSectionSize [ spl 24.005185075441997 ;func 6.7281532831400686 ;linguess 6.728152779303436 ]
[8.82 17.11 16.07 8.82 17.11 16.07 12.66 17.02 16.09 12.65 21.88 25.15
12.29 17.84 12.29 10.22 14.15 10.22 16.65 18.82 23.02 26.62 30.19 15.05
6.73 24.01 6.73]
outliers Removed [17.11 16.07 17.11 16.07 12.66 17.02 16.09 12.65 21.88 12.29 17.84 12.29
10.22 14.15 10.22 16.65 18.82 15.05]
mean 15.232660728295835 res 15.0 median 16.06878068617913
Found duplicate date. Using handling mode: r
replacing old datapoint with 2020-04-06 15.0 ImagePath: Users/Mummel/estimationInput/IMG_20200406_153354.jpg

```

Abb. 9: Kommandozeilenauftruf und Verlauf einer Schätzung

### 3.5 Tests

#### 3.6 Test: Knielange, dunkelrote Haare

Knielange dunkelroten Haare wurden auf einem weißen Hintergrund aufgenommen. Die Ecken wurden mit dunklen Punkten markiert. Siehe Abbildung 11.

es wurde eine Untersuchung in einem Zeitraum von 7 Wochen von dem tägliche Haarausfall angestellt.

Um die Fehlerrate zu testen zu können wurde für einige der eingage Bilder die Anzahl der Haare per Hand gezählt. Mit einer großen Menge an Kalibrations-Bildern konnten die folgenden Fehler in den Schätzungen vermerkt werden:

```
[22 23 67 23 16 28 19 17 14 13 20 27 26 89 25 13 23 19 14 13]
(estimated, actual) [(22, 22), (23, 25), (67, 65), (23, 22), (16, 18),
(28, 26), (19, 21), (17, 18), (14, 9), (13, 17), (20, 27), (27, 30), (26,
27), (89, 95), (25, 40), (13, 15), (23, 31), (19, 31), (14, 11), (13, 16)]
error per estimation [ 0 -2 2 1 -2 2 -2 -1 5 -4 -7 -3 -1 -6 -15 -2 -8 -12
3 -3] mean error 4.05
```

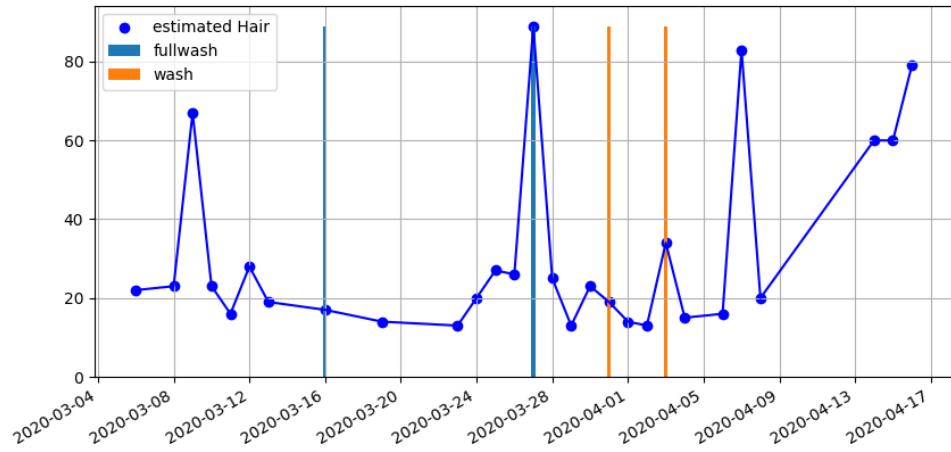


Abb. 10: Schätzungen aus einem Zeitraum von 7 Wochen

Mit nur 3 Kalibrationsbilern der Menge: 22, 60, 120 konnte diese Fehlerrate gefunden werden:

TODO

### 3.7 Test: Hüftlange, feine, blonde Haare

Blonde Haare wurden vor einem Schwarzen Hintergrund aufgenommen.

Für den schwarzen Hintergrund wurde besonders matte schwarze Pappe verwendet. Wenn die struktur der Pappe zu grob ist, oder durch einen glaz der pappe sichtbar wird, wir sie von der Canny Edge detection erkannt.

Auf schwarzem Hintergrund ist Staub besonders gut sichtbar. Um die Wahrscheinlichkeit zu verringern, das Verschmutzungen auf der Pappe für die Markierungen in der Ecke gehalten werden, wurden die Markierungen von einem einfachen Punkt zu einem kreuz geändert.

Mit Kalibrations-Bildern kann folgende Fehlerrate gefunden werden.



Abb. 11: Beispiel input von blonden langen Haaren

Siehe Anhang B für Schätzungs modelle und die Verarbeitung von Blondinen Haaren.

### 3.8 Test: kurze Hellbraune Haare

Für hellbraune haare hat es sich gezeigt, das ein schwarzen Hintergrund einen besseren Kontrast bietet.

Input beispiele.

Obwohl die Haare recht kurz sind, ist auch hier die Fehlerrate recht gering. Getestet wird an:

- Knielangen, dunkelroten Haaren
- Hüftlangen, feinen, blonden Haaren
- Schulterlangen, brauen Haaren
- Den Bildern aus dem Paper „The Hair shedding visual scale: A quick tool to assess hair loss in Women“

### 3.9 benutzung?

## 4 Fazit

Es wird eine praktische Methode gegeben eine Langzeitüberwachung von Haarausfall anzustellen.

So wird der Anwender sich bewusst, welcher tägliche Haarausfall normal ist und welche Menge besorgniserregend ist. Haarausfall ist sehr variabel und wird durch viele Faktoren beeinflusst.

Eine Langzeitüberwachung beruhigt den Nutzer bei saisonalem Haarausfall und gibt Hinweise auf die allgemeine Gesundheit der Haare.

Die Langzeitüberwachung macht den Haarausfall transparenter und berechenbarer. So wird der Nutzer früher aufmerksam auf anomalen Haarausfall und kann darauf besser reagieren. Haarverlust ist ein Indikator für viele Krankheiten und kann somit als Warnsystem dienen.

## A Schätzungs Modelle

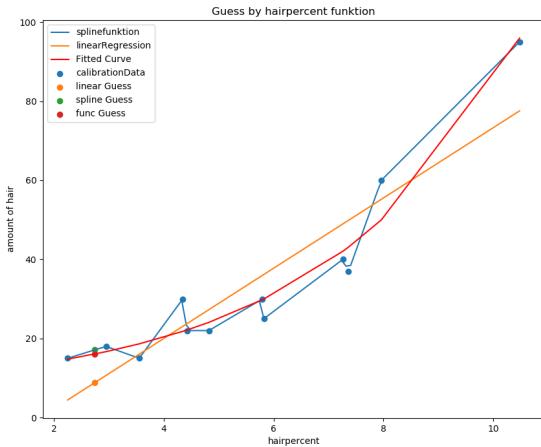


Abb. 12: Cover des Spiels

## B Anhang blondes Haar

### Literatur

- [1] Howard Maibach und Ian Yamaguchi. „Chemically Induced Hair Loss/Alopecia“. In: Jan. 2012, S. 1935–1942. ISBN: 978-3-642-02034-6. DOI: 10.1007/978-3-642-02035-3\_205.
- [2] Leonid Trost, Wilma Bergfeld und Ellen Calogeras. „The diagnosis and treatment of iron deficiency and its potential relationship to hair loss“. In: *Journal of the American Academy of Dermatology* 54 (Juni 2006), S. 824–44. DOI: 10.1016/j.jaad.2005.11.1104.
- [3] E.Y. Hsiang u. a. „Seasonality of hair loss: a time series analysis of Google Trends data 2004 to 2016“. In: *British Journal of Dermatology* 178 (Okt. 2017). DOI: 10.1111/bjd.16075.
- [4] María Martínez-Velasco u. a. „The Hair Shedding Visual Scale: A Quick Tool to Assess Hair Loss in Women“. In: *Dermatology and Therapy* 7 (Feb. 2017). DOI: 10.1007/s13555-017-0171-8.

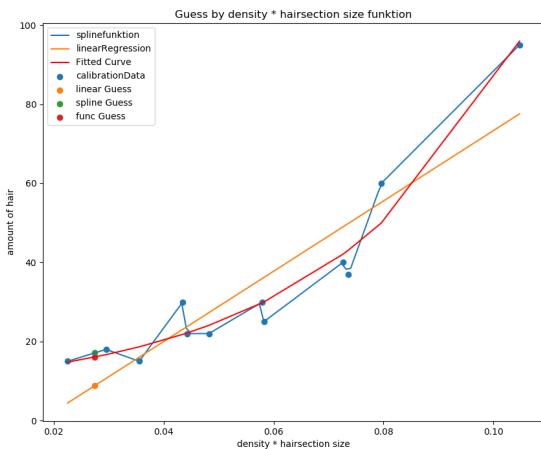


Abb. 13: Cover des Spiels

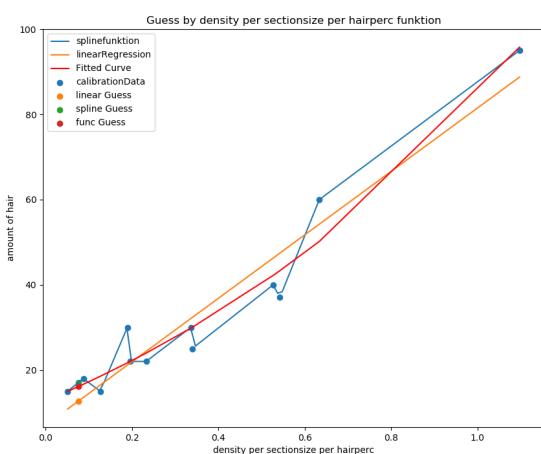


Abb. 14: Cover des Spiels

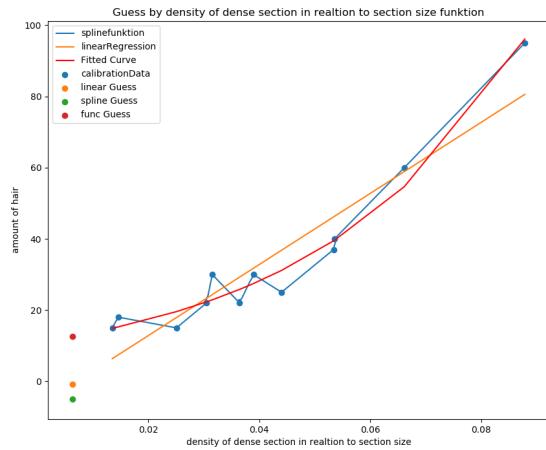


Abb. 15: Cover des Spiels

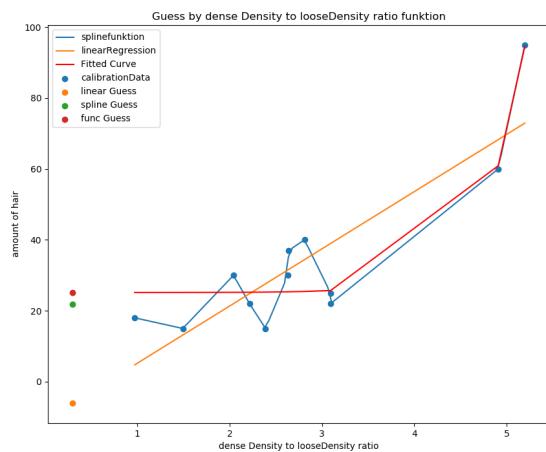


Abb. 16: Cover des Spiels

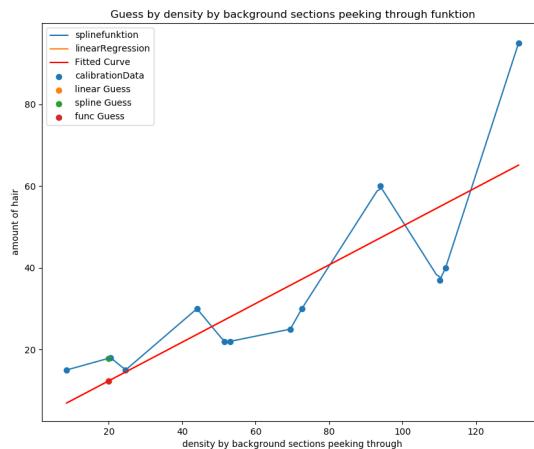


Abb. 17: Cover des Spiels

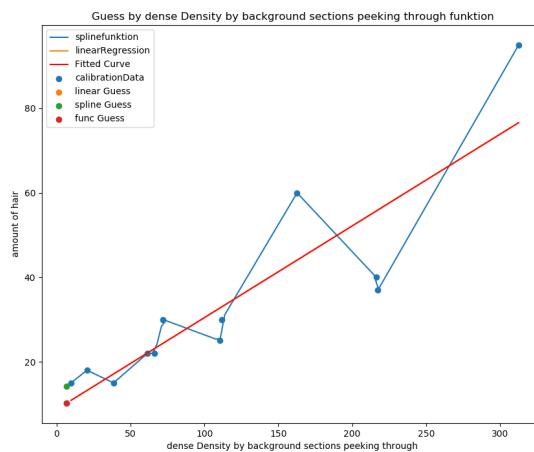


Abb. 18: Cover des Spiels

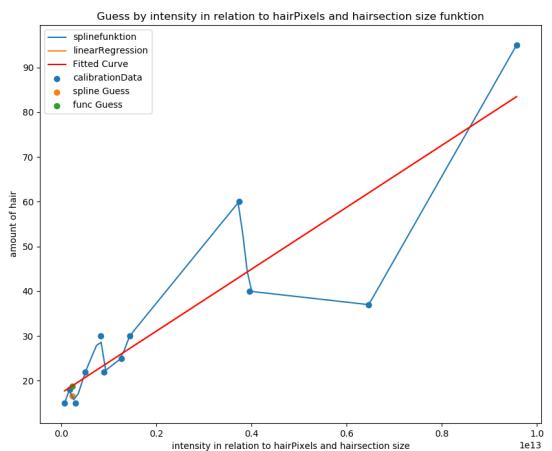


Abb. 19: Cover des Spiels

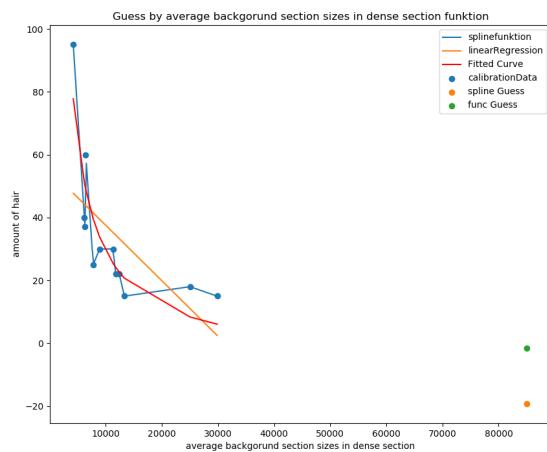


Abb. 20: Cover des Spiels

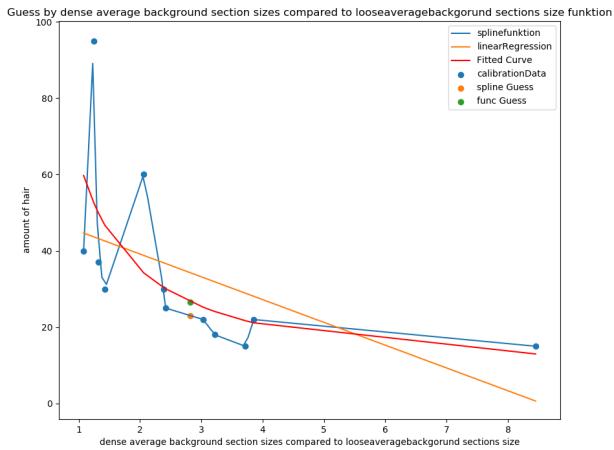


Abb. 21: Cover des Spiels

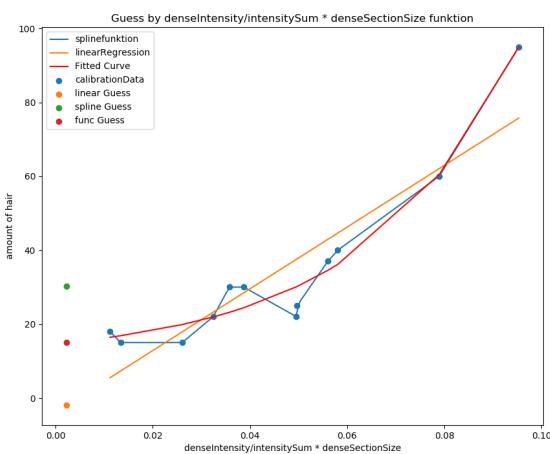


Abb. 22: Cover des Spiels

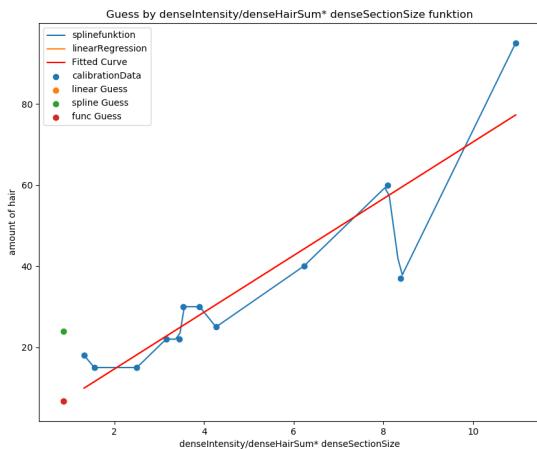


Abb. 23: Cover des Spiels

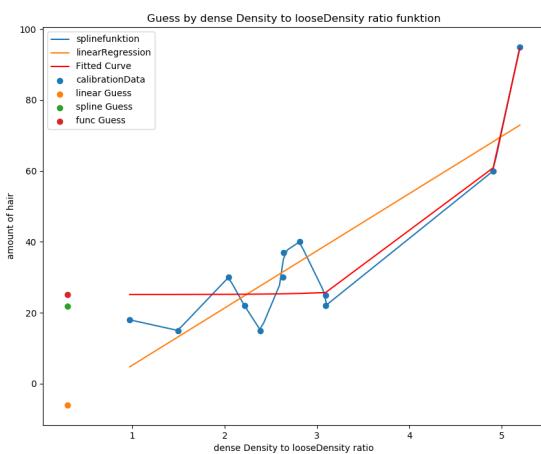


Abb. 24: Cover des Spiels

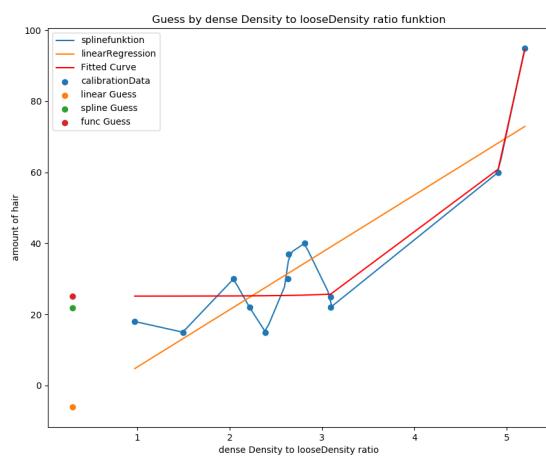


Abb. 25: Cover des Spiels