



FREIE UNIVERSITÄT BERLIN
FAKULTÄT FÜR INFORMATIK
IN ZUSAMMENARBEIT MIT DEM FRAUNHOFER
HEINRICH-HERTZ-INSTITUT

Masterarbeit

im Studiengang Informatik - Vertiefungsgebiet Computergraphik

zur Erlangung des akademischen Grades
Master of Science

Thema: ICP-basierte Anpassung eines statistischen Formmodells
des Kopfes an Portraitbilder

Autor: Marianne Krabi <marianne.krabi@fu-berlin.de>
Matrikelnummer: 4655135

Version vom: 11. Februar 2014

Erstgutachter: Prof. Dr. Raúl Rojas
Zweitgutachter: Prof. Dr. Peter Eisert
Betreuer: David Blumenthal-Barby

Eidesstattliche Erklärung

Eidesstattliche Erklärung zur Masterarbeit

Ich versichere, die von mir vorgelegte Arbeit selbstständig verfasst zu haben. Alle Stellen, die wörtlich oder sinngemäß aus veröffentlichten oder nicht veröffentlichten Arbeiten anderer entnommen sind, habe ich als entnommen kenntlich gemacht. Sämtliche Quellen und Hilfsmittel, die ich für die Arbeit benutzt habe, sind angegeben. Die Arbeit hat mit gleichem Inhalt bzw. in wesentlichen Teilen noch keiner anderen Prüfungsbehörde vorgelegen.

Unterschrift :

Ort, Datum :

Danksagung

An dieser Stelle möchte ich mich bei allen bedanken, die mich während des Schreibens dieser Masterarbeit tatkräftig unterstützt haben.

Besonders bedanken möchte ich mich bei meinen Betreuern am Fraunhofer HHI, Prof. Peter Eisert, Abteilungsleiter der Abteilung “Computer Vision & Graphics”, und David Blumenthal-Barby dergleichen Abteilung für die Vergabe dieses interessanten Themas. David Blumenthal-Barby hat mir sehr bei der Erstellung der Kostenfunktion und damit des Formmodells und der Bereitstellung der Daten geholfen. Weiterhin möchte ich ihnen für die kritische und konstruktive Auseinandersetzung mit meinen Lösungsansätzen und die guten Verbesserungsvorschläge danken.

Dankeschön auch an Johannes Furch für die schnelle Hilfe bei technischen Problemen, die mir etliche Stunden des Rätselratens erspart hat.

Selbstverständlich geht mein Dank auch an Prof. Raúl Rojas für die vielen Ideen und Anregungen das Thema zu erweitern und zu vertiefen.

Zusammenfassung

Das Modellieren von Köpfen ist ein schwieriges Problem in der Computergrafik. Das liegt nicht nur daran, dass der menschliche Kopf ein hoch komplexes Objekt ist. Auch die Leistung des Gehirns, “echte” Menschen von “unechten” zu unterscheiden, trägt einen großen Anteil dazu bei.

Diese Masterarbeit präsentiert zwei Algorithmen, in denen aus zwei orthogonalen Bildern einer Person mit Hilfe eines Formmodells ein 3D Kopf der Person rekonstruiert wird. In den Bildern werden manuell Merkmalspunkte extrahiert, die als Zielpunktwolke für bestimmte Vertices des Formmodells dienen. Es wird versucht, das Formmodell an die Zielpunktwolke anzupassen. Dafür wird ein Iterative Closest Point (ICP) Algorithmus verwendet.

Der Algorithmus verwendet keine Texturinformationen. Das unterscheidet ihn von bisherigen Ansätzen. Am Ende wird die Güte der Ergebnisse durch die Ähnlichkeit des Modells zu den Personen auf den Fotos bewertet.

Abstract

There's no more complicated issue than modelling human heads. The reason for that is not just the complexity of the face. Rather it is the human brain itself that makes it hard to model real faces or heads, because it can differentiate between “real” and “wrong” faces like no computer can.

In this master thesis two algorithms to reconstruct a human head from two orthogonal photographs using a morphing model is presented. Feature points are extracted manually from the images to form a target point cloud for specific corresponding vertices of the morphing model. An Iterative Closest Point (ICP) algorithm is used to fit the morphing model to the target cloud.

In former work texture information is used combined with the morphing model for reconstruction. This algorithm simply uses pure geometry to model the head. Finally the quality of the results is explored by comparing the model with the person on the images.

Inhaltsverzeichnis

Abbildungsverzeichnis	7
Tabellenverzeichnis	8
Abkürzungsverzeichnis	8
1 Einleitung	10
1.1 Stand der Forschung	10
1.2 Ziel und Aufbau der Arbeit	12
2 Theoretische Grundlagen	14
2.1 Hauptkomponentenanalyse	14
2.2 Thin Plate Splines	15
2.3 Iterative Closest Point Algorithmus	17
3 Statistisches 3D Formmodell des Kopfes	19
3.1 Referenzkopf	19
3.2 Originalköpfe	19
3.3 Anpassung der Originalköpfe an den Referenzkopf	21
3.4 Aufbau des statistischen 3D Formmodells	22
4 ICP-basierte Modellierung der 2D-Bilder	26
4.1 Datensätze	26
4.2 Profillinie und Landmarken	27
4.3 3D-3D Registrierung	29
4.3.1 Kostenfunktion der 3D Objektregistrierung	29
4.3.2 Nebenbedingungen der Kostenfunktion	33
4.3.3 Algorithmus der 3D-3D-Registrierung	34
4.4 2D-3D Registrierung	36
4.4.1 Anpassung der Profillinie	36
4.4.2 Ansatz 1: ICP der Landmarken ohne frontale Tiefeninformation	38
4.4.3 Ansatz 2: Ganzheitlich dreidimensionale Registrierung	41
5 Texturierung	44
5.1 Berechnung der Texturkoordinaten	44
6 Implementierung	46
6.1 Benutzeroberflächen	46
7 Ergebnisse	49
7.1 Beurteilung des Formmodells	49
7.2 Ergebnisse der Kostenfunktion	50
7.3 Ergebnisse der Profillinienrekonstruktion	58
7.4 Ergebnisse von Ansatz 1	63
7.5 Ergebnisse von Ansatz 2	67
7.6 Laufzeiten	71
8 Auswertung	73

9 Ausblick	80
10 Fazit	81
Literaturverzeichnis	82
Anhang	85

Abbildungsverzeichnis

1	TPS	16
2	Modellkopf	20
3	Originalscans	21
4	Anpassung von Originalscan an das Modell	22
5	Regionenaufteilung des Referenzkopfes	23
6	Gewichte der Überblendung	25
7	Blendinggrößen	25
8	2D-Daten	27
9	Ebenen	27
10	Landmarken	28
11	Profillinie des Formmodells	37
12	Ablauf der Profillinienrekonstruktion	38
13	Modell und zugehörige Profillinienanpassung	39
14	Verformung durch Thin Plate Splines	42
15	Ablauf des TPS-ICP Algorithmus	43
16	Koordinatensysteme	45
17	Benutzeroberfläche 1	47
18	Benutzeroberfläche 2	48
19	Scanfehler	49
20	Varianten des Formmodells	50
21	Formmodelle mit Gewicht 1	51
22	Der Einfluss der Anzahl der Formparameter auf den mittleren Fehler 22(a) und die Differenz der ICP mit Formparametern und ohne 22(b).	52
23	Der Einfluss des Glattheitsfaktors λ_1 auf den mittleren Fehler 23(a) und die Differenz der ICP mit Formparametern und ohne 23(b).	53
24	Der Einfluss des Faktors für Formbeschränkung λ_2 auf den mittleren Fehler 24(a) und die Differenz der ICP mit Formparametern und ohne 24(b).	57
25	Minimum und Maximum der Testfälle	58
26	Beispielergebnisse der 3D-3D Registrierung	59
27	Durchschnittliche Fehlerentwicklung über alle Bilder der Profillinienrekonstruktion mit Testfall 10-10-0. Das Diagramm zeigt die Entwicklung der Transformation und des darauffolgenden Schrittes der ICP mit Form und Transformation.	60
28	Abstand von 1 cm im Modell	60
29	Durchschnittlicher Fehler der Anpassung des Formmodells an die Profillinie. Der Ausgangsfehler nach der Transformationsberechnung ist gelb dargestellt. Die Parameterkonstellationen mit Fehlerverbesserungen sind grün und die übrigen rot dargestellt. Die Fehlereinheit ist wieder Millimeter.	62
30	Durchschnittliche Anzahl der Iterationen bei der Anpassung an die Profillinie	62
31	Anpassung des Formmodells an eine Profillinie	63
32	Ergebnis von Ansatz 1: Die Testfälle haben die Beschriftung: Formparameter - λ ; mit λ_2 in Abbildung 32(a) und λ_1 in Abbildung 32(b).	64
33	Ergebnis von Ansatz 1: Einfluss des Gewichtes der Frontallandmarken auf den mittleren Fehler pro Vertex.	64

34	Ergebnis von Ansatz 1: Die Anpassung der Modellvertices (rot) an die Ziellandmarken (grün) ausgehend von der Ausgangsposition (blau). . .	65
35	Ergebnis von Ansatz 1: Neue Perspektiven des rekonstruierten Modells.	66
36	Ergebnis Ansatz 2: Mittlerer Fehler der Iterationen des TPS/ICP Algorithmus	67
37	Ergebnis Ansatz 2: Abstände des Formmodells zu den Bildern	68
38	Mittlerer Abstand des Formmodells zu den Bildern	69
39	Ergebnis Ansatz 2: Unterschied der Z-Koordinate von Modell und Bild	70
40	Ergebnis Ansatz 2: Ergebnismodell des Datensatzes mit dem geringsten Fehler	70
41	Laufzeitentwicklung der einzelnen Iterationen der TPS-ICP Verformung	72
42	Formmodell nach Blanz und Vetter	73
43	Ohren verschiedener Formmodelle	74
44	Deformation generischer Modelle	75
45	Neue Gesichtswinkel des nur mit TPS verformten Modells	76
46	Formmodell Ergebnis von Blanz und Vetter	77
47	Formmodell Ergebnis von Jiang	77
48	Formmodell Ergebnis in Breuer et. al.	78
49	3D Modell mit anderen Vergleichsblickwinkeln (1)	85
50	3D Modell mit anderen Vergleichsblickwinkeln (2)	86
51	3D Modell mit anderen Vergleichsblickwinkeln (3)	87
52	3D Modell mit anderen Vergleichsblickwinkeln (4)	88

Tabellenverzeichnis

1	Frontal Landmarken	28
2	Ergebnisse der Testfälle der ICP nur mit Transformationsparametern Skalierung, Translation und Rotation. Die Fehler sind in Millimeter pro Vertex bei einer durchschnittlichen Kopfgröße angegeben.	54
3	Ergebnisse der Testfälle der ICP mit Formparametern für männliche Daten.	55
4	Ergebnisse der Testfälle der ICP mit Formparametern für weibliche Daten.	56

Abkürzungsverzeichnis

2D	2-dimensional
3D	3-dimensional
ICP	Iterative Closest Point
PCA	Hauptkomponentenanalyse (Principal Component Analysis)
SVD	Singulärwertzerlegung (Singular Value Decomposition)
TPS	Thin Plate Splines

Symoltabelle

Symbol	Bedeutung	Referenz
M	Anzahl der Datensätze	
N	Dimension eines Datensatzes ($\hat{=}$ Anzahl der Vertices des Kopfes)	
μ	Mittelwert der Datensätze	Seite 14
\mathbf{d}_i	Datensatz	Seite 14
$\widehat{\mathbf{D}}$	Matrix aller zentrierten Datensätze	Seite 14
\mathbf{C}	Kovarianzmatrix der Daten mit Größe $3N \times 3N$	Seite 14
\mathbf{V}	Diagonalmatrix der Eigenvektoren	Seite 14
$\mathbf{\Lambda}$	Matrix der Eigenwerte	Seite 14
\mathbf{L}	Kovarianzmatrix der Daten mit Größe $M \times M$	Seite 15
\mathbf{v}_i	M Eigenvektoren von \mathbf{L}	
\mathbf{H}	Hauptkomponentenmatrix	Seite 15
\mathbf{w}	Gewichte der Hauptkomponenten	Seite 15
\mathbf{p}_i	Vertex des Formmodells	Seite 17
\mathbf{q}_i	Vertex der Zielpunktwolke	Seite 17
\mathbf{I}_3	Identitätsmatrix der Dimension 3	
\mathbf{t}	Translation	Seite 17
s	Skalierung	Seite 17
$\text{rot}(\boldsymbol{\theta})$	Rotationsdarstellung mit Eulerwinkeln um Winkel $\boldsymbol{\theta}$	Seite 17
\mathcal{R}	Region des Formmodells	Seite 23
K	Anzahl der Regionen	Seite 23
\mathbf{b}_i^j	Vertexkoordinaten der Region j für Datensatz i	Seite 24
\mathbf{M}_i	Formmatrix eines Vertex \mathbf{p}_i	Seite 30
\mathbf{m}	Gewichtung der Formmatrix \mathbf{M}_i	Seite 30
$e()$	Fehler- bzw. Kostenfunktion	Seite 32
$\boldsymbol{\Gamma}$	”Smoothness Prior”, Nebenbedingung für die Randglättung des Formmodells	Seite 33
$\mathbf{\Lambda}$	Nebenbedingung für die Verformungsstärke des Formmodells	Seite 34
λ_1	Gewichtung von $\boldsymbol{\Gamma}$	
λ_2	Gewichtung von $\mathbf{\Lambda}$	

1 Einleitung

Das menschliche Gehirn ist einzigartig bei der Verarbeitung und Erkennung von Menschen und Gesichtern. Sehen wir ein Foto, können wir uns die Person sofort dreidimensional vorstellen. Selbst mit anderer Frisur, Bart oder halb verdeckt erkennen wir die gleiche Person immer wieder. Nicht umsonst stolpern wir beim Schauen von animierten 3D Filmen sofort über “halbmenschliche” Gestalten, die unser Gehirn sofort als unecht einstuft.

Der Bedarf an der Rekonstruktion von realistischen, dreidimensionalen Köpfen und Gesichtern steigt. Filme und Spiele werden zunehmend in 3D produziert, doch die Hauptdarsteller sehen kaum menschlich aus.

Eine weitere Anwendung findet sich in der Kriminalistik. Oft gibt es zur Verbrechensbekämpfung nur einzelne Bilder von gesuchten Personen. Eine 3D Repräsentation kann hilfreich sein, Personen auch aus Blickwinkeln wiederzuerkennen, die man nicht auf den Bildern sieht.

In dieser Arbeit wird daher ein Algorithmus zur 3D Rekonstruktion des menschlichen Kopfes aus zwei Fotos beschrieben. Das Ergebnis ist ein Dreiecksnetz eines Kopfes, der durch den Computer grafisch dargestellt werden kann.

Wie die mathematische Repräsentation erstellt wird und welche Ergebnisse erreicht werden, ist in der folgenden Arbeit erklärt.

1.1 Stand der Forschung

Eine 3D Gesichtsrekonstruktion kann auf verschiedene Art und Weise erfolgen. Eine gängige Unterteilung geschieht nach der Art der Informationsgewinnung. In diesem Fall gibt es die Möglichkeiten, ein einziges 2D Bild, mehrere Bilder oder sogar eine 3D Vorlage als Grundlage für die Rekonstruktion zu verwenden. In diesen Quellen können alle nötigen Informationen enthalten sein oder es müssen noch weitere Parameter berechnet werden.

Eine weitere Unterscheidung geschieht nach der Herangehensweise an die Rekonstruktion. Es gibt zum Beispiel den “Shape from Shading” Ansatz [FOK02][SKHE11], bei dem die Form eines Körpers über die Helligkeitsverteilung bestimmt werden kann, bei “Shape from Silhouettes” wird die Silhouette als Formgrundlage verwendet oder aber der “Analysis by Synthesis” [BV99] [BKK⁺08] Ansatz, in dem veränderbare Formmodelle verwendet werden, um die Übereinstimmung von Foto und 3D Rekonstruktion zu maximieren. Ein anderer, recht intuitiver Ansatz, ist die Verwendung von Stereobildern [Koc96] [SHF09]. Der Aufbau der zwei Kameras ist den menschlichen Augen nachempfunden. Anhand von Merkmalspunkten in diesen zwei Bildern können 3D Punkte berechnet werden, die für eine Rekonstruktion geeignet sind. Eine Übersicht über die verschiedenen Methoden ist in [WD08] gegeben.

Für alle Algorithmen und Ansätze muss zu Beginn der Kopf bzw. das Gesicht aus dem Bild, einem Video oder dem 3D Scan lokalisiert werden. Weiterhin arbeiten alle Ansätze mit mehr oder weniger Merkmalspunkten, die manuell oder automatisch aus den Bildern extrahiert werden.

Der “Analysis by Synthesis” Ansatz benötigt von allen Methoden die wenigsten Daten. Im Gegensatz zu statistischen Methoden, in denen viele Daten für die Erstellung eines Wahrscheinlichkeitsmodells benötigt werden, erreicht dieser Ansatz bereits mit 100 Datensätzen ein gutes Ergebnis. Blanz und Vetter [BV99] lieferten dabei mit ihrem entwickelten Formmodell die ersten Ergebnisse. Sie verwenden eins oder mehrere 2D Bilder und ein Modell mit Form und Textur, das sie aus einer Datenbank erstellt haben. Dieses Modell wird anhand manuell platzierter Merkmalspunkte in Position, Form, Farbe und Helligkeit an die Bilder angepasst. Die besten Parameter dafür werden durch eine Likelihood Berechnung gefunden. Das Ergebnis ist ein texturiertes 3D Modell.

Weitere Verwendung der Formmodelle ist in [BKK⁺08] zu finden. Hierbei wird allerdings nur ein einziges Bild verwendet, um die nötigen Merkmalsinformationen zu erlangen. Eine Neuerung in diesem Ansatz ist die automatisierte Extrahierung der Punkte aus dem Bild. Die Anpassung des Formmodells erfolgt dann ähnlich zu dem Ansatz bei Blanz und Vetter[BV99]. Probanden bewerten die Ergebnisse dieser Arbeit als durchwachsen. Einige Modelle wurden als sehr gut eingeschätzt, ein Drittel der Daten aber auch als schlecht.

Andere Ansätze verwenden generische Gesichtsmodelle, die mit Hilfe von Deformationen wie der Dirichlet Free Form Deformation ([LMT98], [LKT97]) oder der Radialen Basis Funktionen ([WJC12]) das Gesichtsmodell verformen. Die dabei benötigten Merkmale müssen aus mehreren Bildern gewonnen werden, da diese Transformationen nur mit 3D Informationen berechnet werden können. Aus der Markierung der Merkmale in zwei nicht gleichen Bildern wird die 3D Information extrahiert. Die Ergebnisse dieser Methoden sind aufgrund fehlender natürlicher Gesichtsinformationen bei der Deformation weniger detailreich. Daher ist ein wichtiges Augenmerk bei dieser Variante die Texturierung.

Fanany [FOK02] benutzt wiederum einen anderen Ansatz, nämlich ein neuronales Netz in Kombination mit dem “Shape from Shading” Ansatz, um aus mehreren 2D Bildern die Tiefeninformation eines Gesichtes zu extrahieren. Auf dieser Tiefenkarte verwendet er ein neuronales Netz, um die Beleuchtungs-, Positions- und Formparameter zu bestimmen.

Eine weitere Verwendung der 3D Rekonstruktion ist der Ausdruck von Emotionen auf dem erstellten Modell. Patel [PZ10] verwendet dafür ebenfalls ein generisches Gesichtsmodell. Aus einem 2D Bild adaptiert er automatisch bestimmte Gesichtszüge und transformiert das 3D Modell passend zu diesen Informationen. Mit Hilfe des MPEG-4 Standards verformt er die Gesichtszüge zu den im Standard definierten Mimiken.

Pighin et.al. [PHL⁺98] beschreiben ebenfalls eine Möglichkeit für die Rekonstruktion von Gesichtsausdrücken. Sie verwenden dafür mehrere Fotos und darauf markierte Merkmale. Mittels einer Interpolation wird auch hier ein generisches Modell angepasst. Dieses kann der Anwender in einer grafischen Oberfläche mit verschiedenen Gesichtszügen versehen. Grundlage für ein gutes Ergebnis ist auch hier wieder die Texturierung.

1.2 Ziel und Aufbau der Arbeit

Allen Ansätzen ist gemein, dass sie besonderen Wert auf eine gute Texturierung legen. Die darunter liegende Geometrie hat nicht selten wenig Ähnlichkeit mit der Person auf den Fotos. Bei der Erstellung der Formmodelle werden häufig auch Texturmodelle oder Texturinformationen verwendet, die sich gemeinsam mit der Form anpassen [BV99] [RBV02]. Auch arbeiten die meisten Formmodelle [BV99] [BKK⁺08] [JHY⁺05] nur auf den Gesichtsdaten und nicht auf dem gesamten Kopf.

In dieser Arbeit wird versucht, diese Probleme zu lösen. Ziel ist es, aus zwei orthogonalen Fotos unbekannter Kamerakalibrierung mit Hilfe eines eigens berechneten Formmodells eine detailreiche 3D Rekonstruktion des gesamten Kopfes zu erstellen. Dafür werden aus den Bildern manuell Landmarken extrahiert, an welche sich das Formmodell anpassen soll. Die Textur auf den Fotos ist dabei irrelevant und fließt nicht in die Berechnungen ein. Der Algorithmus arbeitet allein mit der Lage der Merkmalspunkte und der Geometrie des Formmodells. Ein Formmodell berechnet sich aus einer Hauptkomponentenanalyse (Abschnitt 2.1) von einem Datensatz mit verschiedenen 3D Köpfen. Das Modell ist die Repräsentation dieses Datensatzes durch gewichtete Eigenvektoren der PCA. Jeder Eigenvektor kann ein bestimmtes Merkmal - wie zum Beispiel die Länge der Nase oder die Größe des Kopfes - verändern, wenn er ein entsprechendes Gewicht bekommt. Durch eine geeignete Kostenfunktion können diese Gewichte so bestimmt werden, dass das Formmodell bestmöglich mit den Positionen der Landmarken in den Bildern übereinstimmt. Die Erstellung des Formmodells ist in Abschnitt 3 erläutert.

Diese Kostenfunktion ist eine Erweiterung des klassischen Iterative Closest Point Algorithmus (ICP). Dieser passt eine Ausgangspunktwolke (in diesem Fall die zu den Landmarken korrespondierenden Vertices des Formmodells) an eine Zielpunkt wolke (die markierten Merkmalspunkte) in Translation, Rotation und Skalierung so an, dass die korrespondierenden Punkte den kleinstmöglichen Abstand zueinander besitzen. In dieser Arbeit wird die Ausgangspunkt wolke weiterhin noch in der Form bestmöglich an das Ziel "gezogen". Das Minimieren der Kostenfunktion ist ein Optimierungsproblem, das mit dem Gauss-Newton-Verfahren - eine komplexere Form des Gradientenabstiegs - gelöst wird. Beschrieben sind Merkmalsextraktion und die verwendete Kostenfunktion in Abschnitt 4.

Die im Folgenden beschriebene 3D Rekonstruktion besteht aus zwei Schritten - der Profillinienanpassung und der Frontalanpassung des Formmodells. Ziel des ersten Schrittes ist es, das Formmodell mit seiner Silhouette bestmöglich an die im Bild markierte anzupassen. Damit bei einer 3D Rekonstruktion durch die ICP eine vollständige Anpassung im Raum erfolgen kann, sollten in der Zielpunktwolke Punkte mit gleicher Dimensionalität vorliegen wie in der Startpunktwolke. Die Markierungen in den Bildern besitzen jedoch aufgrund der Zweidimensionalität eines Fotos eine Dimension weniger. Dadurch wird bei der Profillinie angenommen, sie liege in einer Ebene. So bekommen die Ziellandmarken alle den gleichen Wert für die dritte Koordinate zugewiesen. Abschnitt 4.4.1 befasst sich mit der Anpassung der Silhouette.

Schwieriger ist es bei der Anpassung der Frontallandmarken. Diese besitzen jeweils eine andere, unbekannte Tiefe. Für dieses Problem werden zwei Lösungsansätze entwickelt.

Im ersten Ansatz wird eine gemeinsame ICP aus Profil- und Frontallandmarken berechnet. Die unbekannten Koordinaten der Frontallandmarken bleiben dabei jedoch unberücksichtigt, indem sie nicht in das Gleichungssystem zur Lösung der ICP aufgenommen werden. Die fehlende Information der Tiefe kann allein aus den Profillandmarken gewonnen werden.

Der zweite Ansatz ist eine Kombination aus Deformation des Formmodells durch Thin Plate Splines (TPS, Abschnitt 2.2) und ICP-Anpassung an das aus der Deformation resultierende Modell. Bei der Deformation durch TPS fungieren die Merkmalspunkte als Stützstellen, wobei für die fehlende Koordinate der Merkmalspunkte der Wert des korrespondierenden Vertex des Formmodells verwendet wird. Nach der Deformation befinden sich die Vertices der Merkmalspunkte an den vorgegebenen Positionen und alle anderen Punkte werden glatt interpoliert. Der dadurch deformierte 3D Kopf dient als Zielpunktwolke für die ICP. Diese optimiert dann die Parameter des Formmodells.

Die beiden Ansätze werden im Detail in den Abschnitten 4.4.2 und 4.4.3 beschrieben.

Im Gegensatz zu vielen Ansätzen in der Literatur wird hier besonderer Wert auf die Güte der berechneten Geometrie gelegt. Eine anschließende Texturierung spielt nur eine nebensächliche Rolle. Die verwendete Texturierung und die Ergebnisse sind in den Abschnitten 5 und 7 dargelegt. Anschließend werden die Ergebnisse mit den bisherigen Arbeiten verglichen.

2 Theoretische Grundlagen

2.1 Hauptkomponentenanalyse

Ziel der Hauptkomponentenanalyse ist die Dimensionsreduktion der Datenmenge durch eine Dekorrelierung der Variablen. In dieser Darstellung durch sogenannte Hauptkomponenten enthalten bereits wenige Komponenten einen Großteil der gesamten in der Datenmenge vorhandenen Informationen. Die Daten werden also so transformiert, dass die erste Komponente die größte Varianz und die letzte Komponente die geringste Varianz hat.

Die Hauptkomponenten werden aus der Lösung des Eigenwertproblems der Kovarianzmatrix der Daten gewonnen.

Dafür wird für jede Variable der Mittelwertsvektor $\boldsymbol{\mu}$ gebildet:

$$\boldsymbol{\mu} = \frac{1}{M} \sum_{i=1}^M \mathbf{d}_i \quad (1)$$

wobei M die Anzahl der Datensätze ist und $\mathbf{d}_i = (x_1 \ y_1 \ z_1 \ \dots \ x_N \ y_N \ z_N)^T$ ein Vektor aller Variablen (Punkte) eines Datensatzes. In dieser Arbeit ist ein Datum die 3D Darstellung eines Kopfes. In dreidimensionalen Daten wird jeder Punkt durch seine drei Koordinaten repräsentiert, wodurch \mathbf{d}_i die Länge $3N$ hat; N ist die Anzahl der Punkte. Die Daten haben eine Ausdehnung von $x = [0, 1]$, $y = [0, 1]$ und $z = [0, 1]$. Die Daten können beliebig skaliert und mit einem geeigneten Programm (z.B. MeshLab) dargestellt werden.

Die Datensätze werden nun um den Mittelwert zentriert ($\widehat{\mathbf{D}}$) und daraus die Kovarianzmatrix \mathbf{C} berechnet:

$$\widehat{\mathbf{D}} = [\mathbf{d}_1 - \boldsymbol{\mu} \ \dots \ \mathbf{d}_M - \boldsymbol{\mu}] \quad (2)$$

$$\mathbf{C} = \frac{1}{M} \sum_{i=1}^M (\mathbf{d}_i - \boldsymbol{\mu})(\mathbf{d}_i - \boldsymbol{\mu})^T \quad (3)$$

$$= \widehat{\mathbf{D}}\widehat{\mathbf{D}}^T, \quad (4)$$

wobei M die Anzahl der Datensätze ist. Die hier verwendeten Größen von M und N liegen bei $M \approx 200$ und $N \approx 9000$.

Die Spektralzerlegung der Kovarianzmatrix ergibt $\mathbf{C} = \mathbf{V}\mathbf{\Lambda}\mathbf{V}^T$. $\mathbf{\Lambda}$ ist die Diagonalmatrix der Eigenwerte und \mathbf{V} die Matrix der Eigenvektoren.

In Bild- und 3D-Objektdaten sind diese Kovarianzmatrizen sehr groß $((3N) \times (3N))$ und die Berechnung von $3N$ (hier: 9000^2) Eigenwerten und -vektoren erfordert immensen Rechen- und Speicherbedarf. Aus diesem Grund präsentieren Turk und Pentland [TP91] eine Methode, die die Eigenwerte und -vektoren nicht für die Anzahl der Koor-

dinaten, sondern für die Anzahl der vorhandenen Datensätze berechnet. Sie begründen weiterhin, dass es ohnehin nur $M - 1$ Eigenvektoren mit Eigenwerten ungleich Null gibt, wenn $M < 3N$, es also weniger Datensätze gibt als die Dimension der Daten ist. Mit einer geeigneten Linearkombination dieser Vektoren mit den gemittelten Daten \mathbf{d}_i werden dann die ursprünglichen Eigenvektoren der Matrix \mathbf{C} rekonstruiert. Diese Variante reduziert die Anzahl der Hauptkomponenten von $3N$ auf M , wobei in der Regel $M \ll 3N$. Die zu Grunde liegende Kovarianzmatrix ist hier

$$\mathbf{L} = \widehat{\mathbf{D}}^T \widehat{\mathbf{D}} \text{ wobei} \quad (5)$$

$$L_{mn} = (\mathbf{d}_m - \boldsymbol{\mu})^T (\mathbf{d}_n - \boldsymbol{\mu}), \quad (6)$$

wobei \mathbf{L} die Größe $M \times M$ hat.

Die Multiplikation der Matrix $\widehat{\mathbf{D}}$ mit den Eigenvektoren \mathbf{v}_i von \mathbf{L} ergibt die Hauptkomponentenmatrix \mathbf{H} mit den wichtigen M Hauptkomponenten. Sie hat die Größe $3N \times M$.

$$\mathbf{H} = \widehat{\mathbf{D}} \begin{bmatrix} \mathbf{v}_1 & \cdots & \mathbf{v}_M \\ \vdots & & \vdots \end{bmatrix} \quad (7)$$

Auch die reduzierte Anzahl der Hauptkomponenten ist nach Wichtigkeit sortiert, sodass bereits wenige Komponenten einen großen Teil der Informationen des Datensatzes beinhalten. Eine Spalte dieser Matrix repräsentiert einen Datensatz, Turk und Pentland bezeichnen sie als "Eigenfaces", da sie die Methode im Kontext der Gesichtserkennung mittels PCA auf den Pixelwerten von Fotos entwickelt haben. Eine Spalte ihrer Matrix ergab als 2D Darstellung daher ein Gesicht.

Nun können Daten (\mathbf{d}_i) in den Hauptkomponentenraum und zurück transformiert werden. Dafür müssen Gewichte $\mathbf{w} = (w_1 \dots w_M)$ berechnet werden, mit denen die Hauptkomponenten gewichtet werden. Die beiden Transformationen sind:

$$\mathbf{d}_i \rightarrow \mathbf{H} : \mathbf{w}_i = \mathbf{H}^T (\mathbf{d}_i - \boldsymbol{\mu}) \quad (8)$$

$$\mathbf{d}_i \leftarrow \mathbf{H} : \mathbf{d}_i = \mathbf{H} \mathbf{w}_i + \boldsymbol{\mu}. \quad (9)$$

Das wird auch als Singulärwertzerlegung (SVD) bezeichnet. Eine Anwendung dieser PCA Variante ist in Abschnitt 3.4 zu sehen.

2.2 Thin Plate Splines

Splines werden häufig bei glatten Oberflächeninterpolationen, bei der nur wenige Punkte bekannt sind, eingesetzt. Der Name Thin Plate Splines kommt vom physikalischen Modell einer dünnen Metallplatte, die an den Stützstellen unbeweglich ist und sich sonst verformen kann[Boo89]. Entsprechend der Eigenschaften einer Metallplatte ist

das Ziel der Thin Plate Splines, die Energie für das Verbiegen der Platte zu minimieren, wobei der Abstand der Platte an definierten Stützstellen Null ist. Mathematisch kann das durch die biharmonische Gleichung $\nabla^4 \phi = 0$ ausgedrückt werden. Diese “Energie” $E(\phi(x, y))$ einer Funktion ϕ definiert Bookstein entsprechend der biharmonischen Gleichung im kartesischen Koordinatensystem für alle Punkte der Fläche in \mathbb{R}^2 als das Integral (in \mathbb{R}^2) der Quadrate der zweiten Ableitungen

$$E(\phi(x, y)) = \iint_{\mathbb{R}^2} \left(\left(\frac{\partial^2 \phi}{\partial x^2} \right)^2 + 2 \left(\frac{\partial^2 \phi}{\partial x \partial y} \right)^2 + \left(\frac{\partial^2 \phi}{\partial y^2} \right)^2 \right) dx dy. \quad (10)$$

Die Lösungen dieser biharmonischen Gleichung in \mathbb{R}^1 , \mathbb{R}^2 oder \mathbb{R}^3 bilden die radialen Basisfunktionen der Thin Plate Splines für Daten in der entsprechenden Dimension².

$$\mathbb{R}^1 : U(r) = |r|^3 \quad (11)$$

$$\mathbb{R}^2 : U(r) = r^2 \log r \quad (12)$$

$$\mathbb{R}^3 : U(r) = |r| \quad (13)$$

Damit die Energie so gering wie möglich gehalten wird, sind im Ergebnis der Interpolation durch TPS auch Transformationen (Rotation, Translation, Skalierung) vorhanden. Solch eine Interpolation ist in Abbildung 1 zu sehen.

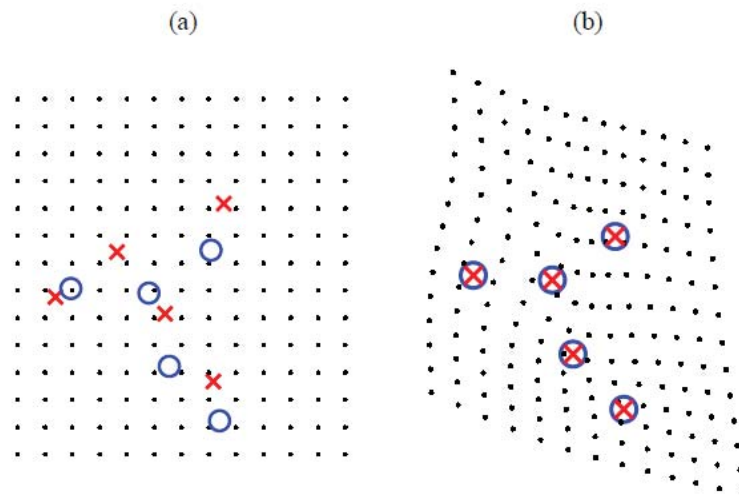


Abbildung 1: Beispiel einer TPS Interpolation. Die blauen Kreise stellen die Zielpositionen der roten Kreuze dar. Es wird angenommen, dass jedes Kreuz einen bestimmten korrespondierenden Kreis hat(a). Nach der Interpolation (b) haben die Kreuze die gleiche Position wie ihre Korrespondenzpunkte und die restliche Oberfläche ist entsprechend der Energieminimierung verzerrt.³

¹Bookstein, vgl. [Boo89] [S.568]

²Bookstein, vgl. [Boo89] [S.584]

³Bildquelle: Donato und Belongie, vgl. [DB03]

2.3 Iterative Closest Point Algorithmus

Im Bereich der Computer Vision, in der Medizinischen Bildverarbeitung oder in der Robotik wird oft versucht, Punktwolken - zum Beispiel von Laser Scans - einander anzupassen. Dabei wird ein Modell-Template so transformiert, dass es bestmöglich auf ein Zielobjekt passt. Passen heißt in diesem Sinne, dass eine Fehlerfunktion minimiert wird. Ein Fehler könnte beispielsweise die euklidische Distanz zweier korrespondierender Datenpunkte sein. Eine weitere Anwendung ist das Zusammensetzen eines Laserscans aus verschiedenen Modellen ohne dass die genaue Lage der Objekte im Scan bekannt sein muss.

Für diese Anwendungen eignet sich der ICP Algorithmus [FH86] [Zha94]. Durch diesen Algorithmus wird eine Punktwolke bewegt, skaliert und rotiert, sodass eine Fehlerfunktion minimiert wird. Dadurch wird erreicht, dass die Überdeckung beider Punktmengen größtmöglich ist. Beinhaltet die Funktion den Abstand, wird die Distanz der Punkte des Modells zu dem nächsten Punkt der Referenz kleinstmöglich. Diesen Vorgang nennt man auch Registrierung.

In dieser Arbeit wird ein Minimierungsproblem gelöst, dass auf Faugeras und Herberts [FH86] Formulierung des Findens der kleinsten Abstände von Punkten basiert. Dabei ist $d(\mathbf{p}_i, \mathbf{q}_i) = \|\mathbf{p}_i - \mathbf{q}_i\|$ der euklidische Abstand eines Punktes $\mathbf{p}_i = (x_i, y_i, z_i)$ des Modells zu einem Punkt $\mathbf{q}_i = (x'_i, y'_i, z'_i)$ der originalen Szene. Es wird versucht, das Modell so zu bewegen, dass die Abstände zu den Szenenpunkten minimal werden.

Um nun die beste Transformation für die Punkte des Modells zu finden, wird folgendes Minimierungsproblem mit Rotation $\text{rot}(\boldsymbol{\theta})$ um einen Vektor $\boldsymbol{\theta}$ und Translation \mathbf{t} formuliert [FH86] und der Fehler e berechnet. Für die Rotation wird eine lineare Approximation der Darstellung durch Euler Winkel mit $\boldsymbol{\theta} = [\theta_x \ \theta_y \ \theta_z]^T$ verwendet:

$$\text{rot}(\boldsymbol{\theta}) \approx \mathbf{I}_3 + \begin{bmatrix} 0 & -\theta_z & \theta_y \\ \theta_z & 0 & -\theta_x \\ -\theta_y & \theta_x & 0 \end{bmatrix} \quad (14)$$

Die Gleichung für das Optimierungsproblem ist dann:

$$\min_{\boldsymbol{\theta}, \mathbf{t}} e(\boldsymbol{\theta}, \mathbf{t}) = \min_{\boldsymbol{\theta}, \mathbf{t}} \sum_{i=1}^N \|\text{rot}(\boldsymbol{\theta})\mathbf{p}_i + \mathbf{t} - \mathbf{q}_i\|^2 \quad (15)$$

Hier wird der quadratische Abstand verwendet, damit das Problem mit der Methode der kleinsten Fehlerquadrate gelöst werden kann.

Da in der Anwendung dieser Arbeit Original und Referenz sich auch in der Skalierung s unterscheiden, wird diese Transformation auch in die Fehlerfunktion aufgenommen. Die Gleichung ändert sich dann in:

$$\min_{s, \boldsymbol{\theta}, \mathbf{t}} e(s, \boldsymbol{\theta}, \mathbf{t}) = \min_{s, \boldsymbol{\theta}, \mathbf{t}} \sum_{i=1}^N \|s \text{rot}(\boldsymbol{\theta}) \mathbf{p}_i + \mathbf{t} - \mathbf{q}_i\|^2 \quad (16)$$

Ein Lösungsansatz ist die in [FH86] vorgeschlagene iterative Lösung für sehr kleine Änderungen Delta (Δ) in den Parametern. Die Formel für den Fehler lautet:

$$\min_{(1+\Delta s)s, \text{rot}(\Delta \boldsymbol{\theta})\text{rot}(\boldsymbol{\theta}), \mathbf{t}+\Delta \mathbf{t}} e((1+\Delta s)s, \text{rot}(\Delta \boldsymbol{\theta})\text{rot}(\boldsymbol{\theta}), \mathbf{t}+\Delta \mathbf{t}) = \sum_{i=1}^N \|(s+\Delta s)\text{rot}(\Delta \boldsymbol{\theta})\text{rot}(\boldsymbol{\theta})\mathbf{p}_i + \mathbf{t} + \Delta \mathbf{t} - \mathbf{q}_i\|^2 \quad (17)$$

Die Änderung der Rotation ist abhängig von der verwendeten Darstellung. Bei der Verwendung von Euler-Winkeln ändert sich die Rotation multiplikativ, die anderen Parameter werden durch Addition verändert. Diese Gleichung kann durch SVD gelöst werden.

Die ICP besteht aus der Iteration zweier Schritte. Im ersten Schritt werden Punktkorrespondenzen im Sinne des kleinsten Abstandes gefunden. Dadurch kann die ICP auch mit unterschiedlich großen Punktwolken gelöst werden. Für das Finden von Korrespondenzen können Strukturen wie K-d-Bäume verwendet werden. Eine weitere Möglichkeit ist die manuelle Festlegung von Korrespondenzpunkten.

Im zweiten Schritt wird die Gleichung 17 durch SVD gelöst und die neue Transformation der Modellpunkte berechnet. Diese beiden Schritte werden wiederholt, bis der Fehler konstant bleibt.

3 Statistisches 3D Formmodell des Kopfes

Ein statistisches Formmodell (engl. morphable model) repräsentiert mögliche Formvariationen eines Objektes, hier: des Kopfes, innerhalb statistisch plausibler Grenzen, die aus einem Datensatz berechnet werden. Es ist die auf einer PCA Analyse basierende Darstellung des Objektes durch gewichtete Eigenvektoren. Anhand dieser kompakten Darstellungsform kann ein Startmodell - die aus den Daten gemittelte Form - mit Hilfe einer geeigneten Gewichtung der Eigenvektoren in verschiedene Objektausprägungen überführt werden.

Formmodelle eignen sich gut für die Objekterkennung in der Bildverarbeitung[BV03], zur Klassifizierung oder Segmentierung von Bilddaten [LLS03] [San10]. Die Erstellung des Formmodells des Kopfes wird im Folgenden beschrieben.

3.1 Referenzkopf

Das Formmodell wird aus einem Datensatz realer Köpfe berechnet. Dafür brauchen die Daten eine einheitliche Repräsentation. Für diesen Fall heißt das, eine einheitliche Topologie eines Dreiecksnetzes mit gleicher Knotennummerierung. Diese ist in den Originaldaten nicht gegeben.

Daher braucht man einen Referenzkopf, an den die Topologie der originalen Köpfe angepasst wird. Dieser Referenzkopf ist ein menschlicher, asexueller Durchschnittskopf. Er wurde mit der kostenlosen Software Blender⁴ erstellt. Es handelt sich um ein trianguliertes, dreidimensionales Netz mit 9.071 Knoten und 18.060 Dreiecken. Der Vorteil des Modells liegt in der symmetrischen Aufteilung der Dreiecke auf den beiden Gesichtshälften und in der variablen Dreiecksanordnung. Feiner definierte Bereiche wie Augen oder Ohren haben eine dichtere Knotenanordnung als ebene Flächen wie die Schädeldecke (vgl. Abbildung 2). Die Symmetrie ist bei der Erstellung des Formmodells hilfreich und durch die Variabilität in der Anordnung werden die Knoten in glatten Bereichen wie Schädel und Hals reduziert, was wiederum in einem Performanzzuwachs resultiert.

Die Besonderheit dieses Modells gegenüber vielen anderen Modellen, bei denen nur das Gesicht vorhanden ist, also der Kopf oberhalb von Ohren und Stirn endet [BV99] [BV03] [JHY⁺05], ist die Gesamtheit des Kopfes. Es werden auch Schädel und Hals in die Modellierung mit einbezogen. Dieses Referenzmodell bildet die Grundlage des Formmodells.

3.2 Originalköpfe

Für das Formmodell dieser Arbeit werden Laserscans kanadischer Soldatinnen und Soldaten verwendet. Diese wurden mit einem Cyberware Scanner aufgenommen.

⁴vgl. www.blender.org

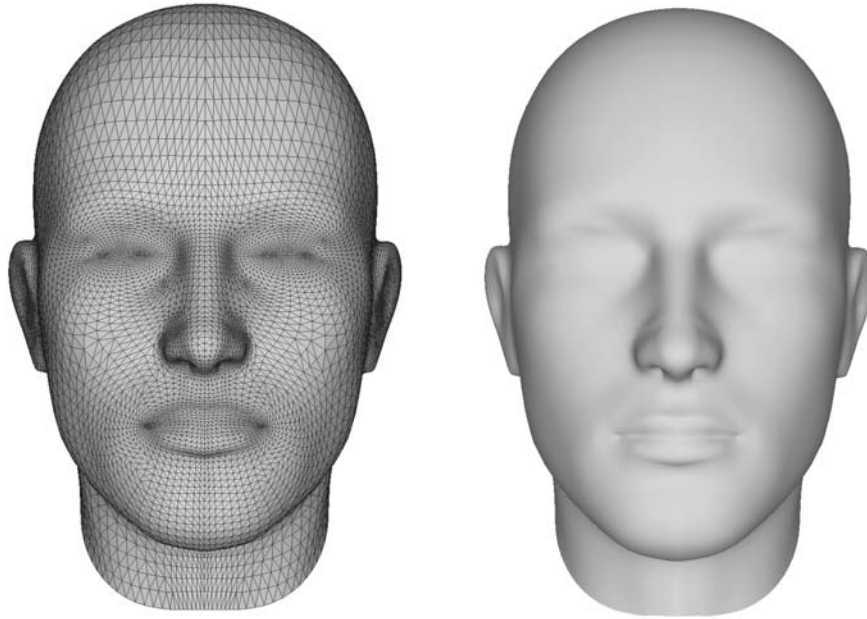


Abbildung 2: Triangulierung des Referenzkopfes und weiches Rendering

Bei einem Laserscan fährt der Scanner 360° um die Person herum und nimmt dabei Spaltenweise das Gesicht auf. Diese Spalten sind in den Aufnahmen meist noch zu erkennen (Abbildung 3).

Bei einem Scan können einige Bereiche des Gesichts beziehungsweise Kopfes nicht erfasst werden. Dazu gehören die Nasenlöcher, exakte Ohrmuschelverformungen und die Mitte der Schädeldecke. Das führt zu fehlenden oder ungenauen Bereichen in dem Ergebnis (Abbildung 3). Ein fertig gescannter Kopf ist eine Punktwolke und hat im Durchschnitt 80.000 gleichmäßig verteilte Vertices und 150.000 Dreiecke. Ein Vertex ist die Bezeichnung für einen dreidimensionalen Objektpunkt. Das Verhältnis von Vertices in Originaldaten und dem Referenzkopf liegt damit in etwa bei 9:1.

Da der Scanner keine Haare erfassen kann, mussten die Personen ihre Haare mit einer Kappe abdecken. Das führt zu einer Kante beim Übergang von Kappe zum Kopf, aber auch zu einer glatten Kopfoberfläche, die in späteren Anwendungszwecken wieder mit Haaren gefüllt werden kann.

Diese Originalscans eignen sich mit ihrer Topologie nicht für ein Formmodell, da sich ihre Vertexanzahlen um bis zu 20% unterscheiden und, wie in Abbildung 3 zu sehen ist, im Halsbereich nicht gleich weit aufgenommen sind. Aus diesem Grund werden alle Originaldaten an das Referenzmodell angepasst, damit alle Daten die gleiche Vertexnummerierung und Triangulierung haben, wobei die individuellen Gesichtszüge aber beibehalten werden. Diese Anpassung wird im folgenden Kapitel beschrieben.



Abbildung 3: Scans eines Mannes und einer Frau

3.3 Anpassung der Originalköpfe an den Referenzkopf

Ziel dieser Anpassung ist, dass alle Originalscans die gleiche Vertex- und Dreiecksanordnung erhalten, wie der Referenzkopf. Die Vertices des Referenzobjektes werden dabei so an die Form des Originalkopfes gezogen, dass kaum noch Unterschiede zur Form des Originalkopfes bestehen, obwohl weniger Vertices im Objekt sind.

Diese Verformung verläuft in 2 Schritten. Zuerst gibt es eine grobe Anpassung durch Thin Plate Splines. Diese sind in Abschnitt 2.2 beschrieben.

Dazu werden in der Referenz und im Original Landmarken gesetzt, die im jeweiligen Kopf das gleiche Merkmal bezeichnen (vgl. Abbildung 4(a)). Diese Landmarken dienen als Stützstellen der Thin Plate Splines. Man kann sich vorstellen, dass beide Köpfe übereinander gelegt werden und an diesen Merkmalspunkten der Referenzkopf auf die exakte Position des Originalkopfes gezogen werden. Alle anderen Vertices zwischen den Landmarken werden durch Splines interpoliert. Der Vorteil von TPS ist das Treffen der Stützstellen und die Glattheit der Kurven an nicht markierten Punkten. Der dadurch entstehende Kopf hat wenig Ähnlichkeit mit dem originalen Kopf (vgl. Abbildung 4(b)).

Deshalb werden im zweiten Schritt die individuellen Gesichter geformt. An den Landmarken sind die Positionen der Vertices von Referenz und Original identisch. Die restlichen Knoten des durch TPS verformten Modells müssen nun auf die exakten Koordinaten vom Originalkopf gezogen werden. Man stelle sich wieder das Übereinanderlegen beider Modelle vor. Aus jedem Vertex des Referenzkopfes wird ein Strahl in die Normalenrichtung berechnet und der dichteste Schnittpunkt mit dem Originalkopf als die neue Position für den Vertex genommen. Liegt der dichteste Punkt oberhalb eines festgelegten Abstandes, gilt er als nicht getroffen und muss im Nachhinein korrigiert werden.

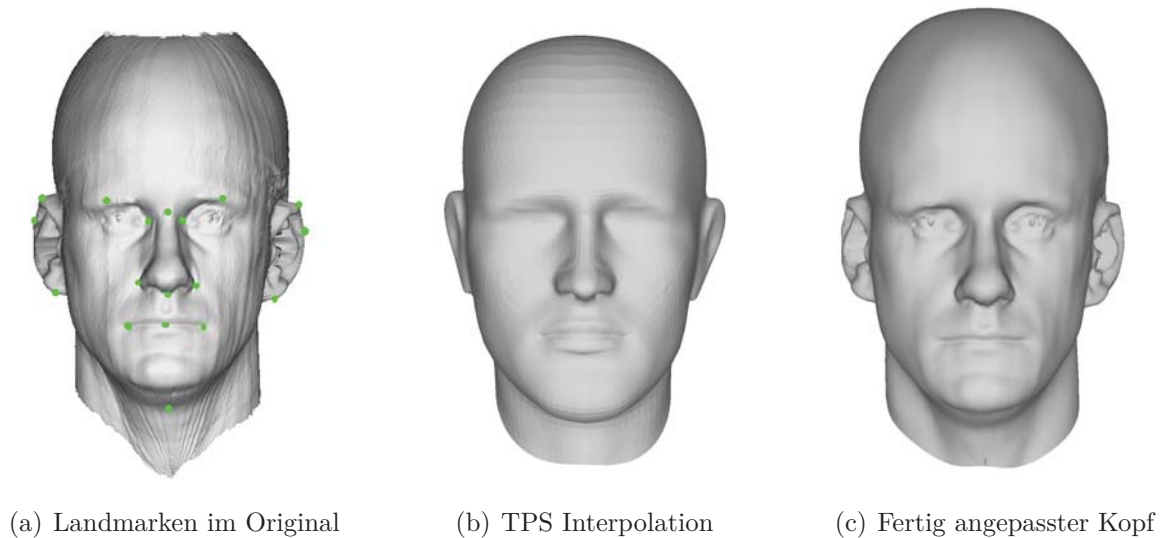


Abbildung 4: Schritte der Anpassung eines Originalscans an den Modellkopf

Das Ergebnis ist ein mit dem Original nahezu identischer Kopf (Abbildung 4(c)). Dieser Kopf ist deutlich weniger verrauscht, da die Knotenanzahl stark verringert und ausreißende Punkte entfernt wurden.

Probleme gibt es bei den in Abschnitt 3.2 erwähnten Löchern in der Schädeldecke. Die Scans unterscheiden sich weiterhin stark in der Länge des Halses. An diesen Stellen trifft der Strahlenschnitt nicht, beziehungsweise liegt oberhalb des Abstandsschwellwertes. Um trotzdem vollständig modellierte Köpfe zu erhalten, werden die fehlenden Vertexpositionen ebenfalls mit Thin Plate Splines interpoliert. Dafür bilden die Nachbarschaftsringe der Löcher die Stützstellen und die kaputten Regionen werden interpoliert. Da die fehlenden Positionen entweder vereinzelt auftreten oder als Regionen in glatten Bereichen wie Hals oder Schädeldecke liegen, ist das Ergebnis eine glatte und realistische Modellierung (vgl. Abschnitt 2.2).

Die neumodellierten Dreiecksnetze besitzen nun die gleiche Topologie, das heißt, die Anzahl der Vertices und Dreiecke mit ihren Nachbarschaften innerhalb des Kopfes sind identisch. Diese können jetzt für die Berechnung des Formmodells verwendet werden.

3.4 Aufbau des statistischen 3D Formmodells

Nachdem nun die Originalscans im vorherigen Abschnitt in eine einheitliche Topologie überführt wurden, kann mit der Berechnung des Formmodells begonnen werden. Mathematisch gesehen ist ein Formmodell die Darstellung eines Objektes durch die Eigenvektoren der Kovarianzmatrix der Objektklasse. Dadurch werden die Daten de-korreliert, sodass in wenigen Eigenvektoren bereits ein Großteil aller Informationen steckt. Gleichung 9 zeigt die Umrechnung eines Datums in ein gewichtetes Formmodell.

Wendet man jedoch die PCA auf ein komplexes Modell wie das Gesicht an, ist festzustellen, dass wenige Hauptkomponenten die Variationen der verschiedenen Körperteile nur schlecht darstellen können. Für eine gute Rekonstruktion eines bestimmten Kopfes braucht man sehr viele Hauptkomponenten, bis eine Ähnlichkeit mit dem Originalkopf besteht.

Aus diesem Grund werden komplexe Formen häufig in kleinere Regionen unterteilt und für jede Region eine eigene PCA berechnet. Dadurch reichen pro Region wieder wenige Eigenvektoren, um ein Datum darzustellen. Blanz und Vetter [BV99] schlagen die Aufteilung des Kopfes in vier Regionen - Augen, Nase, Mund und Umgebung - vor. In [SE09] wird eine verfeinerte Regionenaufteilung mit zusätzlichen vier Regionen vorgeschlagen - Ohren, Kinn, Wangen und Hals.

In dieser Arbeit wird eine Aufteilung ähnlich des zweiten Ansatzes bevorzugt, die jedoch einen natürlicheren Übergang an den Regionengrenzen bei Verformungen bildet. Dafür wird die Halsregion auf Kinn, Wangen und restlichen Schädel aufgeteilt, sodass nur noch sieben Regionen vorhanden sind (siehe Abbildung 5).

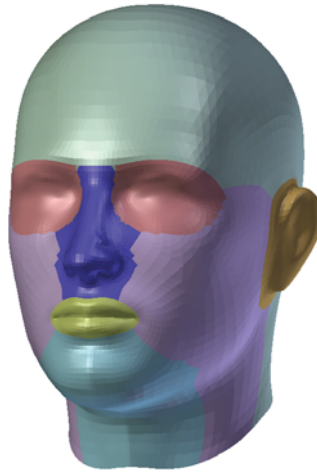


Abbildung 5: Unterteilung des Referenzkopfes in sieben Regionen für die jeweiligen PCA Modelle.

Eine Region \mathcal{R}_j mit $j = 1, \dots, K$ und K ist die Anzahl der Regionen, ist ein Index-Set von Vertices, wobei $i \in \mathcal{R}_j$, wenn ein Vertex \mathbf{p}_i zur Region \mathcal{R}_j gehört. Es sei bemerkt, dass $\mathcal{R}_i \cap \mathcal{R}_j \neq \emptyset$, da angrenzende Regionen sich um mindestens einen Vertex überlappen.

Nun bekommt jede einzelne Region ihre eigene PCA-Matrix $\mathbf{H}^{(j)}$ (siehe Gleichung (9)). Dafür bildet jeder Datensatz jeweils einen Vektor pro Region \mathcal{R}_j :

$$\mathbf{b}_i^j = (x_1^j \ y_1^j \ z_1^j \ \dots \ x_B^j \ y_B^j \ z_B^j)^T, \quad (18)$$

wobei B die Anzahl der Vertices in der Region ist. Die Vertices aller Dreiecksnetze pro Region bilden eine Datenmatrix $\mathbf{B}_j = [\mathbf{b}_1^j \dots \mathbf{b}_M^j]$, M ist die Anzahl der Netze in der Datenbank.

Mit dieser Matrix werden entsprechend der reduzierten PCA aus Abschnitt 2.1 (vgl. [TP91]) die Eigenvektoren und Eigenwerte berechnet. Die PCA-Matrix $\mathbf{H}^{(j)}$ einer jeweiligen Region \mathcal{R}_j wird mit

$$\mathbf{H}^{(j)} = (\mathbf{p}_{ik}^{(j)}) = \begin{cases} ev_{jk}, & \text{falls Vertex in Region } \mathcal{R}_j \\ 0, & \text{sonst} \end{cases} \quad i = 1, \dots, 3N, k = 1, \dots, M \quad (19)$$

beschrieben und hat die Größe $3N \times M$. Hier ist ev_{jk} der Eigenvektor für Region j an Spalte k . Damit die Matrizen eine einheitliche Zeilenanzahl von $3N$ haben, werden alle Vertices mitgelistet, wobei sich jedoch nur an den Vertices der Region ein Eintrag ungleich Null befindet.

Die Gesamt-PCA Matrix aller Regionen ist die Konkatenation der Matrizen $\mathbf{H}^{(j)}$ für die einzelnen Regionen $\mathbf{H} = [\mathbf{H}_1 \dots \mathbf{H}_K]$.

Ein Problem bei Multi-PCA Modellen von Regionen in einem Gesamtmodell sind die Regionengrenzen. Werden die Eigenvektoren der einzelnen Regionen gewichtet, strebt jede Region in ihre eigene Richtung und es entstehen unschöne, gut sichtbare Regionengrenzen. Um dieses zu verhindern, wird ein Überblendungsalgorithmus entwickelt, der die Regionen an den überlappenden Vertices übereinander zieht.

Wie oben beschrieben, befinden sich in der PCA-Matrix nur Einträge an den Elementen $(3n, j)$, an denen ein Vertex n zu einer Region j gehört. Alle anderen Matrixelemente sind Null. Für das Überblenden werden die einzelnen Regionen ringweise vergrößert, sodass sie mit ihren angrenzenden Regionen überlappen. Nun werden für die einzelnen, vergrößerten Regionen wieder die Hauptkomponenten berechnet und die PCA-Matrix neu erstellt. Alle Vertices, die zu mehr als einer Region gehören, werden durch diese Matrix aber mehrfach gewichtet, wenn das Formmodell mit Gleichung 9 rekonstruiert wird.

Um das zu verhindern, wird die PCA-Matrix an den Elementen für die Vertices der überlappenden Bereiche anders gewichtet. Alle Elemente der ursprünglichen Region werden weiterhin mit 1 gewichtet, alle anderen mit 0. Die Einträge der Matrix der erweiterten Regionen bekommen ein Gewicht abhängig von der Entfernung zu der originalen Region. Hier wurde als Maßstab die Anzahl der Iterationen bzw. Ringe um die alte Region verwendet: $w_{neu} = \frac{1}{\text{Anzahl der Iterationen} + 1}$. Jede Region wird einzeln vergrößert und das neue Gewicht auf das ursprüngliche Gewicht (1 oder 0) addiert.

Abbildung 6 soll das verdeutlichen. Es gibt in diesem Beispiel 2 Regionen, mit blau und rot gekennzeichnet. Die Gewichte der PCA-Matrix sind an den Zeilen der Vertices der Originalregion jeweils 1. Wird nun die blaue Region um einen Ring vergrößert, erhalten die Zeilen der neuen Vertices für die blaue Region die Gewichte 0,5. Der zweite

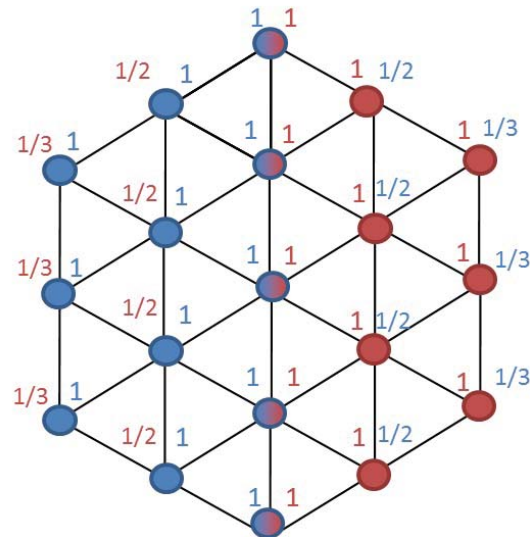


Abbildung 6: Zeigt die Gewichte der Überblendung bei zwei Regionen und Überblendungsgröße zwei.

Ring bekommt das Gewicht $\frac{1}{3}$ und so weiter. Das gleiche passiert mit den Gewichten für die rote Region. Anschließend ist die Summe der Gewichte der einzelnen Vertices einer Region größer als 1. Daher werden alle Gewichte noch auf 1 normiert.

Beispielhaft bedeutet das für Vertices an den Regionengrenzen: eine Region würde den Vertex an Position 1 ziehen und eine weitere an Position 2. Durch die Überblendung, wird der Vertex nun auf eine Position mittig beider Regionen gesetzt. Dadurch werden glattere Regionenübergänge erzeugt. Beispiele dafür sind in Abbildung 7 zu sehen.

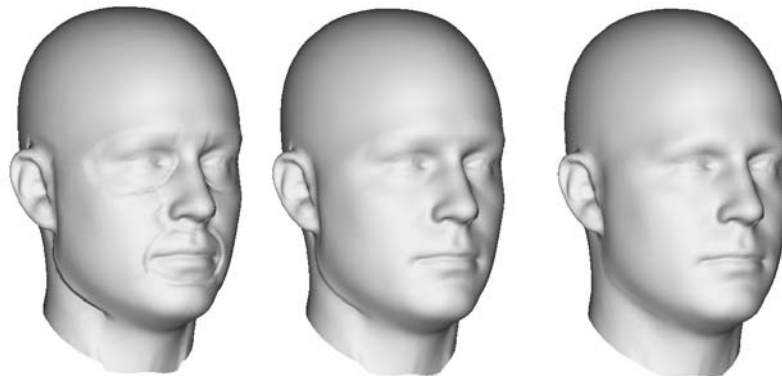


Abbildung 7: Zeigt das Formmodell mit gewichteten Eigenvektoren ohne Überblendung (links), mit Überblendungsgröße 1 (mitte) und 2 (rechts).

4 ICP-basierte Modellierung der 2D-Bilder

Das Ziel dieser Arbeit ist, aus einer Profilaufnahme und einem Frontalbild einer Person ein möglichst ähnliches und realistisches dreidimensionales Modell dieses Menschen zu erstellen. Im Weiteren wird auch der Bezeichnung 2D zu 3D Registrierung - für ein an Fotos angepasstes Formmodell - oder 3D zu 3D Registrierung - für ein an 3D Daten angepasstes Formmodell - verwendet.

Angelehnt an die 3D zu 3D Registrierung von Schneider und Eisert [SE09] wurden zwei Ansätze entwickelt, aus zwei Fotos ein Modell zu berechnen. In beiden Ansätzen spielt die in [SE09] entwickelte Kostenfunktion (siehe Abschnitt 4.3.1) für die Modellregistrierung eine zentrale Rolle. Beide arbeiten auf den gleichen Datensätzen und Landmarken, siehe Abschnitt 4.1 und 4.2.

Die Registrierung erfolgt im Wesentlichen in zwei Schritten: die Anpassung des Modells an die Profillinie der Person und daraufhin die Verformung an den Landmarken der Frontalaufnahme. Es wird angenommen, dass die annotierten Landmarken im Profil in drei und die des Frontbildes in zwei Koordinaten den Zielwerten des Modells entsprechen.

Beide Ansätze unterscheiden sich im zweiten Schritt, bei der Anpassung an die Frontalaufnahme.

Im ersten Ansatz (Abschnitt 4.4.2) werden die markierten Profil- und Frontallandmarken verwendet, um die ICP mit Formparametern zu lösen. Da es keine Information über die Tiefe der Frontallandmarken gibt, wird die Gleichung für diese Koordinate ausgespart und die Tiefeninformation nur durch die Profillandmarken berechnet.

Der zweite Ansatz (Abschnitt 4.4.3) hingegen bildet erst ein dreidimensional verformtes Modell durch Thin Plate Splines und wendet dann iterativ die Kostenfunktion aus [SE09] an.

4.1 Datensätze

Die Portraitbilder, die Ausgangspunkt für die Modellierung sind, wurden im Fraunhofer Heinrich Hertz Institut aufgenommen. Die Fotos haben eine Auflösung von 768 x 1024 Pixeln.

Besondere Anforderungen an die Fotos bezüglich Kamerakalibrierung oder Hintergrund gibt es nicht, solange die Person in Frontal- und Profilbild annähernd den gleichen Abstand zur Kamera hat, sodass der Kopf auf beiden Bildern in etwa die gleiche Größe besitzt. Beide Fotos sollten weiterhin die gleichen Abmessungen haben.

Von den so aufgenommenen Bildern werden für die Modellierung die Profil- und die Frontalansicht verwendet (siehe Abbildung 8).



Abbildung 8: Profil- und Frontalansicht als Beispiel der für die Modellierung verwendeten Fotos mit Abbildung der Koordinatenachsen

4.2 Profillinie und Landmarken

Damit das Formmodell Ähnlichkeit mit einer Person auf einem Bild aufweist, muss es an bestimmten markanten Gesichtspunkten und dem Profil mit der Abbildung übereinstimmen.

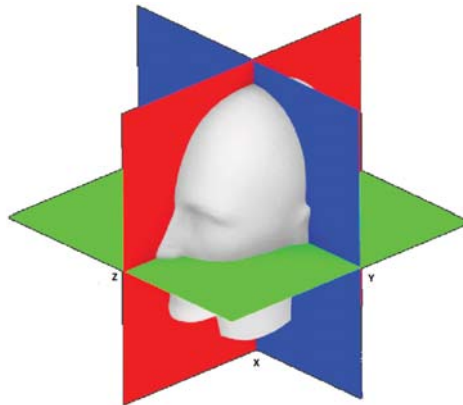


Abbildung 9: Die Ausrichtung des Formmodells an den Ebenen: die XY-Ebene ist blau dargestellt, die YZ-Ebene grün und die XZ-Ebene rot.

Die Silhouette der Person wird manuell im Profilbild markiert. Dafür wird der Kopf einmal umrandet, wobei die Haare möglichst ausgespart werden, da sie durch das Formmodell nicht modelliert werden können. Die einzelnen Punkte der markierten Linie besitzen zwei ablesbare Pixelkoordinaten. Für die dritte Koordinate wird angenommen, dass die Person Achsenparallel ausgerichtet ist und die Sagittalebene damit parallel zur dritten Koordinatenachse liegt. Dadurch kann jeder markierte 2D-Punkt um die

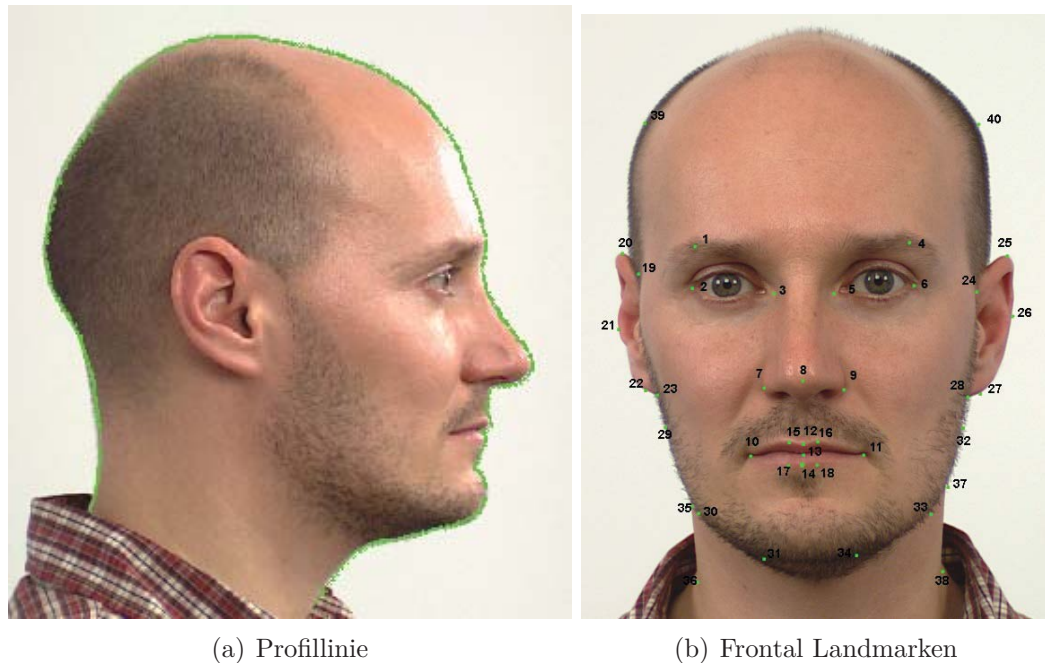


Abbildung 10: Markierte Landmarken in den Abbildungen

gleiche dritte Koordinate erweitert werden. Der Einfachheit halber wurde dafür die Position 0 verwendet. Abbildung 10(a) zeigt eine solche Profillinie. Die Ausrichtung des Formmodells und damit die Ebenen der Fotoaufnahmen ist in Abbildung 9 dargestellt.

1	Augenbraue rechts	2	Auge außen rechts	3	Auge innen rechts
4	Linke Augenbraue	5	Linkes Auge außen	6	Linkes Auge innen
7	Nasenflügel rechts	8	Nasenspitze	9	Linker Nasenflügel
10	Mundwinkel rechts	11	Linker Mundwinkel	12	Mitte Oberlippe
13	Lippenmitte	14	Mitte Unterlippe	15	Philtrumsrand rechts
16	Philtrumsrand links	17	Unterlippenform r.	18	Unterlippenform l.
19	Ohransatz rechts	20	Ohrspitze rechts	21	Ohrmitte rechts
22	Ohrläppchen rechts	23	Ohrläppchenansatz r.	24	Ohransatz links
25	Ohrspitze links	26	Ohrmitte links	27	Ohrläppchen links
28	Ohrläppchenansatz l.	29	Kiefer rechts	30	Kiefermitte rechts
31	Kinn rechts	32	Kiefer links	33	Kiefermitte links
34	Kinn links	35	Halsansatz rechts	36	Halsform rechts
37	Halsansatz links	38	Halsform links	39	Kopfform rechts
40	Kopfform links				

Tabelle 1: Die Tabelle benennt die in Abbildung 10(b) markierten Landmarken.

Die entwickelte Benutzeroberfläche besitzt eine Markierfunktion für die Landmarken. Beliebige andere Bildbearbeitungsprogramme, aus denen Pixelpositionen ausgelesen werden können, können ebenfalls verwendet werden.

Im Frontalbild werden nur einige wichtige Punkte markiert. Diese Punkte sind für den Wiedererkennungswert der Person auf dem Foto besonders wichtig, da sie die individuellen Gesichtszüge erkennen lassen. Die einzelnen Marken sind in der Tabelle 1

beschrieben. Auch hier können anhand der Pixelposition im Bild zwei von drei Koordinaten ausgelesen werden. Die dritte Koordinate ist nun jedoch unbekannt und muss berechnet werden (siehe Abschnitt 4.4.2 und 4.4.3). Die Landmarken und ihre Positionen sind in Abbildung 10(b) zu sehen.

Für die weitere Verwendung der Landmarken müssen die Positionen im gleichen Koordinatensystem wie das Modell vorliegen. Das heißt, die Pixelpositionen müssen im Intervall $[0, 1]$ liegen und auf die Nasenspitze zentriert werden. Dafür werden die Werte durch die längere der Seite der Bilder geteilt, sodass die Kopfhöhe auf das Intervall $[0, 1]$ skaliert ist. Kopfbreite und -länge liegen damit in einem kleineren Intervall. Die Nasenspitze wurde verwendet, weil sie im Gegensatz zu vielen anderen Punkten sowohl auf der Profil- als auch der Frontalaufnahme deutlich zu erkennen ist, sie in der Profillinie markiert wird und bei jedem Menschen klar auszumachen ist.

4.3 3D-3D Registrierung

4.3.1 Kostenfunktion der 3D Objektregistrierung

Die klassische ICP, wie in Abschnitt 2.3 beschrieben, passt ein Referenzmodell in Rotation, Skalierung und Translation an ein Zielmodell an. Das so registrierte Modell hat jedoch noch keine individuellen Formen, sondern stimmt mit dem Formmodell aus Abschnitt 3.4 überein.

Daher muss die Kostenfunktion erweitert werden, sodass eine Formänderung des Modells ermöglicht wird. Schneider [SE09][SE10] erweitert die ICP um sogenannte Formparameter, die bei einem durch PCA erstellten Formmodell der Gewichtung der Eigenvektoren entsprechen (Gleichung 9).

Da auch in dieser Arbeit die Referenz mit Hilfe eines statistischen Formmodells berechnet wurde, kann ein Vertex \mathbf{p}_i des Modells auch beschrieben werden als:

$$\mathbf{p}_i = \mathbf{M}_i \mathbf{m} + \boldsymbol{\mu}_i \quad (5) \quad (20)$$

Diese Gleichung kann folgendermaßen aus Sicht des statistischen Formmodells interpretiert werden: \mathbf{M}_i ($3 \times \text{Anzahl Formparameter}$) ist der Teil der PCA Regionenmodellmatrix \mathbf{H}^j , zu dem der Vertex \mathbf{p}_i (3×1) gehört (vgl. Gleichung 19), $\boldsymbol{\mu}_i$ (3×1) ist der Regionenteil des Mittelwertvektors und \mathbf{m} ($1 \times m$) ist der Gewichtungsvektor, der die Deformation der Region beschreibt. Die Formmatrix \mathbf{M}_i eines Vertex \mathbf{p}_i ist nur an den Stellen der Regionen ungleich Null, zu denen der Vertex auch gehört. Gehört beispielsweise ein Vertex zu Region zwei und fünf und es gibt pro Region drei Parameter und insgesamt sieben Regionen, sieht die Matrix \mathbf{M}_i wie folgt aus:

⁵vgl. Gleichung 9

$$\mathbf{M}_i = \left[\begin{array}{c|c|c|c|c|c|c} 1 & 2 & 3 & 4 & 5 & 6 & 7 \\ \hline 0 & 0 & 0 & \mathbf{a}_3 & \mathbf{a}_4 & \mathbf{a}_5 & 0 & 0 & 0 & 0 & 0 & 0 & \mathbf{a}_{12} & \mathbf{a}_{13} & \mathbf{a}_{14} & 0 & 0 & 0 & 0 & 0 & 0 \end{array} \right]. \quad (21)$$

Wird die Gleichung 16 um den Formparametervektor m erweitert, erhält man die Kostenfunktion, deren Minimierung auch die Form des Referenzmodells ändern kann:

$$e(s, \boldsymbol{\theta}, \mathbf{m}, \mathbf{t}) = \sum_i ||\text{rot}(\boldsymbol{\theta})(\mathbf{M}_i \mathbf{m} + \boldsymbol{\mu}_i) + \mathbf{t} - \mathbf{q}_i||^2, \quad (22)$$

wobei $\text{rot}(\boldsymbol{\theta})$ die Eulerrepräsentation der Rotationsmatrix mit den Eulerwinkeln im Argument ist und \mathbf{q}_i der Zielpunkt des Referenzpunktes \mathbf{p}_i .

Im Gegensatz zur ICP aus Abschnitt 2.3, welche durch Singulärwertzerlegung gelöst werden kann, kann die um Formparameter erweiterte Gleichung nicht in dieser Form ausgerechnet werden. Daher ist hier die iterative Lösung mit der Annäherung durch die erste Ableitung die zu bevorzugende Lösung. Da nur eine sehr kleine Änderung der Parameter angenommen wird, kann diese Gleichung unter Annahme der Linearität durch Approximation der ersten Ableitung gelöst werden.

Die Gleichung für den Fehler bei kleinen Änderungen $(\Delta s, \Delta \boldsymbol{\theta}, \Delta \mathbf{m}, \Delta \mathbf{t})$ der Parameter $(s, \boldsymbol{\theta}, \mathbf{m}, \mathbf{t})$ lautet:

$$\begin{aligned} e((1 + \Delta s)s, \text{rot}(\Delta \boldsymbol{\theta})\text{rot}(\boldsymbol{\theta}), \mathbf{m} + \Delta \mathbf{m}, \mathbf{t} + \Delta \mathbf{t}) = \\ \sum_i ||(1 + \Delta s)s \cdot \text{rot}(\Delta \boldsymbol{\theta})\text{rot}(\boldsymbol{\theta}) \\ (\mathbf{M}_i(\mathbf{m} + \Delta \mathbf{m}) + \boldsymbol{\mu}_i) + \mathbf{t} + \Delta \mathbf{t} - \mathbf{q}_i||^2. \end{aligned} \quad (23)$$

Die Skalierung ändert sich prozentual, eine Rotationsänderung mit Eulerwinkeln erfolgt durch Multiplikation der Änderung mit der bisherigen Rotationsmatrix und die Änderung der Formparameter und der Translation erfolgt durch einfaches Addieren der Änderungen.

Als erstes wird die Linearisierung für die Rotation gebildet. Eine infinitesimale Rotation, die nur wenig von der Einheitsmatrix \mathbf{I}_3 abweicht, kann als $\text{rot}(\boldsymbol{\theta}) \approx \mathbf{I}_3 + \Delta \boldsymbol{\Theta}$, mit einer Matrix $\Delta \boldsymbol{\Theta}$ die eine kleine Drehung beschreibt, notiert werden. $\Delta \boldsymbol{\Theta}$ ist die Ableitung der Rotationsmatrix an der Stelle der Identität, also die Erzeugende einer infinitesimalen Rotation um eine beliebige Achse. Sie lautet:

$$\Delta \boldsymbol{\Theta} = \begin{bmatrix} 0 & -\theta_z & \theta_y \\ \theta_z & 0 & -\theta_x \\ -\theta_y & \theta_x & 0 \end{bmatrix} \quad (24)$$

$$\text{rot}(\boldsymbol{\theta}) \approx \mathbf{I}_3 + \Delta \boldsymbol{\Theta}. \quad (25)$$

Um diese Rotation nach den Winkeln zu lösen, wird ein Basiswechsel mit einem rotierten Punkt $p = [x \ y \ z]^T$ durchgeführt:

$$\text{rot}(\boldsymbol{\theta})\mathbf{p} \approx \mathbf{p} + \Delta\boldsymbol{\Theta}\mathbf{p} \quad (26)$$

$$= \mathbf{p} + \begin{bmatrix} 0 & z & -y \\ -z & 0 & x \\ y & -x & 0 \end{bmatrix} \begin{bmatrix} \theta_x \\ \theta_y \\ \theta_z \end{bmatrix} \quad (27)$$

$$:= \mathbf{p} + [\mathbf{p}]_{\times}\boldsymbol{\theta} \quad (28)$$

Die Schreibweise $[\mathbf{p}]_{\times}\boldsymbol{\theta}$ bezeichnet das Kreuzprodukt zweier Vektoren.

Wird angenommen, dass die anderen Parameter von Gleichung 23 ebenfalls nur kleine Änderungen haben, kann die Linearisierung dieser mit Hilfe der ersten beiden Terme der Taylorreihe berechnet werden, da Terme höherer Ordnung bei kleinen Argumenten gegen Null konvergieren und vernachlässigt werden können:

$$f(x + \Delta x) = f(x) + f'(x)(\Delta x). \quad (29)$$

Da die hier verwendete Fehlerfunktion (Gleichung 23) mehrdimensional ist, erweitert sich die Funktion auf Grundlage der Taylorreihe um die partiellen Ableitungen von f der einzelnen Parameter auf:

$$f(x + \Delta x, y + \Delta y, \dots) = f(x, y) + f_x(x, y)(\Delta x) + f_y(x, y)(\Delta y) + \dots \quad (30)$$

Verwendet man für die Rotationsänderung die Substitution aus Gleichung 25, kann man die Multiplikation in eine Addition überführen:

$$\text{rot}(\Delta\boldsymbol{\theta})\text{rot}(\boldsymbol{\theta}) \approx (\mathbf{I}_3 + \Delta\boldsymbol{\Theta})\text{rot}(\boldsymbol{\theta}). \quad (31)$$

und daraufhin Gleichung 30 anwenden. Damit lauten die Terme für die einzelnen Parameteränderungen aus Gleichung 23:

$$e(s, \boldsymbol{\theta}, \mathbf{m}, \mathbf{t}) = s \operatorname{rot}(\boldsymbol{\theta})(\mathbf{M}_i \mathbf{m} + \boldsymbol{\mu}_i) + \mathbf{t} - \mathbf{q}_i \quad (32)$$

$$\begin{aligned} (\Delta s + 1)s &\mapsto e_s(s, \boldsymbol{\theta}, \mathbf{m}, \mathbf{t})(s\Delta s) \\ &= \operatorname{rot}(\boldsymbol{\theta})(\mathbf{M}_i \mathbf{m} + \boldsymbol{\mu}_i)s\Delta s \end{aligned} \quad (33)$$

$$\begin{aligned} (\mathbf{I}_3 + \Delta\boldsymbol{\Theta})\operatorname{rot}(\boldsymbol{\theta}) &\mapsto e_{\operatorname{rot}\boldsymbol{\theta}}(s, \boldsymbol{\theta}, \mathbf{m}, \mathbf{t})(\Delta\boldsymbol{\Theta}\operatorname{rot}(\boldsymbol{\theta})) \\ &= s\operatorname{rot}(\boldsymbol{\theta})(\mathbf{M}_i \mathbf{m} + \boldsymbol{\mu}_i)\Delta\boldsymbol{\Theta}\operatorname{rot}(\boldsymbol{\theta}) \end{aligned} \quad (34)$$

$$\begin{aligned} \mathbf{m} + \Delta\mathbf{m} &\mapsto e_{\mathbf{m}}(s, \boldsymbol{\theta}, \mathbf{m}, \mathbf{t})(\Delta\mathbf{m}) \\ &= s \operatorname{rot}(\boldsymbol{\theta})\mathbf{M}_i(\Delta\mathbf{m}) \end{aligned} \quad (35)$$

$$\begin{aligned} \mathbf{t} + \Delta\mathbf{t} &\mapsto e_{\mathbf{t}}(s, \boldsymbol{\theta}, \mathbf{m}, \mathbf{t})(\Delta\mathbf{t}) \\ &= \Delta\mathbf{t} \end{aligned} \quad (36)$$

Die Taylorerweiterung von e für die Parameteränderungen ergibt sich aus dem Zusammensetzen der Terme:

$$e = \sum_i \|\tilde{\mathbf{p}}_i + \mathbf{t} + \tilde{\mathbf{p}}_i \Delta s + \Delta\boldsymbol{\Theta}\tilde{\mathbf{p}}_i + s \operatorname{rot}(\boldsymbol{\theta})\mathbf{M}_i \Delta\mathbf{m} + \Delta\mathbf{t} - \mathbf{q}_i\|^2 \quad (37)$$

mit

$$\tilde{\mathbf{p}}_i = s \operatorname{rot}(\boldsymbol{\theta})(\mathbf{M}_i \mathbf{m} + \boldsymbol{\mu}_i). \quad (38)$$

Diese Gleichung stellt die Linearisierung der Gleichung unter der Annahme sehr kleiner Änderungen dar. Somit kann sie durch einen Gradientenabstieg (genauer: Gauss-Newton-Optimierung) gelöst werden.

Aus dieser Gleichung kann ein Lineares Gleichungssystem aufgestellt werden, das nach den Parametern gelöst wird und die Kostenfunktion minimiert. Dafür wird obige Gleichung umgestellt und Gleichung 28 zur Substitution bei der Rotation verwendet:

$$\begin{bmatrix} \tilde{\mathbf{p}}_1 & [\tilde{\mathbf{p}}_1]_{\times} & s \operatorname{rot}(\boldsymbol{\theta})\mathbf{M}_1 & \mathbf{I}_3 \\ \vdots & \vdots & \vdots & \vdots \\ \tilde{\mathbf{p}}_N & [\tilde{\mathbf{p}}_N]_{\times} & s \operatorname{rot}(\boldsymbol{\theta})\mathbf{M}_N & \mathbf{I}_3 \end{bmatrix} \begin{bmatrix} \Delta s \\ \Delta\boldsymbol{\theta} \\ \Delta\mathbf{m} \\ \Delta\mathbf{t} \end{bmatrix} = \begin{bmatrix} \mathbf{q}_1 - \tilde{\mathbf{p}}_1 - \mathbf{t} \\ \vdots \\ \mathbf{q}_N - \tilde{\mathbf{p}}_N - \mathbf{t} \end{bmatrix} \quad (39)$$

Dieses Gleichungssystem bietet die Grundlage für die Berechnung der Parameteränderungen, die für die Registrierung des Modells verwendet werden. Bei der 3D zu 3D Registrierung kann dieses System direkt verwendet werden, da sowohl alle Zielvertices als auch die Referenzvertices des gesamten Modells in jeder Iteration bekannt sind. Bei der 2D zu 3D Registrierung sind als Zielvertices nur diejenigen Vertices bekannt, zu denen es Landmarken im Foto gibt. Werden viele Formparameter verwendet, kann es daher sein, dass das Gleichungssystem unterbestimmt ist. Eine Möglichkeit zur Behe-

bung dieses Problems können die in Abschnitt 4.3.2 formulierten Nebenbedingungen sein.

4.3.2 Nebenbedingungen der Kostenfunktion

Bei Anwendung der Kostenfunktion für das verwendete Regionenmodell treten zwei Probleme auf.

Das erste Problem hat seine Ursache in den Regionen. Der Vorteil, den die Aufteilung des Modells in mehrere einzeln verformbare Regionen ist eine speziellere und vielfältigere Anpassung der einzelnen Körperteile an den vorgegebenen Kopf. Das bringt jedoch den Nachteil mit, dass zwar innerhalb einer Region eine gute Näherung an das Zielobjekt erreicht wird, die Regionengrenzen jedoch nicht zwangsläufig glatte Übergänge bilden. Die Ursache liegt in der Minimierung des Fehlers für jede Region durch die Fehlerfunktion. Einige Vertices des Modells liegen in zwei oder drei Regionen, bilden sozusagen die Grenze zwischen den Regionen. Diese Vertices werden durch die Formparameter für die verschiedenen Regionen an unterschiedliche Positionen gezogen. Die Bildung eines einfachen Mittelwertes für diese Vertices umgeht das Problem nur bei sehr geringen Grenzunebenheiten. Stärkere, zum Teil deutlich sichtbaren Regionenübergänge, werden mit der ersten Nebenbedingung verhindert.

Dafür wird das Gleichungssystem 39 um Gleichungen erweitert, die die Eigenschaft haben, die Regionen an überlappenden Vertices zusammenzuhalten. Der Unterschied zwischen den Positionen, an den ein Vertex durch verschiedene Regionenverformungen erhält, soll so gering wie möglich gehalten werden. Gehört also ein Vertex \mathbf{p}_i zu zwei Regionen R_j und R_k , dann ist die Formmatrix \mathbf{M}_i (siehe Gleichung 20) dieses Vertex' an genau zwei Regionenblöcken $\mathbf{P}_i^{(j)}$ und $\mathbf{P}_i^{(k)}$ ungleich Null. Der Regionenübergang für Vertex i ist glatt, wenn der Abstand der zwei Regionenformen gleich Null ist, also:

$$[\dots \mathbf{P}_i^{(j)} \dots - \mathbf{P}_i^{(k)} \dots] \mathbf{m} =: \mathbf{\Gamma}_i \mathbf{m} = 0. \quad (40)$$

Diese Gleichung kann für jeden Vertex aufgestellt werden, der in zwei Regionen liegt. Vertices, die zu mehr als zwei Regionen gehören, werden ignoriert, da sie einen unerheblichen Teil der Gesamtvertices ausmachen und die Regionengrenzen auch ohne Berücksichtigung dieser geglättet werden.

Diese erste Bedingung wird bei [SE09] "Smoothness Prior" $\mathbf{\Gamma}$ genannt und wird definiert als:

$$\mathbf{\Gamma} = \begin{bmatrix} \mathbf{\Gamma}_1 \\ \vdots \\ \mathbf{\Gamma}_L \end{bmatrix} \quad (41)$$

wobei L die Anzahl der Vertices ist, die zu zwei Regionen gehören. Der Smoothness Prior hat nur Auswirkungen auf die Formparameter \mathbf{m} , nicht auf die Transformationen.

Das zweite Problem ist die Stärke der Verformungen. Dieses Problem tritt weniger bei der 3D-3D Registrierung auf, bei der viele Vertices eine Region beschreiben, sondern eher bei der 2D-3D Registrierung, da eine Region nur durch wenige Landmarken beschrieben wird, das Gleichungssystem der Kostenfunktion 39 bei manchen Regionen unterbestimmt ist und das PCA-Modell daher große Freiheiten bei der Verformung hat.

Wie bei der Nebenbedingung zuvor hat auch diese Beschränkung nur Einfluss auf die Formparameter, indem jeder Parameter ein Gewicht bekommt, das verhindert, dass einzelne Hauptkomponenten ausreißen und das Modell deformiert wird. Diese Bedingung wird Verformungsstärke $\mathbf{\Lambda}$ genannt [SE09] und lautet:

$$\mathbf{\Lambda} = w\mathbf{I}_L, \quad (42)$$

wobei L die Anzahl der Formparameter ist, \mathbf{I} die Identitätsmatrix und w das Gewicht. Andere Gewichtungsschemata, zum Beispiel durch die inverse Wurzel der Eigenwerte der Hauptkomponenten, sind möglich ([SE09]). Sie führen in diesem Fall jedoch nicht zu einem besseren Ergebnis.

Wie bereits erwähnt, wird das Gleichungssystem 39 um die Nebenbedingungen $\mathbf{\Lambda}$ und $\mathbf{\Gamma}$ erweitert. Da deren Einfluss nur auf die Formparameter begrenzt ist, lautet das neue System:

$$\left[\begin{array}{ccc|c} \tilde{\mathbf{p}}_1 & [\tilde{\mathbf{p}}_1]_{\times} & \mathbf{I}_3 & s \operatorname{rot}(\boldsymbol{\theta})\mathbf{M}_1 \\ \vdots & \vdots & \vdots & \vdots \\ \tilde{\mathbf{p}}_N & [\tilde{\mathbf{p}}_N]_{\times} & \mathbf{I}_3 & s \operatorname{rot}(\boldsymbol{\theta})\mathbf{M}_N \\ \hline & & & \lambda_1 \mathbf{\Gamma} \\ & \mathbf{0} & & \lambda_2 \mathbf{\Lambda} \end{array} \right] =: \mathbf{K}. \quad (43)$$

Die Faktoren λ_1 und λ_2 regulieren den Einfluss der Nebenbedingungen. Für eine bessere Lesbarkeit wurden hier der Translations- und der Formterm getauscht.

4.3.3 Algorithmus der 3D-3D-Registrierung

Die einfachste Anwendung der oben vorgestellten ICP ist die Registrierung von Daten gleicher Dimensionen. In diesem Fall die Anpassung des Formmodells auf die originalen Köpfe. Formal stellen die Originale eine dichte Punktwolke $C \subset \mathbb{R}^3$ dar. In der Registrierung wird für jeden Punkt \mathbf{p}_i des Formmodells eine Korrespondenz zu einem Punkt der Punktwolke $\mathbf{q}_i \in C$ hergestellt und die Skalierung, Translation, Rotation und Form bestimmt, die die Kostenfunktion 39 aus Abschnitt 4.3.1 minimiert. Das Minimum kann allerdings nur lokal sein, sodass eine grobe Ausrichtung der Punktwolke mit dem Modell erfolgen sollte. Der Algorithmus besteht aus einer Iteration des Findens der Korrespondenzpunkte und des Minimierens der Kostenfunktion.

Die Anfangswerte der Parameter $s, \text{rot}(\boldsymbol{\theta}), \mathbf{m}$ und \mathbf{t} sind:

- $s \leftarrow 1$
- $\text{rot}(\boldsymbol{\theta}) \leftarrow \mathbf{I}_3$
- $\mathbf{m} \leftarrow \mathbf{0}$
- $\mathbf{t} \leftarrow \mathbf{0}$.

Die Rotation wird entsprechend Gleichung 14 durch eine Matrix und nicht durch einen Vektor von Winkeln beschrieben. Dadurch werden Probleme wie der Gimbal Lock vermieden. Die Schreibweise $\text{rot}\boldsymbol{\theta}$ wird aus Konsistenzgründen beibehalten.

Das Formmodell wird mit dem Mittelwert der Datenbank initialisiert:

- $\tilde{\mathbf{p}}_i \leftarrow \boldsymbol{\mu}_i$, mit $i = 1 \dots N$.

Es ist zu beachten, dass die Translation nicht mit einbezogen ist, $\tilde{\mathbf{p}}_i$ also die aktuelle Registrierung abzüglich der Translation ist. Denn, wie in Gleichung 39 zu sehen, wird die Translation extra behandelt und nicht mit den registrierten Punkten verrechnet.

Der Algorithmus besteht im Wesentlichen aus vier Schritten, die im Folgenden erläutert werden.

Schritt 1 Es werden die Punktkorrespondenzen gesucht. Die Punktwolke, in der die korrespondierenden Punkte gefunden werden sollen, hat idealerweise eine deutlich größere Punktdichte. Dafür wird zu jedem Punkt $\tilde{\mathbf{p}}_i + \mathbf{t}$ der dichteste Punkt \mathbf{q}_i der Zielpunktwolke C gesucht. Mit dichtest ist hier der kürzeste euklidische Abstand von zwei Punkten gemeint. Eine geeignete Datenstruktur hierfür ist zum Beispiel ein K-D-Baum.

Schritt 2 Sind alle Korrespondenzen gefunden, kann das Gleichungssystem der Kostenfunktion mit Daten gefüllt werden und die Parameteränderungen berechnet werden.

Schritt 3 Wurden die Änderungen berechnet, können die Parameter $s, \text{rot}(\boldsymbol{\theta}), \mathbf{m}$ und \mathbf{t} aktualisiert werden:

$$\begin{aligned}
 s &\leftarrow (1 + \Delta s) \cdot s \\
 \text{rot}(\boldsymbol{\theta}) &\leftarrow \text{rot}(\Delta\boldsymbol{\theta}) \cdot \text{rot}(\boldsymbol{\theta}) \\
 \mathbf{t} &\leftarrow \Delta\mathbf{t} + \mathbf{t} \\
 \mathbf{m} &\leftarrow \Delta\mathbf{m} + \mathbf{m}
 \end{aligned} \tag{44}$$

Schritt 4 Zum Schluss muss mit den neuen Parametern die Registrierung aktualisiert werden:

$$\tilde{\mathbf{p}}_i := s \text{rot}(\boldsymbol{\theta})(\mathbf{M}_i \mathbf{m} + \boldsymbol{\mu}_i) \text{ für } i = 1 \dots N. \tag{45}$$

Das Formmodell nähert sich in jedem Schritt an die Zielwolke an. Dadurch muss der Algorithmus so lange wiederholt werden, bis die gefundene Registrierung im mittleren quadratischen Fehler konstant bleibt:

$$e = \frac{1}{N} \sum_{i=1}^N \|\tilde{\mathbf{p}}_i + \mathbf{t} - \mathbf{q}_i\|^2 \quad (46)$$

Auch denkbar ist ein Schwellwert für den Fehler als Abbruchbedingung.

4.4 2D-3D Registrierung

4.4.1 Anpassung der Profillinie

Die Anpassung der Profillinie des Formmodell des Kopfes an die markierte Profillinie im Bild (siehe Abbildung 10(a)) ist in beiden Ansätzen (siehe 4.4.2 und 4.4.3) für die Bild-Modell-Registrierung der erste Schritt. Dieser dient insbesondere dazu, die Translation, Rotation und Skalierung des Modells halbwegs genau zu bestimmen, damit die Anpassung der Form besser gelingt.

Für eine ICP müssen die Positionen der Vertices des zu registrierenden Objektes, in diesem Fall das Formmodell, und die Positionen der Punkte des Zielwertes, die Profillinienpunkte des Bildes, bekannt sein.

Die 3D Punkte des Bildes werden wie in Abschnitt 4.2 bestimmt. Auf einem Foto können nur zweidimensionale Koordinaten abgelesen werden. Die Ebene $y = 0$ wird als Ebene der Profillinie angenommen (siehe Abbildung 9).

Die Vertex-Positionen der Profillinie des Dreiecksnetzes des Kopfes müssen ebenfalls bekannt sein. Dafür können in einem geeigneten Programm (z.B. Blender oder MeshLab) die Vertices der Linie mit zugehörigen Vertex-Indices markiert werden. Die Topologie des Objektes eignet sich gut dazu, da während der Erstellung eine Hälfte des Objektes an der Ebene der Profillinie gespiegelt wurde, wodurch diese gut erkennbar ist (siehe Abbildung 11). Sind die Indices bekannt, kann im späteren Programm die Position ausgelesen werden.

Sobald die Zielpunkte des Bildes und die Startwerte des Formmodells festgelegt sind, kann die Matrix für den ICP Algorithmus aufgestellt werden. Dazu müssen die Korrespondenzen von Ziel- und Modellpunkten ermittelt werden. Dies ist möglich, da in der Profillinie jeder Pixel des Bildes markiert ist, so kann wie bei der 3D-3D Registrierung anhand geeigneter Strukturen (z.B. K-d-Bäume) der nächste Nachbar gefunden werden. Die beiden Punktwolken müssen dafür grob zueinander ausgerichtet werden, damit nicht entfernt voneinander liegende Punkte einander zugewiesen werden. In bestimmten Fällen, wenn nämlich die Punktwolken sich nicht berühren, werden alle Modellpunkte einem Zielpunkt zugeordnet und dadurch wird die ICP nicht terminieren. Um solchen Fällen entgegen zu wirken, wird ein Schwellwert benutzt, der entfernt voneinander liegende Korrespondenzen aussortiert.

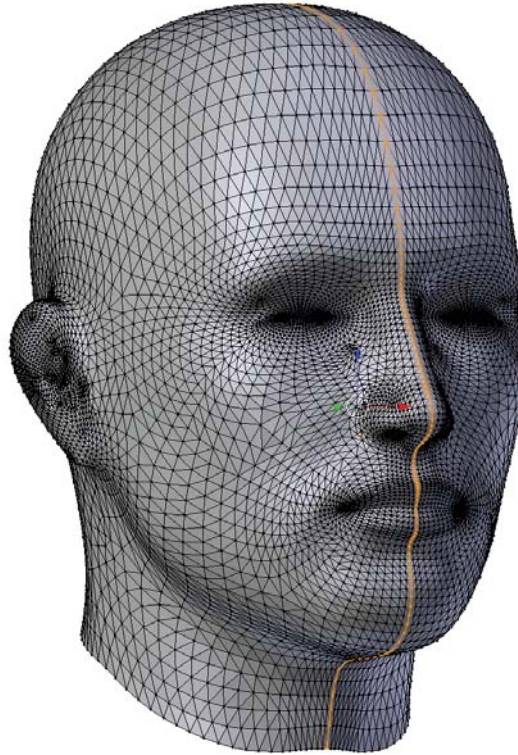


Abbildung 11: Die Vertices der Profillinie wurden im Dreiecksnetz markiert (orange).

Im ersten Schritt der Registrierung werden nur Translation, Skalierung und Rotation, nicht aber die Form des Modells angepasst. Würde die Transformation gleichzeitig mit der Formveränderung einsetzen, würden sich bei schlecht überlappenden Korrespondenzen die Regionen sehr stark verformen, um sich der Vorgabe anzupassen, was zu einem schlechten Ergebnis führt. Daher wird erst eine optimale Ausrichtung des Modells gesucht, ausgehend von derer im zweiten Schritt auch die Form verändert wird.

Mit den neuen Werten werden die Koordinatenpositionen des Modells verändert und anhand dieses neuen Modells wieder die Korrespondenzen zu den Zielwerten bestimmt. Daraufhin werden wieder die Parameter aktualisiert. Dies geschieht solange, bis der Fehler $e(s, \theta, \mathbf{t})$ minimal ist oder unterhalb einer Schwelle sinkt. Dieser Ablauf ist in Abbildung 12 zu sehen.

Ist dies der Fall, wird die Form ebenfalls mit angepasst. Dies geschieht mit der Gleichung 43, bis auch der dadurch entstehende Fehler minimiert wurde. Es gibt bei der Profillinie jedoch deutlich weniger Punkte und damit Gleichungen in der Matrix als bei der Registrierung des gesamten Netzes. Dadurch ist das Gleichungssystem für die Gesamtmatrix der Hauptkomponentenvektoren unterbestimmt. Daher müssen die Komponenten reduziert werden, sodass die Matrix wieder bestimmt ist.

Die Regioneneinteilung des Modells führt des Weiteren dazu, dass die Form und Lage einiger Regionen nicht angepasst werden kann. Dazu gehören die Ohren, Augen

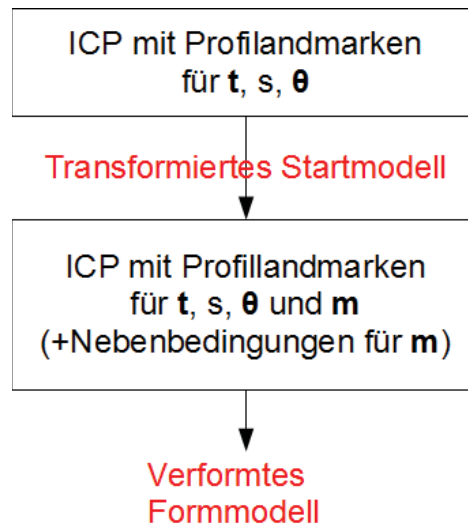


Abbildung 12: Ablauf der Profillinienrekonstruktion

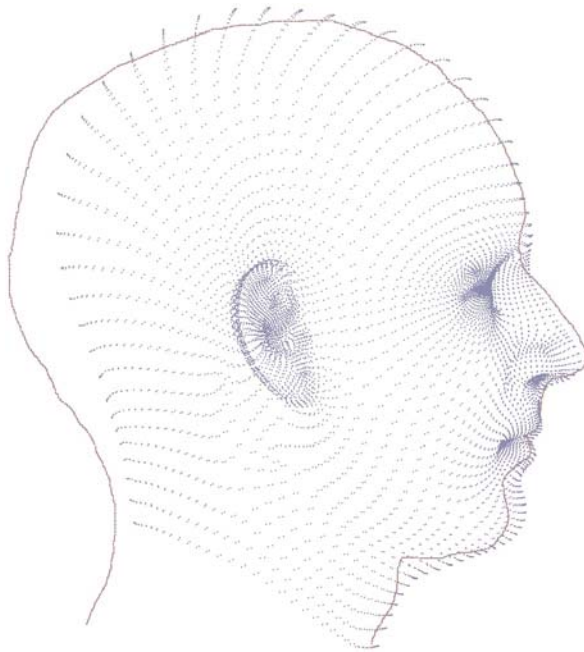
und die Wangenregion, da es keine Zielpunkte in diesen Regionen gibt. Daher ist es wichtig, dass mindestens die erste Nebenbedingung aus Abschnitt 4.3.2 besetzt ist. Dadurch werden die Vertices der Regionengrenzen zusammengezogen. Wodurch auch die Regionen, die keine Korrespondenzpunkte haben, durch einige Gleichungen der überlappenden Vertices beschrieben werden und das Gleichungssystem 43 gelöst werden kann.

Das so registrierte Dreiecksnetz stimmt nach diesem Algorithmus in Form und Lage mit dem Kopf auf dem Profilbild überein. Beispielhafte Ergebnisse für die Anpassung der Lage und der Form des Modellkopfes an die Profillinie sind in Abbildung 13 zu sehen. Die Frontalansicht kann jedoch stark von der eigentlichen Person abweichen. Dadurch werden in den Abschnitten 4.4.2 und 4.4.3 zwei Möglichkeiten zur Registrierung des Modells mit dem Frontalbild beschrieben.

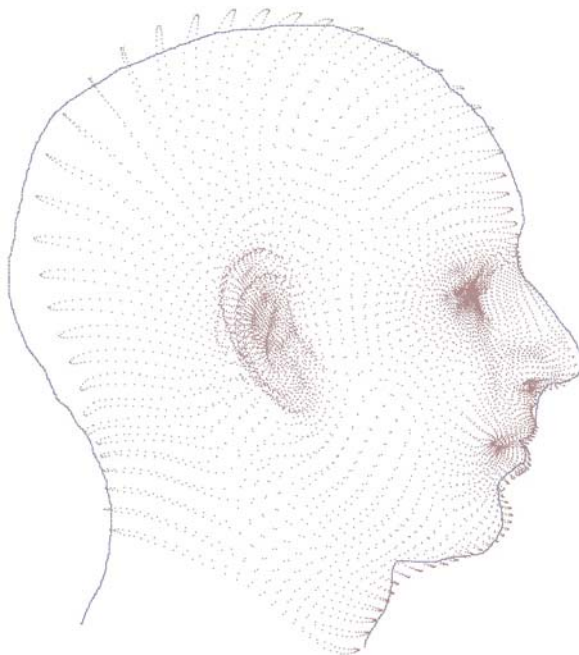
4.4.2 Ansatz 1: ICP der Landmarken ohne frontale Tiefeninformation

Das Fitting durch die Profillinie passt die Seitenansicht des Kopfes an die gewollte Form an. Wird das Modell nun jedoch um 90° nach vorne gedreht, gibt es wenig Ähnlichkeit zu dem Frontalbild des Kopfes. Die Kinnpartie wird kaum stimmen, die Lage und Form der Ohren und Augen ebenfalls nicht. Auch die Nasenflügel können noch nicht passen. Dies sind jedoch alles markante Gesichtsmerkmale, die für die Übereinstimmung des Modells mit den Fotos möglichst an den gleichen Stellen ansetzen und die ähnliche Form aufweisen sollten.

All diese Gesichtspartien sind im Frontalbild zu erkennen. Ihnen fehlt jedoch eine entscheidende Information, nämlich die Entfernung von der Kamera - die Tiefeninformation.



(a) Anpassung der Transformation an die Profilinie



(b) Anpassung von Form und Transformation an die Profilinie

Abbildung 13: Die Abbildung zeigt das Ergebnis der einzelnen Schritte der Profilinienanpassung. Im ersten Bild wurde nur die Transformation berechnet. Das zweite Bild zeigt das Formmodell mit bestmöglicher Anpassung der Form an die Profilinie.

Die zwei aus dem Bild ausgelesenen Koordinaten, die X- und die Y- Koordinate, können aber als fester Zielwert angenommen werden. In diesem Ansatz wird versucht, die fehlende Tiefeninformation - die Z-Koordinate - über andere Landmarken, in denen alle drei Koordinaten vorhanden sind, zu berechnen. Diese Landmarken sind die Profillandmarken. Wird die ICP mit den Formparametern, wie sie auch in der Profillinienrekonstruktion verwendet wird, zusätzlich um die Gleichungen der Frontallandmarken erweitert, sollte das Modell auch an den frontalen Gesichtsmarkern verformt werden.

Die Tiefeninformation - also die dritte Gleichung - der Frontallandmarken wird dabei nicht verwendet. Dies stört jedoch nicht, da durch die Profillandmarken bereits genügend Information über die Z-Koordinaten enthalten ist.

Für die Frontallandmarken müssen auch in dem Formmodell wieder die korrespondierenden Vertices markiert werden, die zu den Landmarken im Bild passen. Diese Korrespondenzen sind bereits eindeutig und müssen nicht wie bei der Profillinie oder der 3D-3D Registrierung durch K-d-Bäume oder ähnliches gefunden werden.

Die Gleichung der ICP ist wie bereits erwähnt mit der der Profillinienrekonstruktion identisch. Unterschiede gibt es nur in der Dimension von $\mathbf{p}'_i = (x'_i, y'_i)$ und damit auch $\mathbf{q}'_i = (x''_i, y''_i)$, der Frontallandmarken \mathbf{p}'_i und der korrespondierenden Vertices \mathbf{q}'_i im Dreiecksnetz. Die Matrix der Kostenfunktion mit den I Profillandmarken wird um die J Punkte der Frontallandmarken ergänzt und sieht wie folgt aus:

$$\begin{bmatrix} \tilde{\mathbf{p}}_1 & [\tilde{\mathbf{p}}_1]_{\times} & s \operatorname{rot}(\boldsymbol{\theta})\mathbf{M}_1 & \mathbf{I}_3 \\ \vdots & \vdots & \vdots & \vdots \\ \tilde{\mathbf{p}}_I & [\tilde{\mathbf{p}}_I]_{\times} & s \operatorname{rot}(\boldsymbol{\theta})\mathbf{M}_I & \mathbf{I}_3 \\ \tilde{\mathbf{p}}'_1 & [\tilde{\mathbf{p}}'_1]_{\times} & s \operatorname{rot}(\boldsymbol{\theta})\mathbf{M}_1 & \mathbf{I}_3 \\ \vdots & \vdots & \vdots & \vdots \\ \tilde{\mathbf{p}}'_J & [\tilde{\mathbf{p}}'_J]_{\times} & s \operatorname{rot}(\boldsymbol{\theta})\mathbf{M}_J & \mathbf{I}_3 \end{bmatrix} \begin{bmatrix} \Delta s \\ \Delta \boldsymbol{\theta} \\ \Delta \mathbf{m} \\ \Delta \mathbf{t} \end{bmatrix} = \begin{bmatrix} \mathbf{q}_1 - \tilde{\mathbf{p}}_1 - \mathbf{t} \\ \vdots \\ \mathbf{q}_I - \tilde{\mathbf{p}}_I - \mathbf{t} \\ \mathbf{q}'_1 - \tilde{\mathbf{p}}'_1 - \mathbf{t} \\ \vdots \\ \mathbf{q}'_J - \tilde{\mathbf{p}}'_J - \mathbf{t} \end{bmatrix}. \quad (47)$$

Sie hat jetzt jedoch nicht mehr $(I + J) \cdot 3$ Gleichungen, sondern nur $I \cdot 3 + J \cdot 2$. Da J in etwa dreimal kleiner ist als I , haben die Profillandmarken einen deutlich größeren Einfluss auf die Lösung als die Frontallandmarken. Aus diesem Grund sollten die Frontallandmarken gewichtet werden, sodass dieser Unterschied ausgeglichen wird.

Dieses System braucht weiterhin noch mindestens eine der Nebenbedingungen aus Abschnitt 4.3.2. Denn in den Bildern wird kein Punkt in der Wangenregion gesetzt, wodurch die Matrix für einen kleinen Teil der Parameter unbestimmt ist. Die Nebenbedingungen verhindern dies, da sie jeder Region Beschränkungen auferlegen.

Der gesamte Algorithmus dieses Ansatzes ist wie folgt: zu Beginn erfolgt die Änderung der Transformation mit darauffolgender Formanpassung anhand der Profillandmarken wie in Abschnitt 4.4.1 beschrieben um eine grobe Anpassung des Kopfes zu

erhalten. Daraufhin wird anhand der Gleichung oben eine ICP mit allen vorhandenen Landmarken durchgeführt.

4.4.3 Ansatz 2: Ganzheitlich dreidimensionale Registrierung

Im ersten Ansatz bilden einzig die Landmarken und deren Positionen in Frontal- und Profilaufnahme die Grundlage für die Registrierung eines gesamten dreidimensionalen Modells. Der zweite Ansatz versucht, ein dreidimensionales Zielobjekt mit gleicher Topologie wie das Referenzmodell zu erstellen. Dieses Zielobjekt hat die Eigenschaft, ebenfalls an den Landmarkvertices die Positionen der markierten Landmarken des Bildes einzunehmen. Jedoch verfügt es, da es sich um ein komplettes Modell handelt, über deutlich mehr Informationen über die restlichen Vertices als der erste Ansatz, da sie alle in die Berechnungen mit einfließen.

Die Grundlage für diesen Ansatz ist ebenfalls wieder der bereits verwendete ICP-Algorithmus. Im ersten Schritt wird das Formmodell an die Profillinie ausgerichtet, um eine Übereinstimmung in der XZ-Ebene (siehe Abbildung 9, Seite 27) zu erreichen. Dies geschieht mit dem in Abschnitt 4.4.1 beschriebenen Algorithmus.

Wenn das Formmodell an das Profil ausgerichtet ist, wird es an die Frontallandmarken angepasst. Anstatt jedoch gleich mit der ICP für die Landmarken zu beginnen, wird erst ein passendes 3D-Modell erzeugt. Dafür wird das Formmodell nach der Anpassung der Profillinie als Grundlage genommen.

Von den Frontallandmarken sind die X- und die Y-Koordinate bekannt, die Tiefeninformation, die Z-Koordinate, fehlt. Die Frontalregistrierung in diesem Ansatz beginnt mit einer Verzerrung des Modells durch Thin Plate Splines an die in den Fotos markierten Stellen hinsichtlich der X- und Y-Koordinaten. Wie in Abschnitt 2.2 beschrieben, wird ein dreidimensionales Dreiecksnetz an vorgegebenen Vertices auf bestimmte Positionen gezogen und die nicht vorgegebenen Vertices werden glatt interpoliert. Dafür werden aber zum einen Zielpositionen in der gleichen Dimension benötigt, in der das Netz vorliegt und zum anderen müssen bestimmte Vertices festgelegt sein, die mit den neuen Positionen korrespondieren. Dafür werden die Vertices aus dem vorläufig an die Profillinie angepassten Modell ausgelesen und die fehlende Z-Position der Landmarken aus dem korrespondierenden Vertex extrahiert. Damit wird nun eine Verformung durch TPS durchgeführt. Das Formmodell wird an den Vertices der Landmarken an die exakte Position gezogen. Trotz einigermaßen glatten Ergebnisses (siehe Abbildung 14) ist diese Verformung zum Teil noch unrealistisch (rote Kreise).

Aus diesem Grund wird ein weiterer Schritt durchgeführt, nämlich eine ICP des Modells vor der TPS-Verformung auf das Zielmodell nach der TPS-Verformung. Dadurch wird erreicht, dass die zum Teil kantigen Ergebnisse wie in Abbildung 14 natürlicher werden. Denn die ICP hat als Grundlage die vorhandene Datenbank und versucht anhand der Eigenvektoren dieser, das Modell an die Vorgabe anzugleichen.

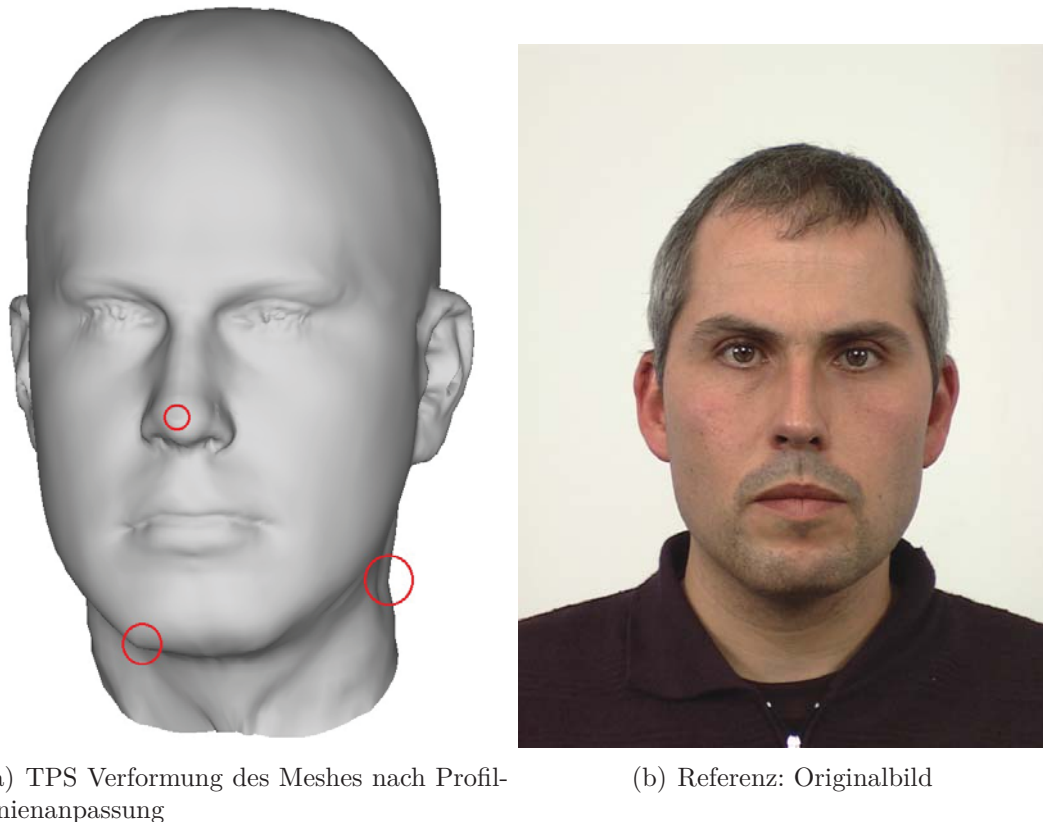


Abbildung 14: Das Formmodell nach der Anpassung an die Profillinie wurde mit Hilfe von Thin Plate Splines an den Stellen der frontalen Landmarken verformt. Die anderen Vertices wurden entsprechend der Glattheit der Splines angepasst. Die roten Kreise zeigen Vertices, deren Glattheit noch verbessert werden muss.

Wird der TPS-ICP Algorithmus das erste Mal ausgeführt, liegen Ist- und Soll-Wert noch weit voneinander entfernt. Lässt man die beiden Berechnungen iterativ hintereinander ablaufen, nähern sich die beiden Modelle immer weiter einander an. Denn die Zielwerte der Z-Koordinate der Vertices der Front-Landmarken bei der TPS-Verformung werden nach jeder ICP korrigiert und auf die berechneten Koordinaten gesetzt. Wenn sich der Fehler, der durch die ICP minimiert wird, nicht weiter verringert, bricht der Algorithmus ab. Der Ablauf ist in Abbildung 15.

Der Vorteil dieses Ansatzes ist, dass neben den markierten Vertices aus den beiden Bildern auch alle anderen Vertices als Zielwerte dienen. Dies liegt an den Thin Plate Splines, die das gesamte Formmodell verändern. Dadurch wird die Matrix der darauffolgenden ICP deutlich größer und kann aufgrund der Überbestimmtheit genauer gelöst werden als eine kleinere Matrix.

Ein Nachteil liegt jedoch auch darin, dass die wichtigen markierten Vertices das gleiche Gewicht haben, wie alle anderen. Da es wichtiger ist, dass diese Vertices besonders berücksichtigt werden, wenn die Formparameter berechnet werden, können den Gleichungen in der ICP-Matrix höhere Gewichte zugeordnet werden. Dadurch wird die ICP in Richtung der Landmarken verändert.

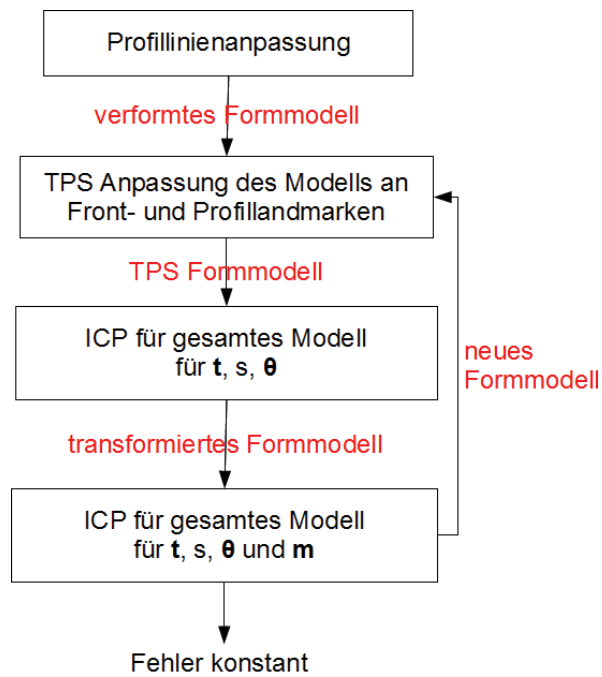


Abbildung 15: Ablauf des TPS-ICP Algorithmus

Weiterhin wird in diesem Ansatz die dritte Koordinate der Zielvertices nur geschätzt, da die exakte Position unbekannt ist. Dies führt zwangsläufig zu einem Fehler in der Rekonstruktion. Die Auswertung dieses Fehlers ist im Diagramm 36 in Abschnitt 7.5 beschrieben

5 Texturierung

Eine Textur ist ein zweidimensionales Bild, das auf ein virtuelles dreidimensionales Objekt abgebildet wird. Ein Vorteil dieser Darstellung ist die Erhöhung des Detailgrades des Objektes, ohne das Polygonnetz zu verändern oder zu verfeinern. Diese einfache Art der Texturierung nennt man “Texture Mapping”. Damit werden die zugrundeliegenden Ausgangsbilder der Köpfe auf den modellierten Kopf projiziert. Das Ergebnis erlaubt eine optische Einschätzung der Übereinstimmung und Genauigkeit des Modells mit dem Foto.

5.1 Berechnung der Texturkoordinaten

Damit die Textur auf das dreidimensionale Objekt abgebildet werden kann, muss jedem Vertex ein Punkt in der Textur zugewiesen werden. Da in dieser Arbeit besonderen Wert auf die Geometrieänderung gelegt wird, ist bei der Texturierung nur ein sehr einfacher Algorithmus verwendet worden. Es handelt sich um eine orthografische Abbildung des Fotos auf das Gesicht des Modells. Diese Annahme ermöglicht die Projektion bei unbekannter Kamerakalibrierung, ist dadurch aber nicht sehr genau, sodass die Textur an den Rändern (Ohr, Kopf, Hals) zu klein erscheint.

Bei dieser orthografischen Abbildung werden ausgehend von einem Punkt, der sowohl im Bild, als auch im Modellkopf bekannt ist, alle weiteren Texturkoordinaten anhand der X- und Y-Koordinaten im 3D Objekt übernommen.

Dafür werden die Pixelpositionen (U, V) im Bild in Texturkoordinaten (u, v) umgerechnet. Im Bild ist der Koordinatenursprung in der linken oberen Ecke mit den Ausdehnungen der Bildbreite und Bildhöhe in der rechten unteren Ecke, hier 768×1024 Pixel. Die Texturkoordinaten haben ihren Ursprung in der linken unteren Ecke und reichen bis $[1, 1]$ in der rechten oberen Ecke. Weiterhin sind die Koordinaten im Modell zu berücksichtigen, die die Dimensionen der annotierten Landmarken (siehe Abschnitt 4.2) haben, wo die Pixelposition durch die Bildhöhe geteilt wird, um eine Skalierung zwischen 0 und 1 zu erreichen und der Ursprung auf die Nasenspitze verschoben ist. Weiterhin ist das Koordinatensystem so gedreht, dass die Transversalrichtung des Kopfes durch die Y-Koordinate beschrieben wird, die Longitudinalrichtung durch die negative X-Koordinate und die Sagittalrichtung durch die negative Z-Koordinate. Das entspricht der Richtung des durch Blender erstellten Referenzkopfes (Abschnitt 3.1). Die drei Koordinatensysteme sind in Abbildung 16 zu sehen.

Anhand dieser Konstellation der verschiedenen Koordinatensysteme werden die Texturkoordinaten u und v aus den 3D Koordinaten X, Y und Z , den Bildpunkten U und V , der Bildposition der Nasenspitze in Pixeln (U_n, V_n) und der Bildgröße (h, b) wie folgt berechnet:

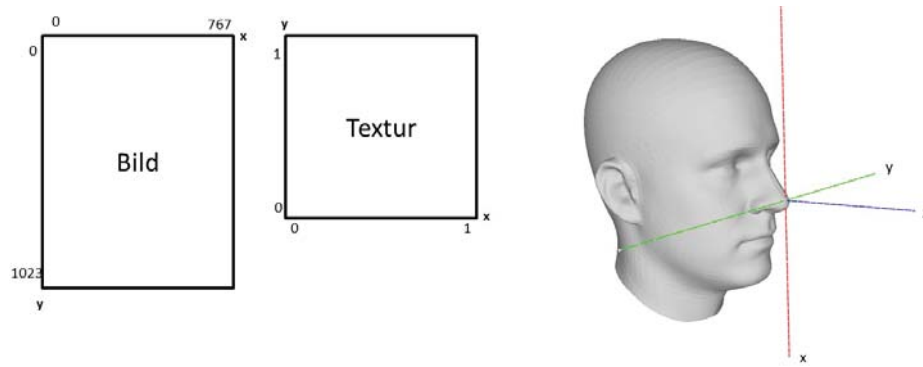


Abbildung 16: Koordinatensysteme von Bild, Textur und Modell

$$u = \frac{U_n + Y \cdot h}{b} \quad (48)$$

$$v = 1 - \frac{V_n}{h} + X \quad (49)$$

Texturierte Ergebnisse sind im Anhang auf Seite 85 zu sehen.

6 Implementierung

Für diese Arbeit bietet es sich an, Benutzeroberflächen zu implementieren, da Landmarken in 3D Dreiecksnetzen und auf Fotos markiert werden und verschiedene Parameterkombinationen ausprobiert werden müssen. Weiterhin bieten diese Oberflächen eine einfache Betrachtungsmöglichkeit der Ergebnisse.

Aus diesem Grund wurde bei der Implementierung C++ verwendet und Qt für die Benutzeroberflächen benutzt.

Im Fraunhofer HHI wurde bereits eine Bibliothek für das Laden und Verwenden von 3D Objekten entwickelt, die in dieser Arbeit benutzt wurde. Weiterhin wurden auch Funktionen zum Lösen von Gleichungssystemen und zur Berechnung von Thin Plate Splines bereits im Fraunhofer HHI implementiert und konnten daher so verwendet werden.

6.1 Benutzeroberflächen

Die verschiedenen Abschnitte, zu denen grafische Interaktion durch Benutzeroberflächen benötigt wurde, sind:

- Das Setzen der Landmarken für die topologische Anpassung der Originaldaten an den Referenzkopf.
- Das Berechnen des statistischen Formmodells und Testen der Gewichte der Eigenvektoren der Regionen.
- Das Testen der 3D Registrierung an Originaldatensätzen.
- Das Setzen der Landmarken in den Frontal- und Profilaufnahmen.
- Das Starten der Berechnung für die 2D-3D Registrierung mit Parameteränderungen.
- Das Anzeigen und Speichern der jeweiligen Ergebnisse.

Diese Anwendungen wurden auf zwei Oberflächen verteilt. In der ersten finden alle Schritte statt, die zur Vorbereitung für die Registrierung dienen. Dazu gehören das Setzen der Landmarken und die Anpassung der Originaldaten an die Referenztopologie und die Berechnung des Formmodells inklusive der Speicherung der Datenmatrizen der Eigenvektoren und Mittelwerte, die später in der Registrierung benötigt werden. Die erste Oberfläche ist in Abbildung 17 zu sehen.

Auf der linken Seite findet die Interpolation mit dem Referenzkopf statt. Dafür kann ein Originaldatum geladen werden. Dieses wird im mittleren Fenster mit Hilfe von OpenGL angezeigt. Dort kann es skaliert und gedreht werden. In diesem Fenster können mit Rechtsklick Landmarken gesetzt oder verschoben werden; und zwar immer

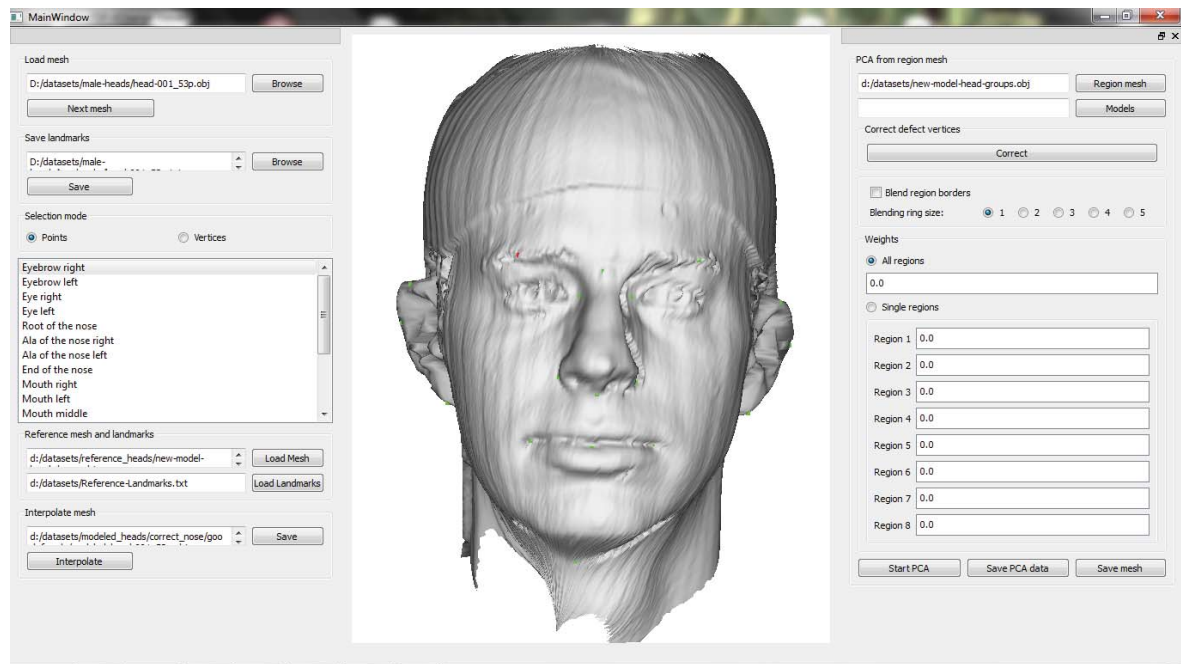


Abbildung 17: Benutzeroberfläche 1

passend zu dem markierten Namen im Fenster auf der linken Seite. Die aktuell markierte Landmarke wird rot angezeigt, alle anderen grün. Diese Landmarken sind immer an der aktuellen Mausposition des Klickes positioniert. Sind alle Landmarken gesetzt, können sie gespeichert werden, damit sie für weitere Berechnungen zur Verfügung stehen.

Im gleichen Fenster können auch die Referenzlandmarken positioniert werden. Dafür besteht die Möglichkeit, das Setzen der Landmarken von der aktuellen Mausposition auf "Vertices" umzustellen. Dann wird die Markierung auf die nächstgelegene Vertex-Position festgesetzt und der Index des Vertices gespeichert. Diese Indices werden für die Interpolation der Originalköpfe an den Referenzkopf benötigt.

In der gleichen Oberfläche findet auch die Berechnung des Formmodells statt. Dafür kann ein Ordner mit Daten und ein Referenzmodell angegeben werden, woraufhin die Datensätze gelesen und das Formmodell berechnet wird.

Das Modell wird ebenfalls in dem mittleren Fenster angezeigt. Es können auch die verschiedenen Gewichtungen der Eigenvektoren für die Regionen verändert und angezeigt werden. Dafür gibt es die Möglichkeit, alle Regionen mit den gleichen Gewichten zu berechnen oder alle Regionen einzeln zu gewichten. Auch kann zwischen verschiedenen Überblendungsgrößen der Regionen gewählt werden.

Ebenfalls in dieser GUI können die PCA-Matrix und damit die Eigenvektoren, sowie der Mittelwertvektor des Formmodells gespeichert werden. Diese werden als Textfile formatiert und werden für die Registrierung von 2D oder 3D Daten in der zweiten Benutzeroberfläche benötigt.

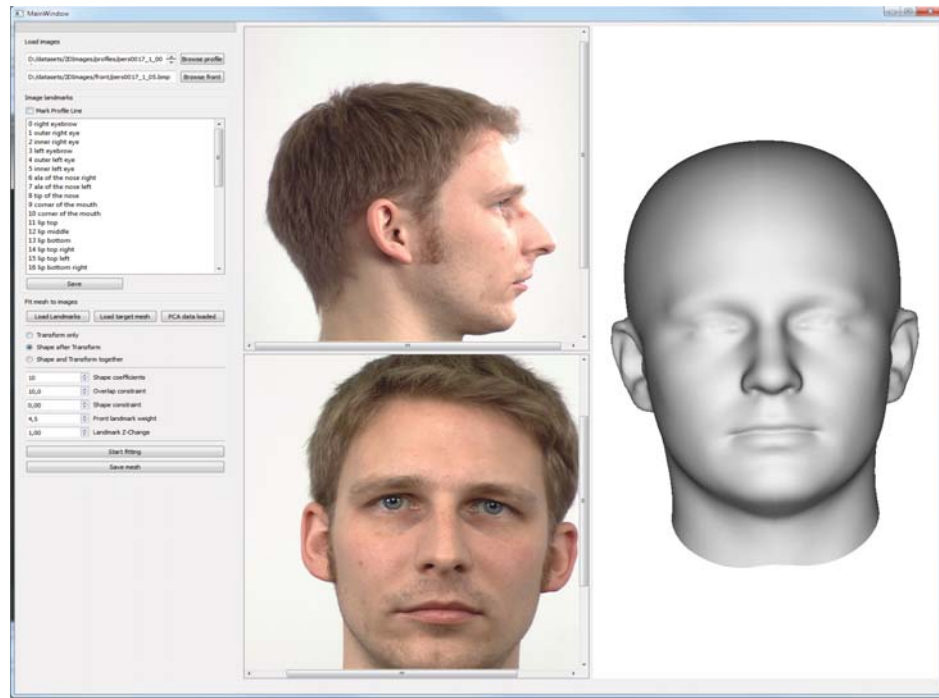


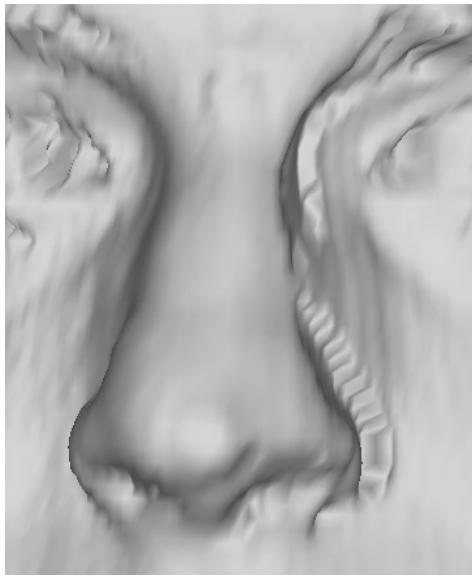
Abbildung 18: Benutzeroberfläche 2

Die zweite Oberfläche ist in Abbildung 18 zu sehen. In den beiden mittleren Fenstern können bei der 2D Registrierung das Profilbild und die Frontalaufnahme geladen werden. Hier können auch die Landmarken gesetzt werden, indem eine Taste gedrückt und dabei die linke Maustaste geschoben wird. Dadurch kann eine zusammenhängende Linie, wie in der Profillinie benötigt, erzeugt werden. In dem kleinen Fenster auf der linken Seite werden die Landmarkennamen für die Frontalaufnahme dargestellt. Klickt man mit rechts auf eine Landmarke, öffnet sich ein kleines Infofenster und die ID der Markierung wird angezeigt. So kann man kontrollieren, dass alle Landmarken die richtige ID haben. Ebenfalls gibt es in diesem Infofenster die Option, eine Landmarke als Nullpunkt festzulegen. In diesem Fall ist das die Nasenspitze. Auch kann so eine Landmarke entfernt werden.

Das rechte Fenster ist wieder eine OpenGL-Anzeige zur 3D Darstellung. Dort wird bei der 3D Registrierung das Originaldatum und nach fertiger Registrierung das Ergebnis angezeigt.

Je nachdem, ob für die Registrierung ein “Target-Mesh” oder die 2D-Bilder geladen werden, setzt die jeweilige Registrierung automatisch ein. Für beide muss jedoch vorher eine PCA-Matrix und ein Mittelwertsvektor geladen werden.

Auch hierfür können wieder die Parameter für die oben beschriebene Kostenfunktion geändert werden. Dazu gehören sowohl diejenigen der 3D- Registrierung als auch die, der beiden 2D Ansätze für die Fotos.



(a) Scanfehler der Nase



(b) Scanfehler der Haare bei Frauen

Abbildung 19: Scanfehler der Nase und der Haarpartie

7 Ergebnisse

7.1 Beurteilung des Formmodells

Das Formmodell kann aus verschiedenen Originaldaten erstellt werden. In dieser Arbeit wird versucht, sowohl Männer als auch Frauenfotos in ein dreidimensionales Modell zu verwandeln. Es gibt die Möglichkeit, ein Modell für Männer und ein Modell für die Frauen zu erstellen oder ein Gesamtmodell für beide Gruppen. In dieser Arbeit stehen 187 Datensätze männlicher Probanden und 182 Datensätze weiblicher Probanden zur Verfügung. Diese Daten sind jedoch von unterschiedlicher Qualität. Einige Personen wurden vom Scanner unvollständig erfasst, sodass einige Regionen fehlen, andere weisen Scanfehler auf. Bei den Frauendaten ist auffällig, dass viele Daten am Haaransatz hinter den Ohren und am Hals ein Nackenschild ähnlich wie bei Kobras aufweisen. Dieses ist auf das Tragen einer Kappe über den Haaren zurückzuführen. Bei den Männerdaten gibt es 46 Datensätze, die einen ähnlichen Scanfehler an der linken Nasenseite aufweisen. Für die Erstellung des Formmodells werden die Daten vorverarbeitet, wie in Abschnitt 3.3 beschrieben. Für diese Vorverarbeitung können nicht alle Daten verwendet werden. Diejenigen Daten, bei denen aufgrund fehlender Scanbereiche nicht alle Landmarken gesetzt werden können, werden aussortiert. Nach der Vorverarbeitung bleiben noch 169 Männerdatensätze und 176 Frauendatensätze übrig.

Diese Daten weisen jedoch noch wiederholt gleiche Fehler auf. Bei den Männerdaten weisen 46 Datensätze den Scanfehler der Nase auf. (Zu sehen in Abbildung 19(a)) Die Frauendaten sind von wesentlich schlechterer Qualität. Nur 21 Datensätze sind

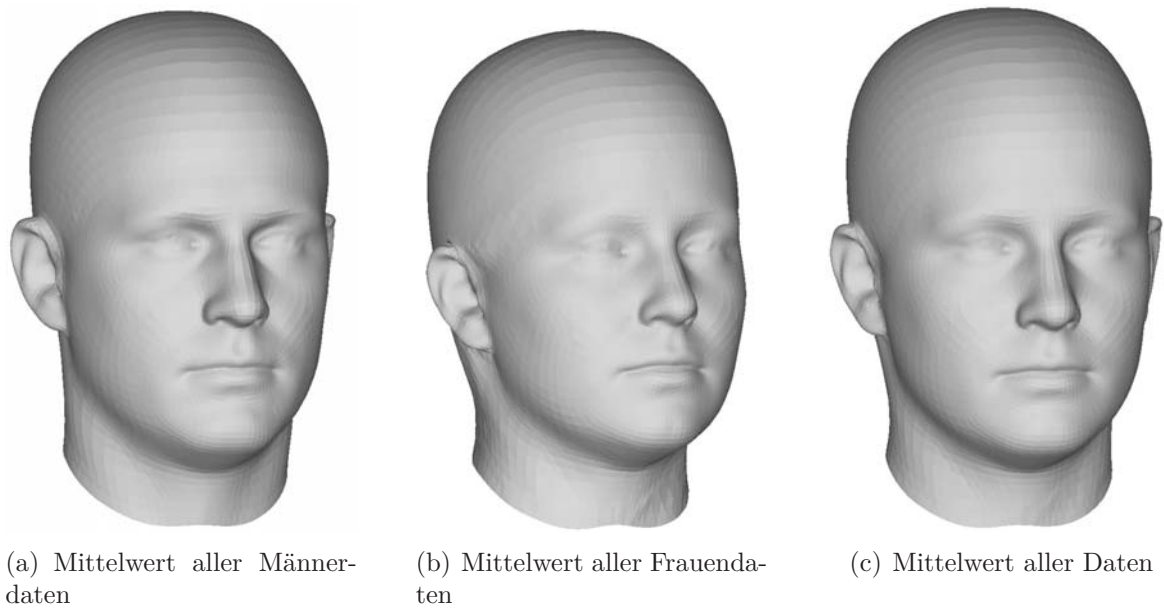


Abbildung 20: Varianten des Formmodells

frei von Badekappenartefakten (siehe Abbildung 19(b)), vollständig, unverrauscht und scanfehlerfrei.

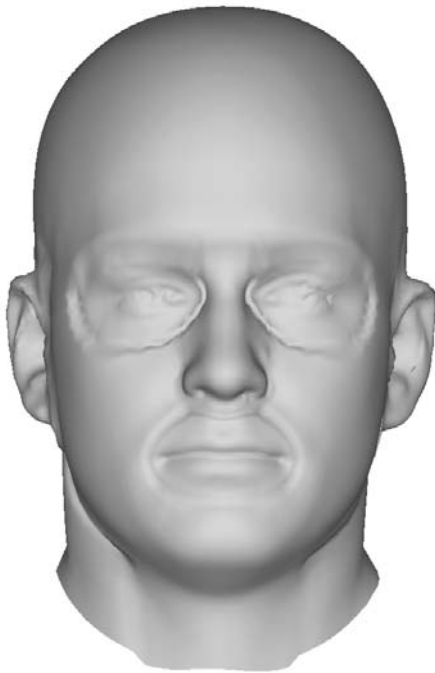
Die Mittelwerte μ der Datensätze aller Männer und Frauen inklusive fehlerbehafteter Daten sind in Abbildung 20 dargestellt.

Es ist zu erkennen, dass das Modell der Frauenköpfe kleiner ist und weichere Gesichtszüge aufweist, als das Modell der Männerdaten. Auch erkennt man bereits in dem gemittelten Datensatz den Scanfehler am Haaransatz. Als Beispiel für eine Formänderung ist in Abbildung 21(a) das Modell der Männer und in Abbildung 21(b) der Frauen mit gewichteten Eigenvektoren zu sehen. Der jeweils erste Eigenvektor jeder Region wurde mit dem Faktor 1 gewichtet. Es sind deutlich die Regionengrenzen zu erkennen. Um dies zu verhindern, wurde eine Überblendung der Grenzen implementiert (siehe Abschnitt 3). Das Ergebnis ist in Abbildung 21(c) und 21(d) zu sehen. In beiden Varianten ist zu erkennen, dass bereits der erste Eigenvektor die spezifischen Fehler gelernt hat. Bei dem Männerkopf ist die gezackte linke Nasenseite zu erkennen und bei dem Frauenkopf der Fehler durch die Badekappe.

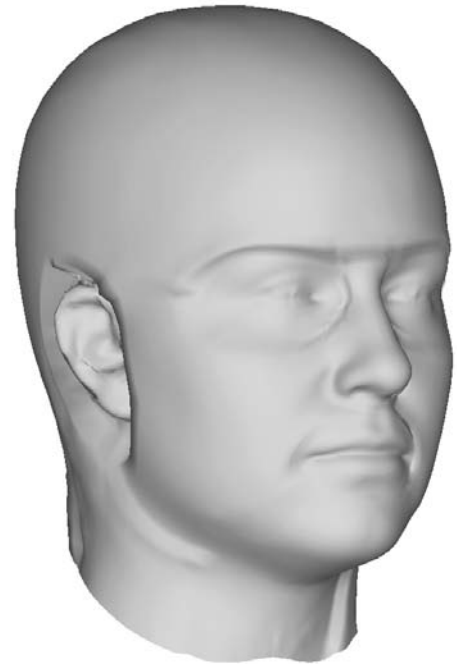
Aus diesem Grund werden für die Registrierung sowohl der 3D-Originaldaten als auch der 2D Registrierung der Fotos die fehlerhaften Daten aussortiert. Übrig bleiben 123 männliche Datensätze und nur 21 weibliche.

7.2 Ergebnisse der Kostenfunktion

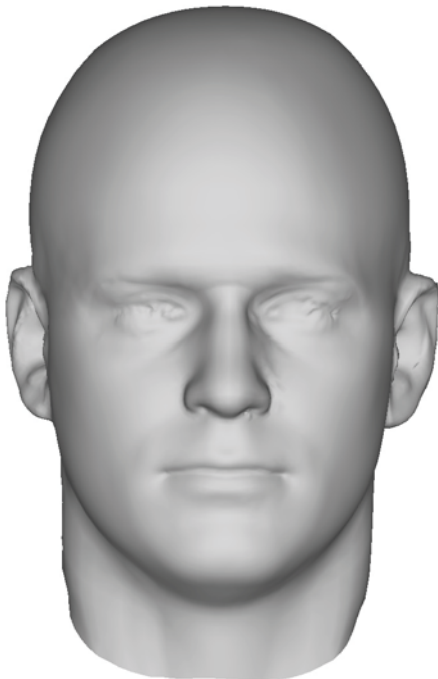
In Abschnitt 4.3.1 wird die Kostenfunktion für die erweiterte ICP mit Formparametern vorgestellt. Diese Funktion wird in den Ansätzen zur 2D-3D Registrierung verwendet. Die Güte der Minimierung kann jedoch schlecht anhand der zweidimensionalen Bilder als Zielwerte gemessen werden. Denn diese Zielwerte können aufgrund der unbekannten



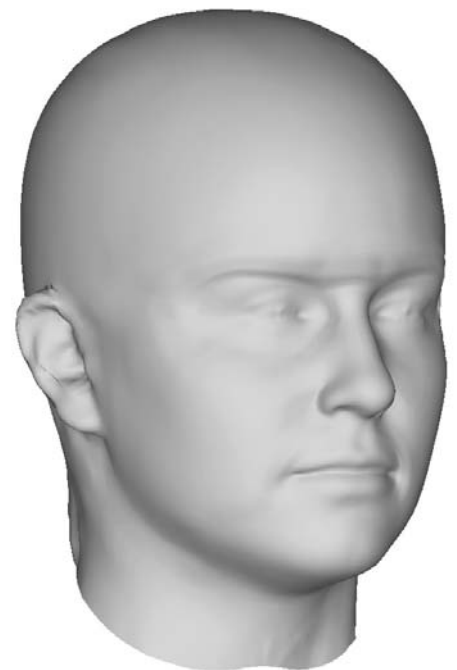
(a) Formmodell Mann mit Gewicht 1



(b) Formmodell Frau mit Gewicht 1



(c) Formmodell Mann mit Überblendung



(d) Formmodell Frau mit Überblendung

Abbildung 21: Die Abbildungen zeigen das Formmodell aller Männer- bzw. Frauendaten mit gewichteten Eigenvektoren. Der jeweils erste Eigenvektor einer Region wurde mit 1 gewichtet. In den überblendeten Modellen sind die Regionengrenzen deutlich weniger bis gar nicht zu erkennen.

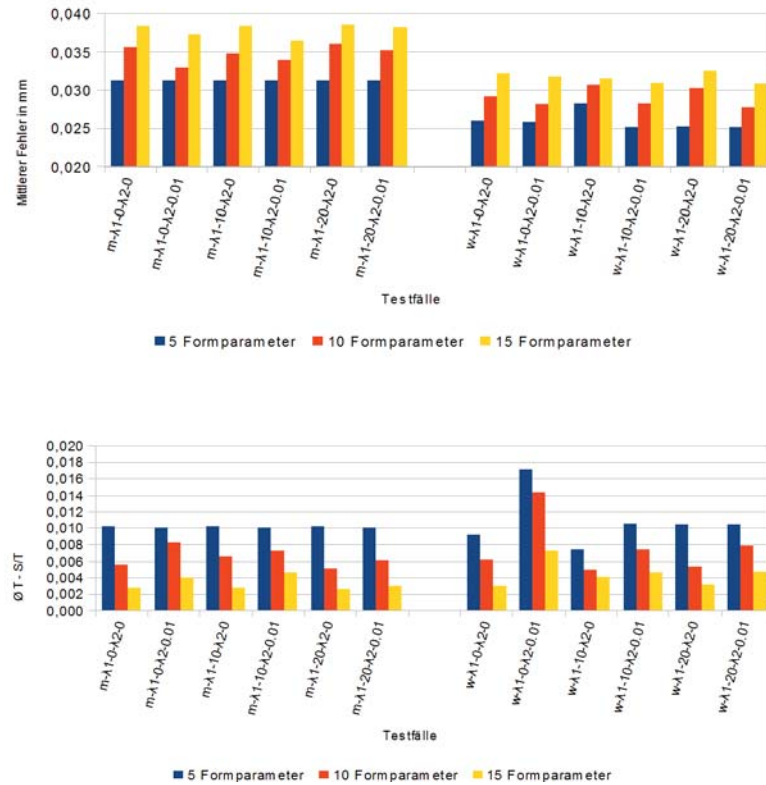


Abbildung 22: Der Einfluss der Anzahl der Formparameter auf den mittleren Fehler 22(a) und die Differenz der ICP mit Formparametern und ohne 22(b).

Kamerakalibrierung und der fehlenden dritten Koordinate nur geschätzt werden. Für eine exakte Auswertung des Fehlers durch den ICP-Algorithmus werden daher die vorhandenen dreidimensionalen Datensätze verwendet.

Alle vorhandenen Daten, auch diejenigen mit Scanfehlern, werden mit dem Formmodell rekonstruiert. Dabei gibt es verschiedene Parameter, die Einfluss auf die Kostenfunktion und damit die Registrierung nehmen. Als erstes spielt die Anzahl der verwendeten Eigenvektoren pro Region eine Rolle. Mehr Eigenvektoren machen die ICP-Matrix komplexer und schwerer lösbar, bringen aber auch mehr Formvariabilität. Der zweite Parameter (λ_1) bezieht sich auf die erste Nebenbedingung und verhindert Diskontinuitäten an den Regionengrenzen. Als nächstes kann die Stärke der Verformung durch den dritten Parameter (λ_2) reguliert werden. Auch Einfluss auf die Rekonstruktion hat die Datenbank, die für die Erstellung des Formmodells verwendet wird, da sie die Variabilität in den Formvektoren bestimmt. Zuletzt verändert auch die Überblendungsgröße der Regionen in der PCA-Matrix die Registrierung.

Zum Testen wurden alle 181 Männerdaten und 181 Frauendaten, inklusive der Datensätze mit Scanfehlern, verwendet. Alle Datensätze werden mit der neuen ICP registriert. Dabei wird der Algorithmus verwendet, der zuerst die einfache ICP mit der Berechnung der besten Transformation gelöst und daraufhin die komplexe mit Trans-

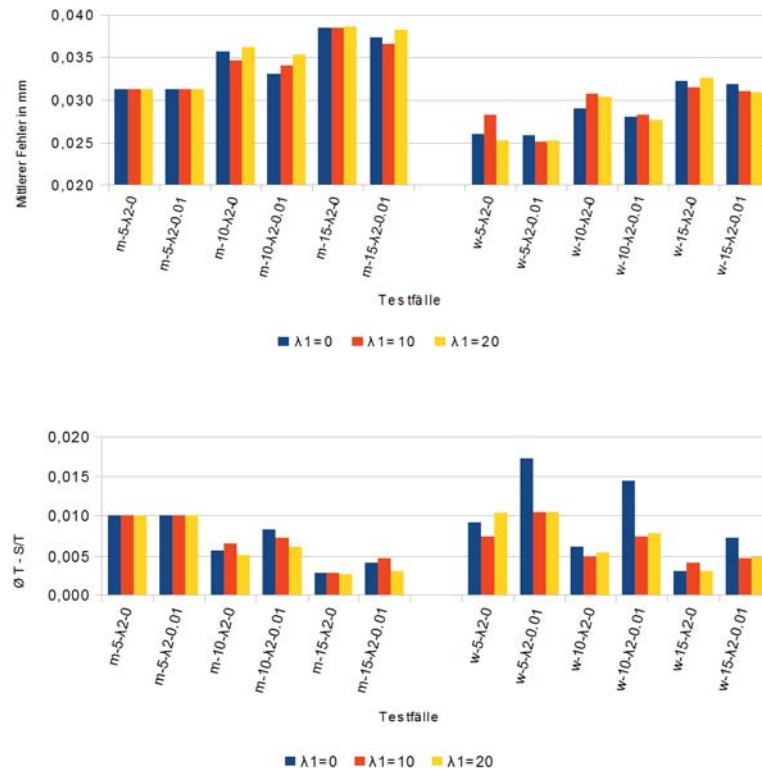


Abbildung 23: Der Einfluss des Glattheitsfaktors λ_1 auf den mittleren Fehler 23(a) und die Differenz der ICP mit Formparametern und ohne 23(b).

formation und Formveränderung. Diese beiden Schritte berechnen jeweils einen Fehler, der ausgewertet werden kann. Die einfache ICP wird immer ein Ergebnis liefern, solange Start- und Zielpunktwolken nicht zu weit voneinander entfernt liegen. Dieses Ergebnis wird mit dem berechneten Fehler der erweiterten ICP verglichen.

Die verwendete PCA-Matrix der Männerdaten wurde aus den Datensätzen ohne Nasenfehler erstellt. Die Frauen-PCA Matrix aus allen Frauendaten, da es sonst zu wenige fehlerfreie Daten gibt. Für die Matrix aller Daten wurden beide Datenbanken zusammengeführt.

Tabelle 2 zeigt die Ergebnisse der Transformations-ICP verschiedener Datenbanken auf die Frauen- bzw. Männerdaten angewandt.

Für die Ergebnistabellen werden folgende Abkürzungen verwendet:

- E_{min} - der durchschnittliche Fehler pro Vertex des Kopfes mit dem besten Ergebnis der ICP
- E_{max} - der durchschnittliche Fehler pro Vertex des Kopfes mit dem schlechtesten Ergebnis der ICP
- E_{μ} - mittlerer Fehler aller Köpfe pro Vertex
- E_{σ} - Standardabweichung des Fehlers aller Köpfe

Datenbank	Geschlecht	E_{min} in mm	E_{max} in mm	E_{μ} in mm	E_{σ} in mm
alle	männlich	0,034	0,2	0,105	0,036
alle	weiblich	0,023	0,185	0,082	0,044
männlich	männlich	0,023	0,129	0,041	0,01
weiblich	weiblich	0,02	0,159	0,036	0,017

Tabelle 2: Ergebnisse der Testfälle der ICP nur mit Transformationsparametern Skalierung, Translation und Rotation. Die Fehler sind in Millimeter pro Vertex bei einer durchschnittlichen Kopfgröße angegeben.

Die Einheiten der Tabelle wurden umgerechnet in Millimeter bei einer Kopfgröße von 30cm. Das entspricht einer durchschnittlichen menschlichen Kopflänge. Da während der Berechnungen der Fehler quadriert wurde, ist hier die Wurzel der Fehler dargestellt.

Die PCA Matrix und das Formmodell aller Daten führt zu einem deutlich schlechteren Fitting als eine Matrix mit getrennten Geschlechtern. Das ist darauf zurückzuführen, dass das Formmodell aller Daten deutlich kleiner ist als das Männer-Formmodell und größer als das Frauen-Formmodell. Die Registrierung ist daher schwieriger, weil Modell und Ziel zu Beginn schlecht übereinander passen. Weiterhin wird das Formmodell zu Teilen auf einen Datensatz registriert, zu dem auch die Datenbank des Formmodells gehört, wodurch auch bessere Ergebnisse erreicht werden, wenn nur gleichgeschlechtliche Datenbanken verwendet werden.

Die ICP der Transformation ist unabhängig von den Formparametern und deren Nebenbedingungen λ_1 und λ_2 . Dagegen sind die Testfallmöglichkeiten bei der Veränderung der Form komplexer. Die Ergebnisse dieser Konstellationen sind in den folgenden Diagrammen zusammengestellt. Dabei beschreibt die X-Achse die Konstellationen nach folgendem Schema: Datenbank - Anzahl der Formparameter - λ_1 (Regionenübergänge) - λ_2 (Verformungsbegrenzung). “m” steht dabei für männlich und “w” für weiblich. Ein Beispiel für einen Testfall wäre “m-10-0-0”, mit einer Datenbank aus nur männlichen Daten, 10 Formparametern, $\lambda_1 = 0$ und $\lambda_2 = 0$.

Für den Vergleich von Transformations-ICP mit der ICP mit Formparameter werden weitere Abkürzungen verwendet;

- \emptyset T-S/T - durchschnittliche Verbesserung durch die Formparameter (Shape/Transform) nach der Transformations-ICP (T)
- % T=S/T - prozentualer Anteil der Datensätze, bei denen es durch die Formparameter in der ICP keine Verbesserung gegenüber ausschließlich der Transformation gibt

Es werden jeweils 5, 10 und 15 Formparameter mit $\lambda_1 = 0$; 10; 20 und $\lambda_2 = 0$; 0,01 getestet. Weiterhin wurden testweise einmal 20 Formparameter und einmal die gesamte Datenbank auf die männlichen bzw. weiblichen Datensätze getestet. Kein Datum wurde in Lage und Größe vorregistriert. Daher kann es vorkommen, dass der Startwert

Testfall	E_{min}	E_{max}	E_{μ}	E_{σ}	$\emptyset T - S/T$	% T=S/T
m-5-0-0	0,0203	0,0501	0,0312	0,0060	0,0101	0
m-5-0-0.01	0,0203	0,0530	0,0312	0,0061	0,0101	0
m-5-10-0	0,0203	0,0501	0,0312	0,0060	0,0101	0
m-5-10-0.01	0,0203	0,0530	0,0312	0,0061	0,0101	0
m-5-20-0	0,0203	0,0501	0,0312	0,0060	0,0101	0
m-5-20-0.01	0,0203	0,0530	0,0312	0,0061	0,0100	0
m-10-0-0	0,0210	0,0767	0,0356	0,0080	0,0056	0,0063
m-10-0-0.01	0,0204	0,0545	0,0330	0,0068	0,0083	0,0018
m-10-10-0	0,0204	0,0576	0,0347	0,0076	0,0066	0,0041
m-10-10-0.01	0,0205	0,0547	0,0340	0,0066	0,0073	0,0022
m-10-20-0	0,0220	0,0624	0,0361	0,0071	0,0051	0,0035
m-10-20-0.01	0,0214	0,0629	0,0352	0,0067	0,0060	0,0024
m-15-0-0	0,0214	0,0794	0,0385	0,0087	0,0028	0,0156
m-15-0-0.01	0,0224	0,1310	0,0373	0,0104	0,0040	0,0108
m-15-10-0	0,0216	0,0915	0,0385	0,0087	0,0028	0,0138
m-15-10-0.01	0,0210	0,0930	0,0365	0,0087	0,0046	0,0081
m-15-20-0	0,0222	0,0711	0,0386	0,0079	0,0026	0,0140
m-15-20-0.01	0,0226	0,1091	0,0382	0,0090	0,0030	0,0099
m-20-10-0	0,0216	0,0576	0,0391	0,0076	0,0016	0,0183
m-20-10-0.01	0,0232	0,1031	0,0388	0,0096	0,0025	0,0140

Tabelle 3: Ergebnisse der Testfälle der ICP mit Formparametern für männliche Daten.

schlecht über dem Ziel liegt. Die ICP geht aber von einer geringen Start/Ziel Entfernung aus, wodurch die hohen Werte der maximalen Fehler zu erklären sind, der Mittelwert aber trotzdem viel kleiner ist. Die genauen Ergebnisse der getesteten Parameter sind in den Tabellen 3 und 4 zu sehen.

Die Tabellen zeigen die ermittelten Fehler aller rekonstruierten Köpfe in Millimeter mit den angegebenen Parametern wieder skaliert auf eine Kopflänge von 30cm. Aus der Tabelle wird ersichtlich, dass die Standardabweichung mit Anzahl der Formparameter zunimmt. Dieser Zuwachs geht einher mit dem Anstieg des prozentualen Anteils nicht verbesserter Daten bei größerer Parameterzahl und der dadurch größeren Differenz der Daten.

Für eine bessere Übersichtlichkeit wurden der mittlere Fehler und die Verbesserung durch eine ICP mit Formparametern weiterhin in Diagrammen dargestellt. Die Diagramme 22(a) und 22(b) zeigen den Einfluss der Formparameter auf die Ergebnisse. Es ist zu erkennen, dass der mittlere Fehler mit der Anzahl der Formparameter steigt, wobei der Fehler der männlichen Daten generell höher ist als der der weiblichen Daten. Die Begründung dafür ist in der Dimension der Fehlerfunktion zu finden. Bei mehr Formparametern steigt die Dimension und die Funktion wird komplexer. Da der ICP Algorithmus jedoch nur zu einem lokalen Minimum konvergiert, kann der Fehler bei mehr Parametern durchaus höher sein als bei wenigen, da es mehr lokale Minima gibt.

Testfall	E_{min}	E_{max}	E_{μ}	E_{σ}	$\text{ØT} - \text{S/T}$	% T=S/T
w-5-0-0	0,0163	0,0530	0,0261	0,0060	0,0092	0,0006
w-5-0-0.01	0,0155	0,0508	0,0259	0,0063	0,0172	0
w-5-10-0	0,0156	0,0836	0,0282	0,0084	0,0074	0,0012
w-5-10-0.01	0,0155	0,0834	0,0251	0,0072	0,0105	0
w-5-20-0	0,0155	0,0864	0,0253	0,0074	0,0103	0
w-5-20-0.01	0,0155	0,0859	0,0252	0,0073	0,0104	0
w-10-0-0	0,0157	0,0698	0,0291	0,0081	0,0062	0,0083
w-10-0-0.01	0,0157	0,0806	0,0281	0,0086	0,0143	0,0024
w-10-10-0	0,0174	0,1403	0,0307	0,0135	0,0049	0,0067
w-10-10-0.01	0,0160	0,0953	0,0282	0,0091	0,0074	0,0026
w-10-20-0	0,0169	0,1403	0,0303	0,0130	0,0054	0,0047
w-10-20-0.01	0,0156	0,0842	0,0277	0,0080	0,0079	0,0016
w-15-0-0	0,0165	0,1575	0,0322	0,0167	0,0030	0,0166
w-15-0-0.01	0,0166	0,1575	0,0318	0,0176	0,0073	0,0099
w-15-10-0	0,0183	0,1403	0,0315	0,0126	0,0041	0,0132
w-15-10-0.01	0,0178	0,1575	0,0310	0,0143	0,0046	0,0104
w-15-20-0	0,0154	0,1575	0,0326	0,0144	0,0031	0,0128
w-15-20-0.01	0,0172	0,1289	0,0309	0,0134	0,0047	0,0065
w-20-10-0	0,0181	0,1575	0,0345	0,0182	0,0011	0,0223
w-20-10-0.01	0,0185	0,1575	0,0339	0,0181	0,0017	0,0191

Tabelle 4: Ergebnisse der Testfälle der ICP mit Formparametern für weibliche Daten.

Die Verbesserung der ICP mit Formparametern fällt dementsprechend bei mehr Formparametern geringer aus als bei wenigen (siehe Diagramm 22(b)).

In den Diagrammen 23(a) und 23(b) ist der Einfluss des Parameters λ_1 dargestellt. Auf den mittleren Fehler hat dieser Parameter wenig Auswirkungen. Nur bei der ICP mit Formparametern tritt bei den weiblichen Daten eine signifikante Verbesserung bei $\lambda_1 = 0$ ein. Das ist ein zu erwartendes Ergebnis für einen Glattheitsparameter. Denn dieser hat hauptsächlich Auswirkungen auf die Optik und weniger auf den Fehler an korrespondierenden Punktwolken.

Zuletzt ist noch der Faktor λ_2 , also die Formbeschränkung in den Diagrammen 24(a) und 24(b) dargestellt. Zu erwarten ist, dass bei einer Beschränkung der Formparameter etwas schlechtere Ergebnisse resultieren, da die ICP-Gleichung nicht allein nach den Korrespondenzen gelöst wird, sondern auch bei der Größe der Änderungen eingeschränkt wird. Das ist auch in den Diagrammen zu erkennen, wobei der Einfluss auf die weiblichen Datensätze wieder größer ist, als bei den männlichen.

Im Diagramm 25 ist die Fehlerentwicklung der jeweils kleinsten und größten Fehler der Registrierung der männlichen und weiblichen Daten zu sehen. Es wird deutlich, dass der kleinste Fehler bei den männlichen Daten immer größer ist als bei den weiblichen Daten, allerdings ist die Streuung der Werte untereinander gering. Das heißt, dass jeder Testfall unabhängig der verschiedenen Parameter ein gutes Ergebnis erzielen

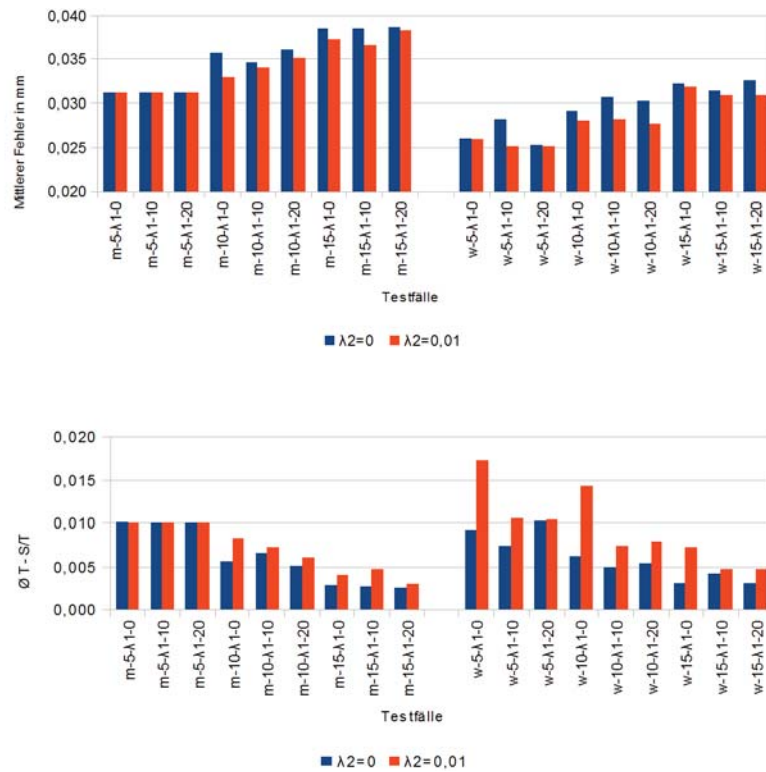
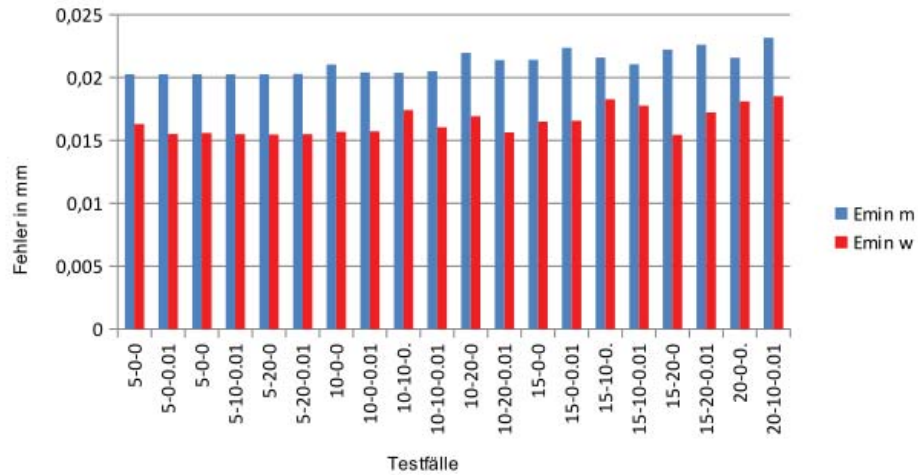


Abbildung 24: Der Einfluss des Faktors für Formbeschränkung λ_2 auf den mittleren Fehler 24(a) und die Differenz der ICP mit Formparametern und ohne 24(b).

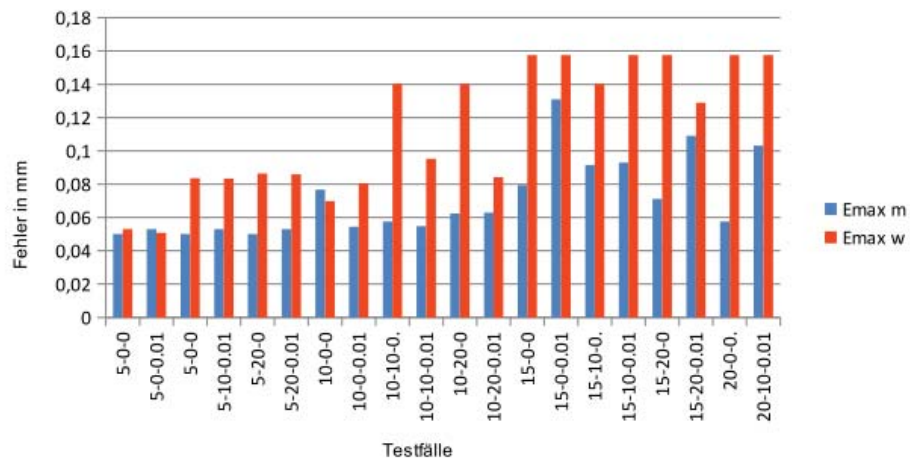
kann, wenn das Zieldatum gut ausgerichtet ist und nicht zu stark vom Mittelwert, also dem Formmodell, abweicht. Im Mittel jedoch zeigt sich aus den vorherigen Diagrammen, sind einige Varianten besser als andere. Der Maximalwert steigt mit wachsender Formparameteranzahl insbesondere bei den Frauendaten. Die anfängliche Ausrichtung ist bei diesen Daten schlechter als bei den Männerdaten. Bei den Frauendaten ist eine Tendenz zu einer Verbesserung des Maximalwertes bei Vorhandensein von λ_2 zu erkennen, die bei den Männerdaten eher umgekehrt ausfällt.

Abschließend lässt sich feststellen, dass bei reiner Analyse der Fehlerergebnisse wenige Formparameter vielen vorzuziehen sind, da sie in allen statistischen Größen aus den oben genannten Gründen die besten Ergebnisse liefern.

Optisch sind die Ergebnisse sehr verschieden. Modellerte Köpfe mit wenig Parametern weisen meist starke Regionengrenzen auf, da aufgrund geringer Variabilität die vorhandenen Parameter ausgereizt werden müssen. Viele Parameter liefern ein zum Teil sehr gutes Ergebnis, aber auch völlige Deformationen, da das Lösen einer großen Matrix schwieriger ist und zu unrealistischen Ergebnissen führen kann. Für eine optische Auswertung lässt sich nicht sagen, welche der getesteten Fälle das insgesamt beste Ergebnis liefern. Bei jeder Konstellation gibt es sowohl sehr gute als auch schlechte Ergebnisse. Einige Beispiele sind in Abbildung 26 zu sehen.



(a) Kleinster Fehler der Testfälle



(b) Größter Fehler der Testfälle

Abbildung 25: Minimum und Maximum der Testfälle

Die Ergebnisse dieser 3D Kostenfunktion lassen dennoch Schlüsse für die Parameterwahl der 2D-3D Registrierung zu. Als Startwert scheinen 10 Formparameter geeignet, da sie genügend Variabilität erzeugen, aber die Matrix noch überbestimmt bleibt. Für die Regionengrenzen ist ebenfalls 10 der Startwert. Das ist ein gutes Mittel, bei dem die meisten Übergänge relativ glatt erscheinen. λ_2 bleibt vorerst außen vor, da sich keine deutliche Tendenz gezeigt hat.

7.3 Ergebnisse der Profillinienrekonstruktion

In dieser Arbeit werden zwei Ansätze zur Rekonstruktion eines 3D Kopfes aus Fotos vorgestellt. Beide beginnen zuerst mit der Anpassung des Modells an die Profillinie des Fotos. Wie auch bei der 3D Registrierung wird erst die Transformation des Modells optimiert und daraufhin die Form und Transformation.

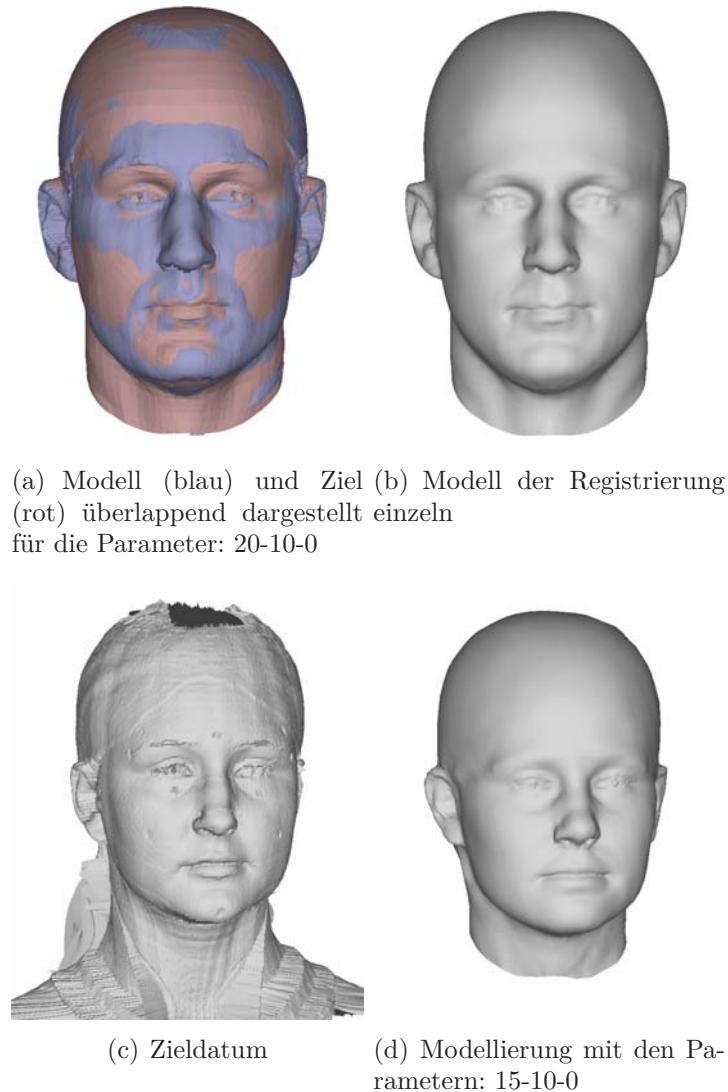


Abbildung 26: Beispielergebnisse der 3D-3D Registrierung

Der erste Schritt beginnt bei einem grob ausgerichteten Formmodell an die Profilinie, indem die Nasenspitze an den Koordinatenursprung ausgerichtet wird und die Nasenspitze ebenfalls in der Profilinie markiert und die Landmarken dementsprechend transformiert. Die Optimierung läuft iterativ in kleinen Schritten ab. Das Diagramm 27 zeigt die Fehlerentwicklung bei der Optimierung. Es wurden 38 Bilderpaare getestet. Die Personen haben meist kurze Haare oder Frisuren, bei welchen die Kopfform erkennbar ist. Offene oder lange Haare eignen sich schlecht für die Platzierung der Landmarken. Hier muss die Form des Hinterkopfes geschätzt werden. Die verwendete Datenbank für die Erstellung des Formmodells beinhaltet alle 144 fehlerfreien Scans (siehe Abschnitt 7.1) um eine größtmögliche Formvariabilität zu erhalten.

Im Diagramm 27 erkennt man die durchschnittliche Fehlerentwicklung von 24 Datensätzen der Anpassung des Formmodells an die Profilinie in Transformation, Rotation und Skalierung. Die X-Achse stellt die Anzahl der Iterationen dar und die Y-Achse den Fehler. Der Startwert ist im Diagramm nicht dargestellt. Dieser liegt im Durch-

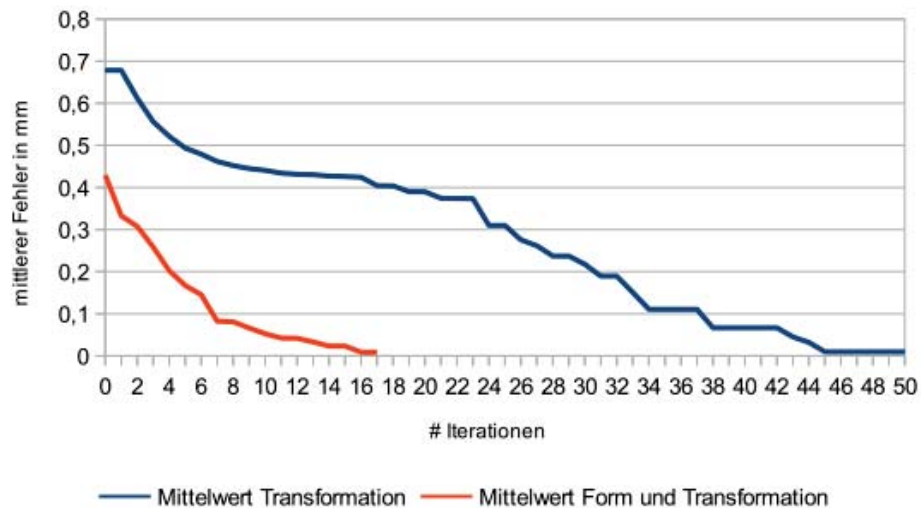


Abbildung 27: Durchschnittliche Fehlerentwicklung über alle Bilder der Profillinienrekonstruktion mit Testfall 10-10-0. Das Diagramm zeigt die Entwicklung der Transformation und des darauffolgenden Schrittes der ICP mit Form und Transformation.

schnitt bei 12 mm, was nicht im gleichen Diagramm dargestellt werden kann. Die größte Verbesserung wird somit in der ersten Iteration erreicht. Nach durchschnittlich sieben Iterationen gibt es kaum sichtbare Verbesserungen des Fehlers. Die einzelnen Testfälle brauchen unterschiedlich lange bis zur Terminierung. In den getesteten 38 Fällen werden minimal 17 Iterationen und maximal 58 Iterationen benötigt. Im Durchschnitt benötigt ein Datensatz 34 Iterationen. Dabei wird insgesamt ein mittlerer Fehler von 8 mm erreicht, verteilt auf 125 Profilvertices. Das macht im Mittel eine Abweichung von 0,064 mm pro Vertex. Ein Abstand von 1 cm ist in Abbildung 28 als orangene Linie an der Nasenwurzel im Verhältnis zum Gesamtmodell verdeutlicht.

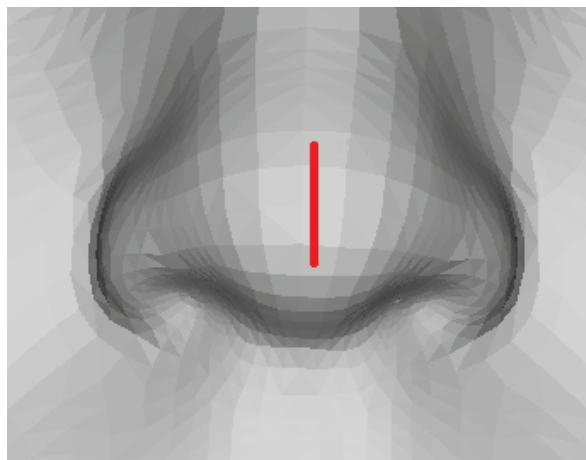


Abbildung 28: Die orange Linie zeigt den durchschnittlichen Fehler von 1 cm im Modell.

Das Diagramm 27 zeigt weiterhin die Fehlerentwicklung im darauffolgenden Schritt, der Anpassung des Modells in Form und Transformation an die Profillinie, auch wieder beispielhaft für einen Teil der getesteten Daten. Hier wird bereits nach durchschnittlich 3 Iterationen die größte Entwicklung verzeichnet. Die folgenden Iterationen bewirken meist nur sehr geringe Verbesserungen des Fehlers. Die für dieses Diagramm verwendeten Parameter sind $\lambda_1 = 10$ und $\lambda_2 = 0$ bei 10 Formparametern.

Die Parameterkombinationen fallen bei der Anpassung an die Profillinie geringer aus, da beachtet werden muss, dass die Matrix der Kostenfunktion überbestimmt bleibt. Einige Regionen sind bei der Profillinie nicht vorhanden - Augen, Ohren, Wangen. Daher hat die Matrix an diesen Positionen keine Einträge und kann nicht berechnet werden. Um dies zu vermeiden, sollte der Parameter für die Glattheit der Regionengrenzen λ_1 zumindest einen kleinen Wert erhalten. Das führt dazu, dass alle Regionen durch ausreichend Gleichungen in der Matrix vertreten sind und diese damit lösbar wird, ohne künstliche Informationen zu verwenden, die nicht in der Profillinie vorhanden sind.

Es wurden die Kombinationen für $\lambda_1 = 1; 10; 20$ und $\lambda_2 = 0; 0,5$ für 5, 10 und 15 Formparameter im Hinblick auf Anzahl der Iterationen und den finalen Fehler getestet. Die durchschnittlichen Ergebnisse für alle 38 Bilddatensätze sind in den Diagrammen 29 und 30 zu sehen. Bei dem durchschnittlichen Fehler (Diagramm 29) fällt auf, dass im Durchschnitt nur 11 von 18 Testfällen (grün) eine Verbesserung des Fehlers durch die Formparameter aufweisen. Die übrigen Werte (rot) liegen zum Teil deutlich über dem bereits durch die Transformation erreichten Fehler (gelb). Diese hohen Werte werden durch Ausreißer verursacht, bei denen die erste Iteration mit Formparametern bereits eine vollständige Deformation verursacht. Von diesen Ausreißern gibt es einen bis maximal sechs bei der Konstellation 15 Formparametern, $\lambda_1 = 1$, $\lambda_2 = 0$. Diese Ausreißer sind darauf zurückzuführen, dass die Matrix mit mehr Unbekannten auch mehr Gleichungen benötigt, um nach den Unbekannten zu lösen. Das ist bei einigen Konstellationen und Datensätzen nicht der Fall, weswegen das vom Programm durchgeführte Lösen des Gleichungssystems versagt.

Werden diese Ausreißer jedoch nicht berücksichtigt, ergibt sich das Fehlerbild mit dunkelroten Balken. Es zeigt sich, dass nur noch 4 von 18 Testfällen mit Formänderung knapp schlechter sind, als die Transformation allein. Aus diesem Diagramm wird ersichtlich, dass für die Anpassung der Profillinie weniger Parameter ein besseres Ergebnis liefern und bei mehr Parametern die Nebenbedingungen wichtig werden. Sie müssen zunehmend höher gewählt werden, damit die Kostenfunktion lösbar bleibt.

Die Diagramme in Abbildung 30 zeigen die durchschnittliche Anzahl der Iterationen, die berechnet werden, bis der Fehler nicht mehr kleiner wird. Aus allen Testfällen wird deutlich, dass die Anzahl der Iterationen mit zunehmenden λ_1 steigt. λ_2 wirkt sich bei 10 und mehr Formparametern kaum auf die Anzahl der Iterationen aus. Eine Besonderheit stellen 5 Formparameter dar, bei welchen die Anzahl der Iterationen tausendfach höher ist, als bei den anderen Fällen. Weiterhin hat λ_2 deutlich größere

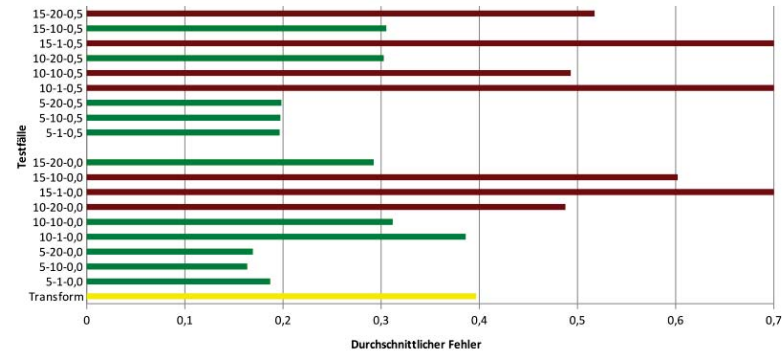


Abbildung 29: Durchschnittlicher Fehler der Anpassung des Formmodells an die Profillinie. Der Ausgangsfehler nach der Transformationsberechnung ist gelb dargestellt. Die Parameterkonstellationen mit Fehlerverbesserungen sind grün und die übrigen rot dargestellt. Die Fehlereinheit ist wieder Millimeter.

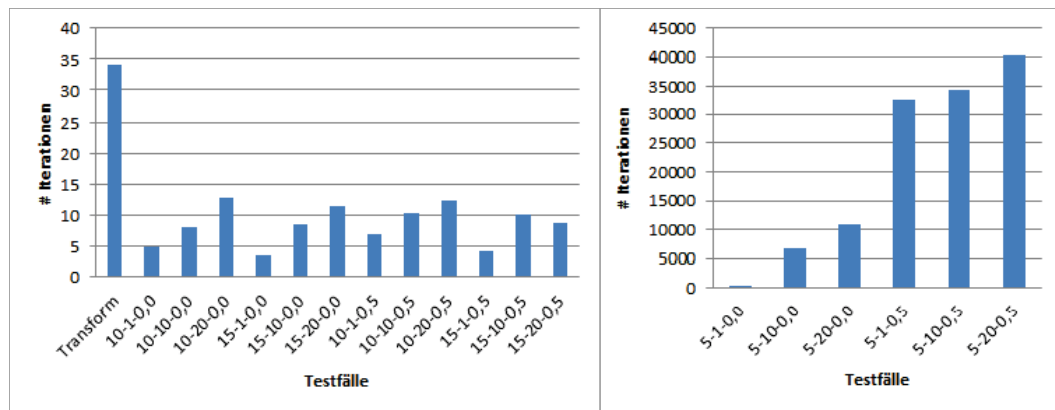


Abbildung 30: Durchschnittliche Anzahl der Iterationen bei der Anpassung an die Profillinie

Auswirkungen. Diese hohe Anzahl der Iterationen ist ebenfalls wieder auf die Matrixgröße und das damit verbundene Lösen des Gleichungssystems zurückzuführen. Das System hat bei 5 Formparametern 70 Unbekannte weniger als bei 15 Formparametern (aufgrund der 7 Regionen). Ohne die Gleichungen der Nebenbedingungen gibt es nur 125 Vertexgleichungen. Das hat zur Folge, dass die Berechnung durch den Computer bei 5 Formparametern wesentlich besser gelöst werden kann, als bei 15.

Betrachtet man Fehler und Iterationen, lässt sich feststellen, dass viele Iterationen nicht zu einem besseren Fehler führen. Den kleinsten Fehler erreichen die Testfälle mit 5 Formparametern. Trotzdem können auch die Konstellationen 10-10-0, 15-20-0 und 10-20-0,5 ein gutes Ergebnis liefern.

Abbildung 31 zeigt ein an die Profillinie in Transformation und Form angepasstes Modell. Es sind nur an einigen Stellen geringe Abweichungen des Modells zu der Profillinie erkennbar.

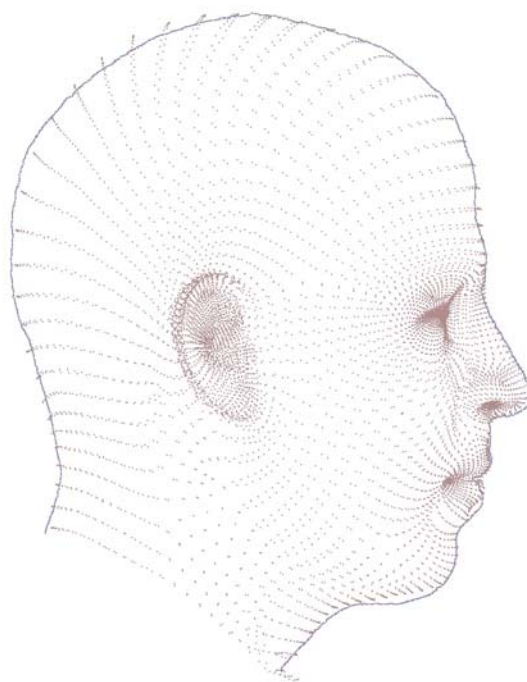


Abbildung 31: Anpassung des Formmodells an eine Profillinie

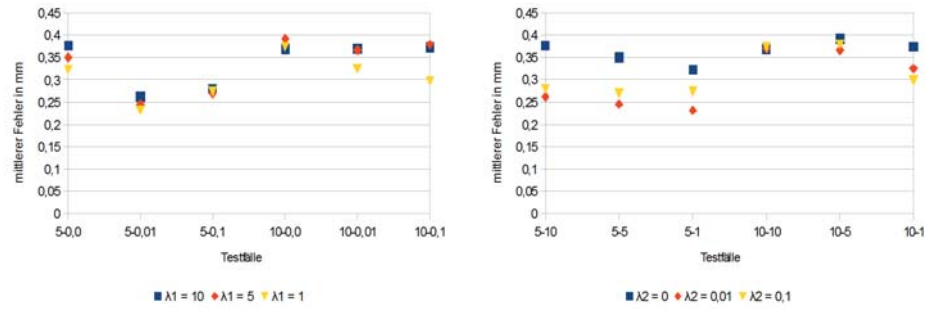
7.4 Ergebnisse von Ansatz 1

Der erste Ansatz für die Modellierung des Formmodells an die frontalen Landmarken schließt sich an die Profillinienrekonstruktion an. Die Idee ist, das Formmodell an die Profil- und Frontlandmarken anzupassen ohne die Position der Tiefenkoordinate der Frontallandmarken zu kennen. Durch eine ICP werden Parameter berechnet, anhand derer das gesamte Formmodell rekonstruiert werden kann.

Für die Bestimmung der Güte dieses Algorithmus werden die erreichten Fehler mit verschiedenen Parameterkonstellationen bestimmt und dargestellt. Weiterhin gibt es beispielhaft zwei optische Auswertungen am Foto bzw. erstellten Modell.

Auch hier können wieder die bekannten Parameter λ_1 und λ_2 und die Anzahl der Formparameter variiert werden. Zusätzlich dazu gibt es einen weiteren Parameter für die Gewichtung der Frontlandmarken. Werden diese nicht extra gewichtet, erlangen die Profillandmarken ein zu starkes Gewicht, da sie in größerer Anzahl vorliegen. Der Algorithmus wird an 37 Bildpaaren mit Frontal- und Profilaufnahme getestet.

Da in der Datenbank des Formmodells wenig Variation in Geschlecht und Alter vorliegen und zudem in einigen Bildern durch die Frisuren die Landmarken schlecht zu markieren sind, gibt es einige Bilder, bei denen die ICP ein ungenügendes Ergebnis liefert. Diese ausreißenden Datensätze werden in den Diagrammen nicht berücksichtigt. Die Anzahl der Ausreißer steigt mit Anzahl der Formparameter und fällt mit steigender Formbeschränkung. Insgesamt sind 13,8% aller getesteten Rekonstruktionen zu diesen Daten zu zählen, wobei zum Beispiel auch Kinder rekonstruiert wurden, obwohl diese nicht im Formmodell vorhanden sind.



(a) Einfluss von λ_1 auf den mittleren Fehler
(b) Einfluss von λ_2 auf den mittleren Fehler

Abbildung 32: Ergebnis von Ansatz 1: Die Testfälle haben die Beschriftung: Formparameter - λ ; mit λ_2 in Abbildung 32(a) und λ_1 in Abbildung 32(b).

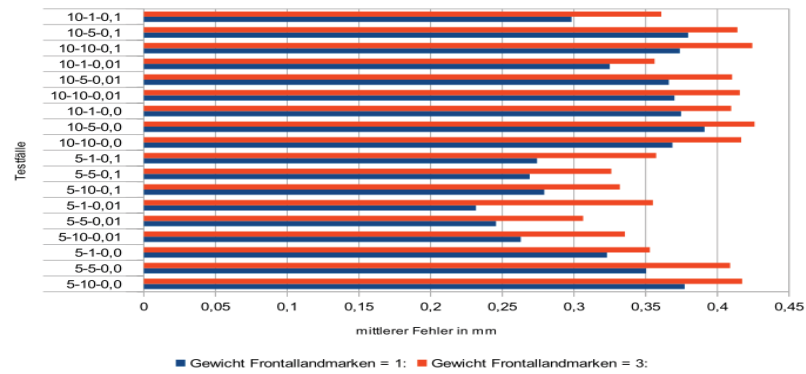


Abbildung 33: Ergebnis von Ansatz 1: Einfluss des Gewichtes der Frontallandmarken auf den mittleren Fehler pro Vertex.

Die Darstellung des mittleren Fehlers pro Vertex ist wieder in Diagrammen dargestellt. Dieser durchschnittliche Fehler wurde als euklidischer Abstand von markierten Landmarken und Vertices im Modell berechnet. Dabei sind von den Frontallandmarken nur zwei Koordinaten bekannt, sodass der Abstand nur in 2D berechnet wurde. Ein üblicher Fehler für einen Vertex liegt bei 0,32 mm bei einem Frontallandmarkengewicht von 1 und bei 0,38 mm bei einem Gewicht von 3, wobei der kleinste getestete Wert 0,23 mm beträgt und der größte 0,43 mm.

Im Diagramm 32 ist der Einfluss der Parameter der Nebenbedingungen dargestellt. Daraus geht hervor, dass λ_1 wie zu erwarten einen geringen Einfluss auf den Fehler hat, wobei eine die schwächste Beschränkung zu dem kleinsten mittleren Fehler führt. Bei der Formbeschränkung λ_2 ist dies anders. Keine Beschränkung führt dazu, dass beim Lösen der ICP zu große Parameter entstehen, wodurch ein lokales Minimum übersprungen werden kann. Interessanterweise führt eine kleine Beschränkung von 0,01 zu besseren Ergebnissen als eine von 0,1.

Ein weiterer getesteter Parameter ist der Einfluss des Gewichtes der Frontallandmarken. Also wie stark sich das Formmodell an die wenigen Gesichtslandmarken anpassen

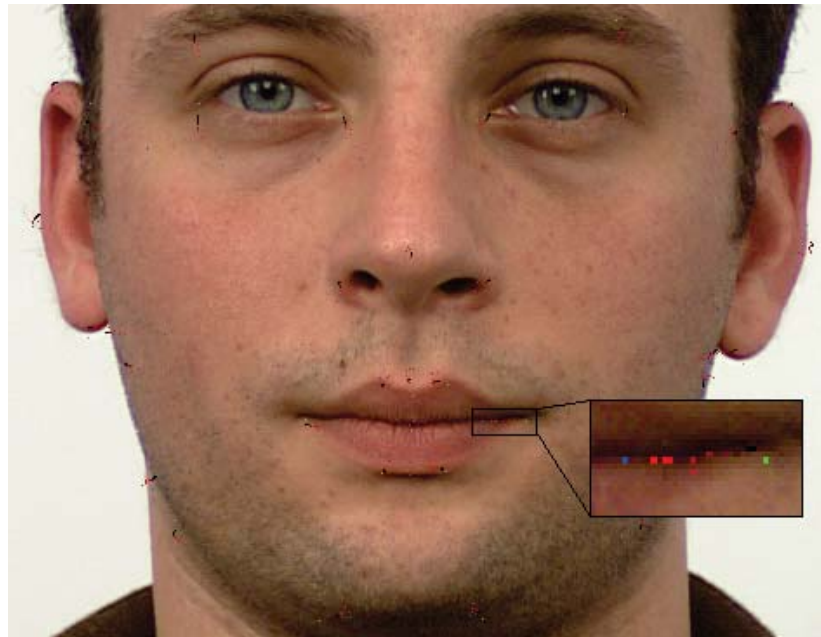


Abbildung 34: Ergebnis von Ansatz 1: Die Anpassung der Modellvertices (rot) an die Ziellandmarken (grün) ausgehend von der Ausgangsposition (blau).

soll. Wird dieses Gewicht größer, steigt der mittlere Fehler, wie im Diagramm 33 zu sehen ist.

Neben diesen zahlenmäßigen Betrachtungen des Fehlers, muss eine Rekonstruktion des Kopfes natürlich auch optisch ausgewertet werden. Denn die rechnerisch kleinsten Fehler führen natürlich nicht immer auch zu den schönsten Ergebnissen. Die erste Variante der Auswertung zeigt Abbildung 34. Hier ist die Vertexverschiebung des Formmodells an den beobachteten Landmarken dargestellt. Es ist zu sehen, dass sich die Vertices von ihren Ausgangspositionen in Richtung der Ziele verschieben. Allerdings gibt es einige Landmarken, die sich nur schlecht an die Vorgaben anpassen. Das sind insbesondere die Ohren, Hals und Kopflandmarken. Erklärbar ist das durch die angenommene orthografische Projektion der Fotos, was natürlich nur eine grobe Annäherung an die eigentliche perspektivische Aufnahme darstellt. Dadurch befinden sich die Ziellandmarken im Zentrum der Aufnahme noch fast an den tatsächlichen Positionen. Dieser Fehler wird in der Peripherie am größten.

Zuletzt zeigt Abbildung 35 neue Perspektiven des fertigen Modells im Vergleich zu den originalen Fotos einmal mit und ohne Textur.

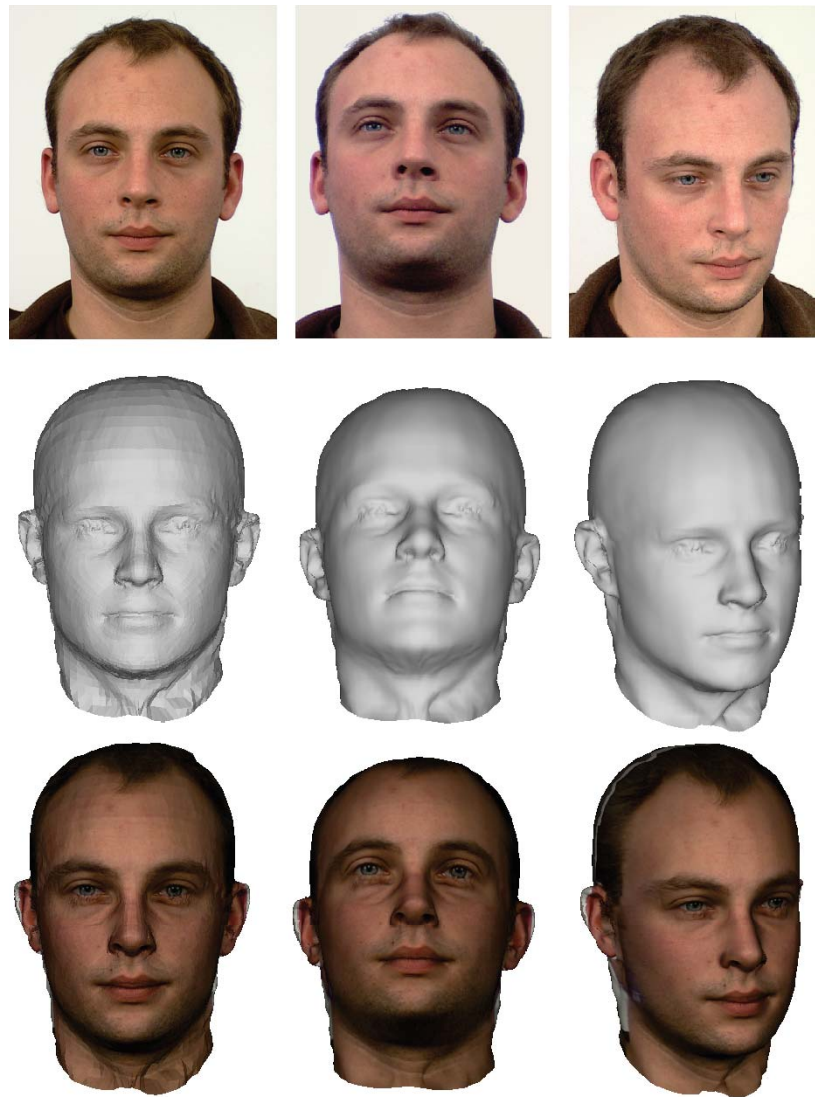
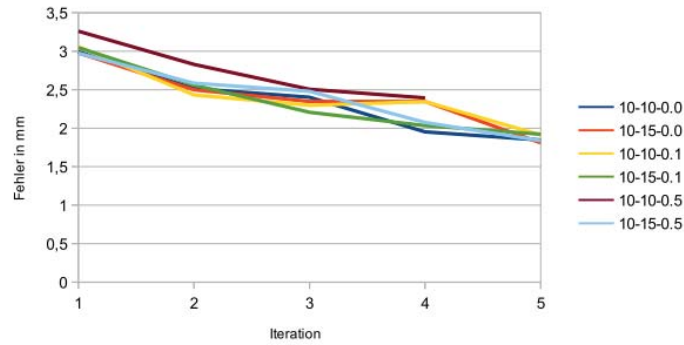
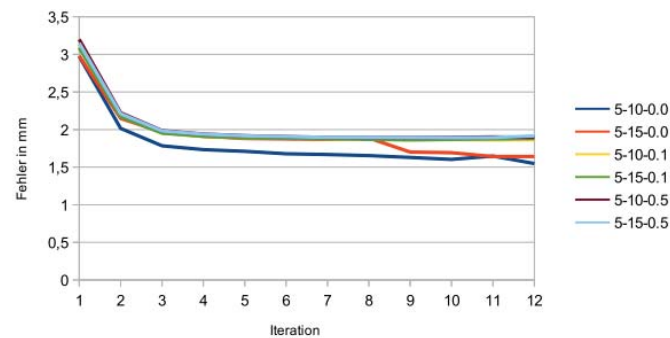


Abbildung 35: Ergebnis von Ansatz 1: Neue Perspektiven des rekonstruierten Modells.



(a) Testfälle mit 10 Formparametern



(b) Testfälle mit 5 Formparametern

Abbildung 36: Ergebnis Ansatz 2: Das Diagramm zeigt den mittleren Fehler der einzelnen Iterationen gemittelt über alle Testdatensätze.

7.5 Ergebnisse von Ansatz 2

Nachdem die Profillinie angepasst wurde, wird die Frontansicht des Formmodells in diesem Ansatz mit Hilfe von Thin Plate Splines und dem ICP Algorithmus an die Landmarken der Fotos angeglichen. In jeder Iteration werden die zu den Landmarkenpositionen korrespondierenden Vertices mittels TPS an die aus den Bildern bekannten Positionen verformt. Dafür werden sowohl die frontalen als auch die Profillandmarken verwendet. Daraufhin wird versucht, das Formmodell auf dieses TPS Modell mit Hilfe des ICP anzupassen.

Es wird wieder der Einfluss verschiedener Parameter auf die Anzahl der Iterationen und des Fehlers bei der ICP überprüft. Weiterhin wird die tatsächliche Position der fehlenden Koordinate der Frontlandmarken manuell im Profilbild markiert und mit der berechneten Position der Vertices des Formmodells verglichen. Zuletzt werden die Ergebnisse optisch ausgewertet.

In Anbetracht der Tatsache, dass auch hier wieder eine 3D-3D Anpassung von Formmodell an TPS-Modell gemacht wird und im Hinblick auf die Ergebnisse aus Abschnitt 7.2, werden nur die besseren Parameterkonstellationen getestet. Die Parameter sind: 5 und 10 Formparameter, $\lambda_1 = 10; 15$, $\lambda_2 = 0; 0, 1; 0, 5$. Es wurden 37 Datenpaare getestet.

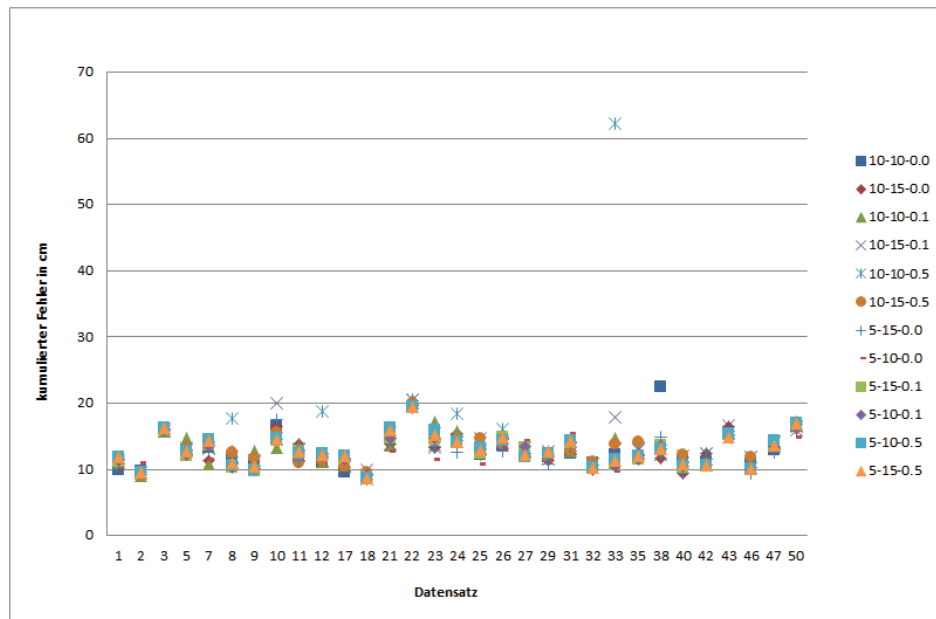


Abbildung 37: Ergebnis Ansatz 2: Kumulierter Fehler der Abstände der Landmarken zu den korrespondierenden Vertices.

In den Diagrammen in Abbildung 36 sind die Ergebnisse des über alle Bilder pro Iteration gemittelten Fehlers für die oben genannten Parameterkonstellationen. Es ist zu erkennen, dass in allen Fällen der Fehler zu Beginn sehr groß ist (zwischen 2 und 3 mm pro Vertex) und innerhalb von 3-5 Iterationen bereits fast minimal ist (1-2 mm). Nur bei 5 Formparametern werden wie in allen bereits getesteten Algorithmen wieder deutlich mehr Iterationen gebraucht. Durchschnittlich liegt die Anzahl der Iterationen bei 10 Formparametern bei 3,5 und bei 5 Formparametern bei 12-20, abhängig von der Größe von λ_2 . Auch wird wieder deutlich, dass der rechnerische Fehler in erster Linie von der Anzahl der Formparameter abhängt.

Der zweite Test ist die Messung der Differenzen der Z-Werte der Frontallandmarken. Dafür werden im Profilbild manuell die Punkte der rechten Seite markiert und die deren Tiefe als Abstand zur Nasenspitze interpretiert. Diese Werte werden mit den berechneten Vertexpositionen des Formmodells verglichen. Von den 38 Landmarken gibt es 19 markierbare in einem Profilbild. Für die Abstandsmessung wurde der einfache euklidische Abstand gewählt. Die kumulierten Abstände sind in Abbildung 37 dargestellt. Ein durchschnittlicher Fehler für eine Markierung ist 5,2 mm.

In dem Diagramm erkennt man, dass einige Bilder besser durch das Formmodell interpretiert werden können als andere. In diesen Datensätzen ist die Wahl der Parameterkonstellationen wenig relevant für den entstehenden Fehler. In anderen Datensätzen dagegen schwankt der Fehler stark. Dennoch gibt es für jeden Datensatz bestimmte Konstellationen, die zu einem guten Ergebnis führen. In der Abbildung 38 ist der mittlere Fehler jeder Konstellation zu erkennen. Der Fall 10 Formparameter, $\lambda_1 = 10$ und $\lambda_2 = 0,5$ erreicht dabei den schlechtesten Wert und die Konstellation 5 Formparame-

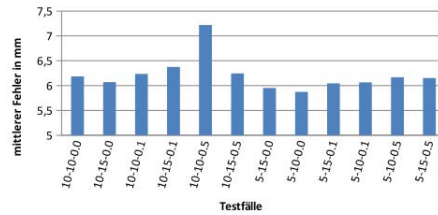


Abbildung 38: Abstand der Vertices von den eigentlichen Landmarkenpositionen gemittelt über alle Datensätze.

ter, $\lambda_1 = 10$ und $\lambda_2 = 0$ den besten. Allgemein sind die Schwankungen der einzelnen Testfälle untereinander jedoch nicht besonders groß.

Diese Fehler sind recht groß im Vergleich zu den Fehlern, die bei einer 3D-3D Rekonstruktion erreicht wurden (Abschnitt 7.2). Das liegt an der gewählten Auswertungsmethode. Es wurde versucht, die frontalen Landmarken von der Seite zu markieren. Dies gelingt jedoch schlecht an der Kinnpartie und an den Ohren, da es keine markanten Punkte gibt, die sowohl in der Frontal-, als auch in der Profilaufnahme zu erkennen sind. Daher bergen diese Punkte bereits einen Fehler in der Markierung unabhängig von der Anpassung des Modells an die Landmarken. Das ist in Abbildung 39 zu erkennen, denn in dieser Profilaufnahme stimmt auch die X-Koordinate nicht mit den Markierungspunkten überein, obwohl diese anhand der Thin Plate Splines auf die exakte Landmarkenposition des Frontalbildes gezogen wurde. Die Unterschiede an Lippen und Augen sind recht gering, wohingegen die Unterschiede am Kinn sehr groß sind, das lässt auf einen Fehler bei der manuellen Profilbildmarkierung schließen..

Aus diesem Grund werden die Ergebnis auch am Modell mit Textur ausgewertet, um zu sagen, ob der beste Fehlerwert auch das beste Ergebnis liefert und inwieweit das Modell an den vermeintlich falschen Markierungspunkten auf die Textur passt. In Abbildung 40 ist das Ergebnis des besten Vergleichswertes dargestellt. Es ist zu erkennen, dass zwar die Landmarkenpositionen gut mit dem Formmodell übereinstimmen, das Ergebnis optisch jedoch nicht überzeugt, da die Regionengrenzen viel zu stark sichtbar sind. Die Ergebnisse sind deutlich besser, wenn λ_1 mindestens einen Wert von 10 hat, damit die Grenzen geglättet werden. Auch wenn dies mit einem rechnerisch etwas größeren Fehler einher geht.

Weitere Ergebnisse, die auch optisch überzeugen, sind im Anhang im Anhang ab Seite 85 zu sehen. Es zeigt sich, dass die Ergebnisse stark von dem zugrunde liegenden Datensatz für die Erstellung des Formmodells abhängen. In dem hier verwendeten Datensatz befindet sich eine überdurchschnittlich große Zahl Männer mit angloamerikanischer Herkunft. Eine ethnische Vielfalt ist kaum vorhanden. Auch Frauen sind in der Datenbank unterrepräsentiert. Aus diesem Grund klappt die Anpassung in einigen Fällen besser.

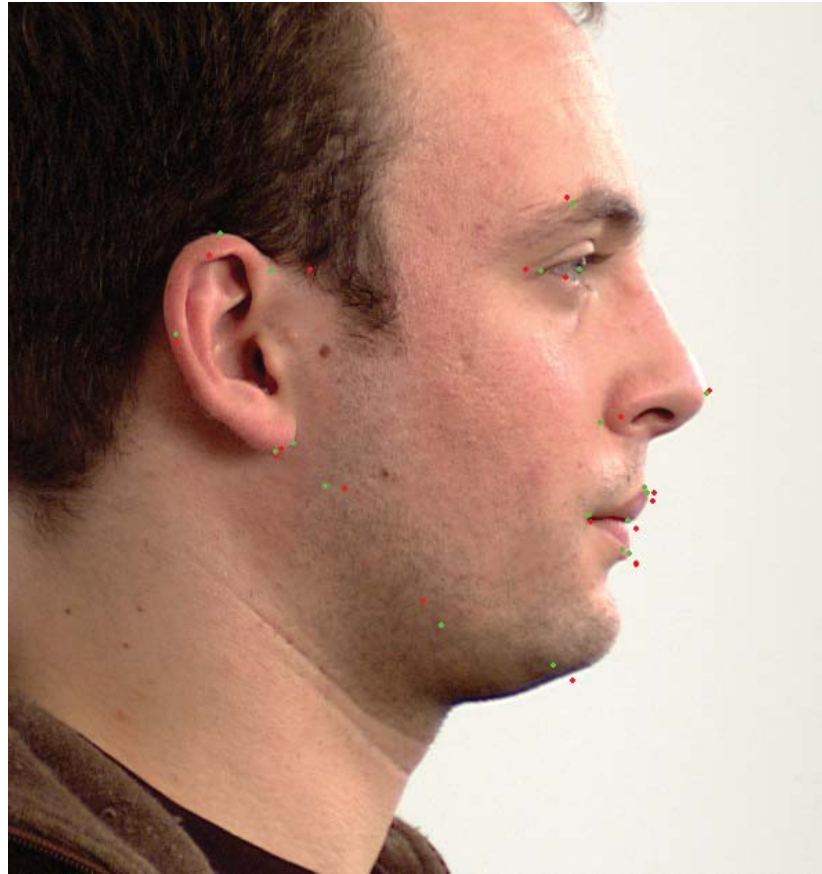


Abbildung 39: Ergebnis Ansatz 2: Unterschied der Z-Koordinate von Modell (rot) und der manuell markierten korrespondierenden Position im Bild (grün).

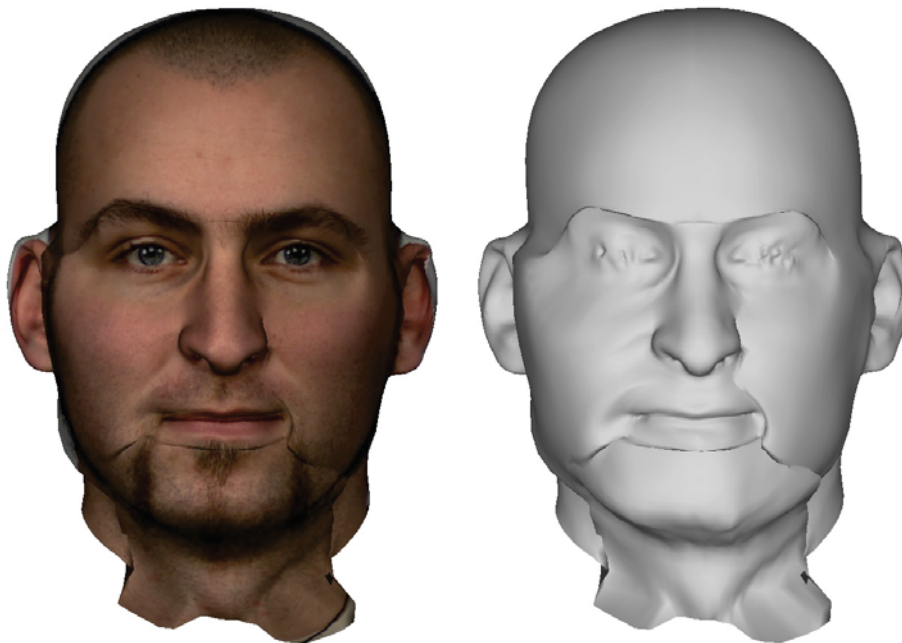


Abbildung 40: Ergebnis Ansatz 2: Zeigt das Ergebnis der TPS-ICP Anpassung des Formmodells mit der geringsten Abweichung zu den gesetzten Landmarken.

Trotzdem gibt es doch eine Ähnlichkeit der Modelle zu den entsprechenden Fotos. Das liegt nicht zuletzt auch an der Thin Plate Spline Verformung, die unabhängig von Datensätzen oder gelernten Modellen berechnet werden kann. Die hier gezeigten Ergebnisse präsentieren eine große Vielfalt im Hinblick auf Herkunft, Geschlecht und Alter.

7.6 Laufzeiten

Alle Tests wurden auf einem Rechner mit zwei 2,67GHz Intel(R) Xeon(R) X5550 Prozessoren und 12 GB Arbeitsspeicher auf einem Windows 7, 64 Bit System durchgeführt.

3D-3D Rekonstruktion

Die Laufzeit der 3D-3D Registrierung ist abhängig von der Wahl der Parameter. Ohne Nebenbedingungen ist die Laufzeit am kürzesten, das Ergebnis jedoch auch schlecht. Je größer die Werte für die Nebenbedingungen ausfallen, desto langsamer wird die Berechnung. Dabei fällt die Formbedingung stärker ins Gewicht als die Überlappungsbedingung. Auch die Anzahl der Formparameter wirkt sich aus. Interessanterweise haben 10 Parameter die kürzeste Laufzeit. Bei weniger als 10 und mehr als 10 Parametern verlängert sich die Berechnungsdauer. Die kürzesten Zeiten liegen bei durchschnittlich ca. 10 Sekunden für 10 Parameter und Überlappungsfaktor 10, 5 Parameter bei gleichem Überlappungsfaktor brauchen schon ca. 30 Sekunden pro Datensatz und 15 Parameter ca. 15-20 Sekunden. Der doppelte Überlappungsfaktor verdoppelt bis vervierfacht die Zeit abhängig vom Datensatz.

Profillinienrekonstruktion

Die Laufzeit einer Profillinienrekonstruktion ist ebenfalls abhängig von der Anzahl der Formparameter und der Restriktionen durch λ_1 und λ_2 und damit von der Anzahl der Iterationen, die zur Bestimmung des geringsten Fehlers benötigt werden. Die längste Laufzeit wurde bei 5 Parametern, $\lambda_1 = 0,5$ und $\lambda_2 = 20$ erreicht. Hier betrug der durchschnittliche Wert 13:10 Minuten. Die Konstellationen mit 5 Formparametern und $\lambda_1 = 0$ sortieren sich zwischen 6 Sekunden und 3:51 Minuten bei der durchschnittlichen Laufzeit ein. Sowohl 10 als auch 15 Parameter haben mit den Konstellationen $\lambda_2 = 1; 10; 20$ und $\lambda_1 = 0; 0,5$ jeweils eine Laufzeit von 2-3 Sekunden. Das wird auch durch die annähernd gleichen Iterationsanzahlen deutlich.

Ansatz 1: ICP der Landmarken ohne frontale Tiefeninformation

Die Laufzeit des Algorithmus ist ebenfalls von der Anzahl der Iterationen abhängig. Die meisten Iterationen benötigen die Testfälle mit $\lambda_2 \neq 0$, wobei hier die Fälle mit 5 Formparametern nochmals um ein Vielfaches mehr Iterationen brauchen. Auch hängt die Anzahl stark von der Gewichtung der Frontallandmarken ab. So beträgt die durchschnittliche Anzahl der Iterationen bei 10 Formparametern bei Gewicht 1 4,6 und

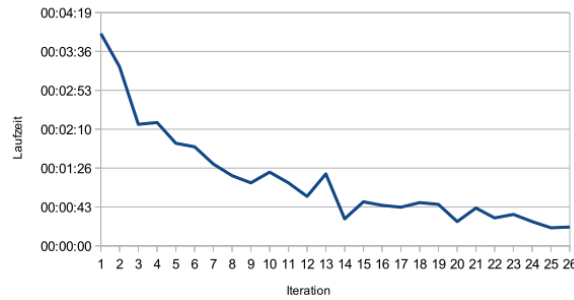


Abbildung 41: Laufzeitentwicklung der einzelnen Iterationen der TPS-ICP Verformung

bei Gewicht 3 2,8. Bei 5 Formparametern mit $\lambda_2 \neq 0$ bei Frontalgewicht 1 über 1000 Iterationen und bei Gewicht 3 nur bei 319. Dementsprechend variiert auch die Laufzeit von 1-2 Sekunden bei den wenigen Iterationen bis zu 28 Sekunden bei maximaler Iterationszahl. Hierbei ist zu sagen, dass der Algorithmus aus Performanzgründen bei 1000 Iterationen abbricht, da sich der Fehler nicht mehr merklich ändert.

Ansatz 2: TPS-ICP Algorithmus

Hier wird für jeden Datensatz die gesamte Laufzeit und die Laufzeit pro Iteration, also einen Durchlauf TPS-Verformung und ICP-Anpassung, bestimmt. Hier ist es ebenfalls wie in den anderen Algorithmen zuvor so, dass die Variante mit 5 Formparametern am meisten Iterationen und am längsten braucht. Der maximale Fall von 5 Formparametern, $\lambda_1 = 10$ und $\lambda_2 = 0,5$ benötigt pro Iteration 55 Sekunden im Schnitt und ein Datensatz wird in durchschnittlich 17:19 Minuten bearbeitet. Ähnlich ist es bei der Variante mit $\lambda_1 = 15$ und $\lambda_2 = 0,5$. Bei gleichen Formparametern und λ_1 aber $\lambda_2 = 0,1$ halbiert sich die Laufzeit auf 8:40 Minuten. Bei $\lambda_1 = 0$ verkürzt sich die Laufzeit nochmal auf 50 Sekunden bei $\lambda_1 = 10$ und auch 25 Sekunden für $\lambda_1 = 15$. 10 Formparameter haben in allen Varianten wieder eine Laufzeit von ca. 2 Sekunden.

Interessant ist auch die Laufzeitentwicklung der Iterationen innerhalb eines Datensatzes. Diese ist in Abbildung 41 dargestellt. Da die TPS-Verformung immer den gleichen Zeitaufwand bedeutet, zeigt sich, dass der ICP Algorithmus mit zunehmender Iterationsanzahl schneller terminiert bis er schließlich keine Verbesserung des Modells mehr erzeugen kann.

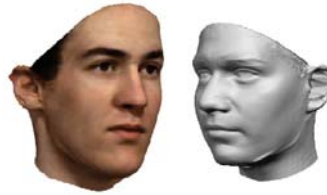


Abbildung 42: links: Texturiertes Modell von Blanz und Vetter, vgl. [BV99], Seite 4, rechts: Modell nach Blanz und Vetter ohne Textur, vgl. [BKK⁺08], Seite 8

8 Auswertung

Die Güte der hier vorgestellten Ergebnisse der 3D Rekonstruktion des Kopfes hängt von verschiedenen Faktoren ab, die im Laufe des Algorithmus Einfluss auf die Berechnungen haben.

Als ersten Aspekt ist hier das Formmodell zu nennen. Von diesem hängt die Variabilität und die Genauigkeit der Rekonstruktion ab. Das hier entwickelte Modell weist im Vergleich zu bisherigen Arbeiten auf diesem Gebiet einige Vorteile auf. Zum einen ist es ein ganzheitliches Modell. Im allgemein als Vorreiter für Formmodelle des Kopfes benannten Modell von Blanz und Vetter [BV99] fehlt der gesamte Schädel, es wird nur auf den Gesichtsdaten gearbeitet (siehe Abbildung 42). Gleiches findet sich bei Jiang[JHY⁺05].

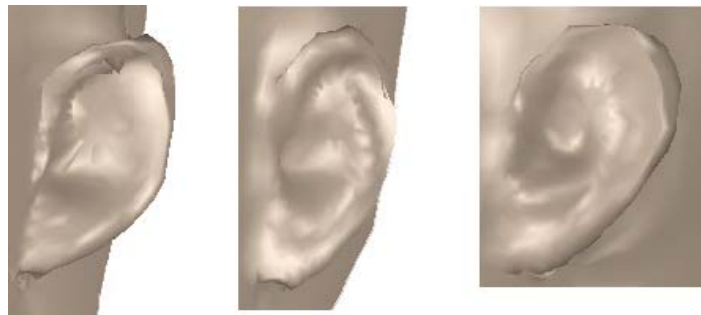
Eine weitere Verbesserung des Modells sind die Ohren. Diese sind in bisherigen 3D Rekonstruktionen entweder ausgespart, sehen fast immer gleich aus oder haben eine unnatürliche Form der Ohrmuschel. Die Ohren der Modelle in Schneider [SE09] und Breuer et.al.[BKK⁺08] sind in den Abbildungen 43(a) und 43(b) zu sehen. Die Ohren des Formmodells aus dieser Arbeit in Abbildung 43(c). Es ist zu erkennen, dass individuelle Feinheiten sowie eine natürlichere Form bei den Ohren des Formmodells und seiner Ausprägungen in dieser Arbeit besser ausgearbeitet sind. Das liegt nicht zuletzt an in Abschnitt 3.3 beschriebenen Anpassung der Originalscans an die Topologie des Referenzkopfes. Hier wurde viel Wert auf eine fehlerfreie Rekonstruktion der Ohren gelegt.

Der zweite Aspekt ist die verwendete Datenbank für das Formmodell. Eine große Datenmenge mit einer großen Variabilität ist die Voraussetzung für eine genaue Anpassung des Modells an die Fotos. Die hier verwendeten Daten sind leider etwas einseitig, was die ethnische Verteilung angeht. Die Scans der Frauen waren außerdem zum großen Teil zu schlecht, um in das Formmodell einzufließen. Eine vielfältigere Datenmenge würde das Ergebnis sicher verbessern.

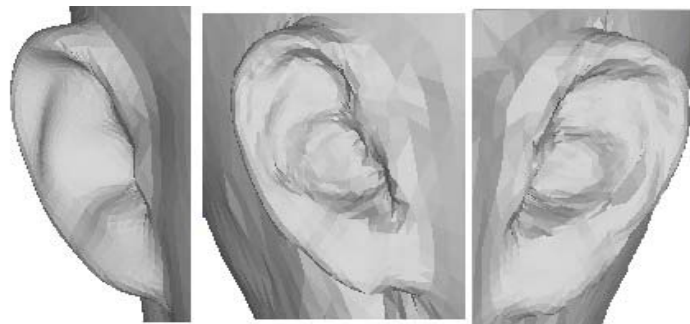
Weiterhin ist das Ergebnis der Rekonstruktion aus 2D Bildern zu vergleichen. In dieser Arbeit wurden 2 Ansätze vorgestellt. Beide verformen ein Formmodell auf der



(a) Ohren des Formmodells nach Blanz und Vetter in Breuer et. al. [BKK⁺08]



(b) Ohren des Modells von Schneider [SE09]

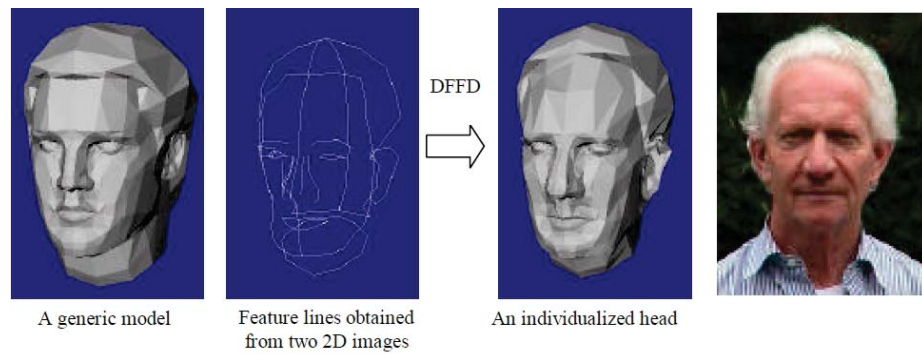


(c) Ohren des Modells aus dieser Arbeit

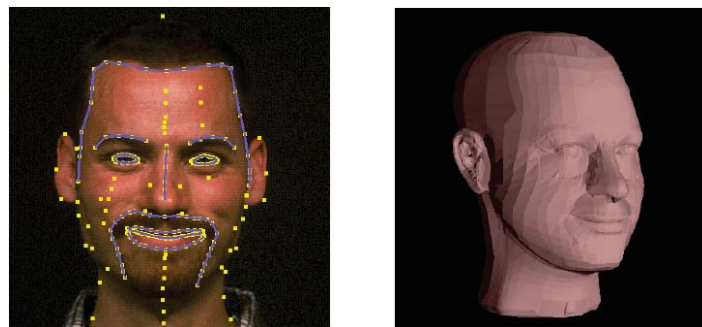
Abbildung 43: Ohren verschiedener Formmodelle

Grundlage von in Bildern markierten Landmarken. Der erste Algorithmus hat den Vorteil, dass er ausschließlich vorhandene Daten verwendet. Der zweite Algorithmus dagegen verwendet eine suggestive Z-Koordinate der Frontallandmarken. Jedoch ist der Nachteil des ersten Verfahrens die Verwendung einer reinen ICP, denn diese liefert nicht immer ein beschauliches Ergebnis. Der zweite Ansatz dagegen liefert anhand der TPS-Verformung immer ein einigermaßen zu den Bildern passendes Ergebnis. Im Folgenden sollen die Ergebnisse dieser Arbeit mit denen anderer Autoren verglichen werden.

Obwohl diverse Arbeiten auf dem Gebiet der 3D Gesichts- und Kopfrekonstruktion existieren, ist ein quantitativer Vergleich der Arbeiten kaum möglich. Jeder Autor verwendet andere Kostenfunktionen und erreicht damit andere Fehlergrößen, die für ein gutes bzw. schlechtes Ergebnis sprechen. Aus diesem Grund ist nur ein qualitativer



(a) Modifikation eines generischen Kopfes (links) mit den Merkmalslinien der 2D Bilder nach Lee [LMT98]



(b) Kombination der annotierten Landmarken aus 5 verschiedenen Bildern und das rekonstruierte Modell mit 112 korrespondierenden Merkmalspunkten, aus Pighin [PHL⁺98]

Abbildung 44: Deformation generischer Modelle

Vergleich über die Optik möglich, wobei anzunehmen ist, dass die verwendeten Bilder und Daten in den Artikeln und Arbeiten eher zu den besseren Ergebnissen zu zählen sind. Weiterhin sind reine Geometrien der Köpfe selten abgebildet.

In den Arbeiten von Lee [LMT98] und Pighin [PHL⁺98] wird eine Deformation durch Interpolation ähnlich einer Spline Deformation verwendet. Eine TPS Deformation wird auch in Ansatz 2 dieser Arbeit verwendet, um das Modell an die Frontal- und Profil-landmarken zu ziehen.

Das Ergebnis von Lee aus Abbildung 44(a) ist deutlich gröber als das von Pighin oder das hier verwendete Modell. Dementsprechend ist die Ähnlichkeit zu dem Foto gering. Das Modell von Pighin ist dagegen deutlich feiner und gibt die Emotionen des Bildes gut wieder. Das Ergebnis dieser Deformation weist jedoch große Ähnlichkeit mit dem Ausgangsmodell auf, wodurch nicht klar wird, ob ähnlich gute Ergebnisse auch für andere Personen erreicht werden können.

Im zweiten Ansatz dieser Arbeit wird bei der Frontalanpassung eine Thin Plate Spline Verformung verwendet, um das Modell an die Front- und Profillandmarkpositionen zu ziehen. Wird allein diese Variante zur 3D Rekonstruktion des Kopfes genutzt, erreicht man als Ergebnis ein glatt verformtes Objekt. Ein Beispiel für die Verformung des Mittelwertsvektors durch TPS ist in Abbildung 45 zu sehen.

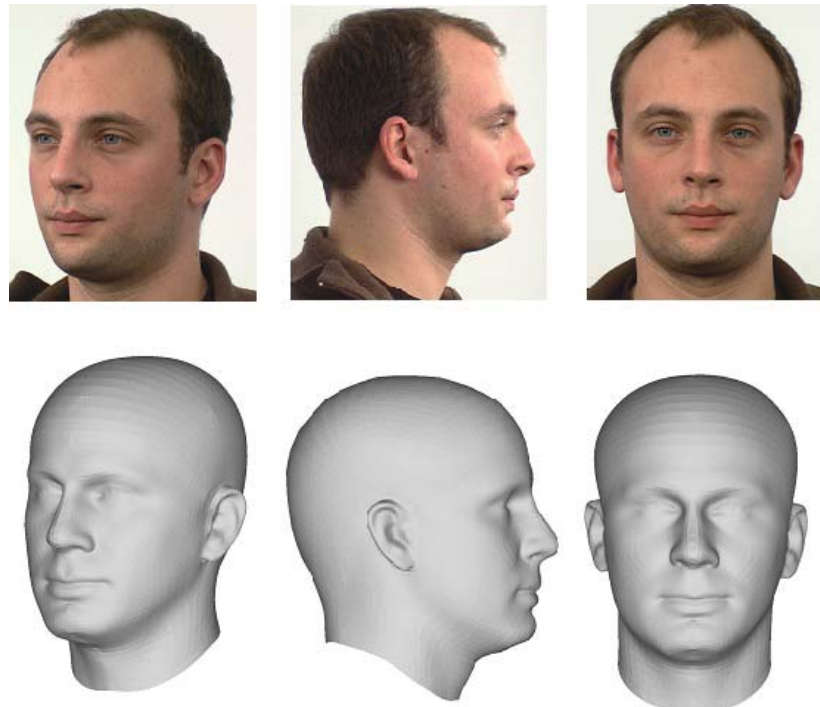


Abbildung 45: Neue Gesichtswinkel des nur mit TPS verformten Modells

Obwohl das Modell nur aus 2 orthogonalen Bildern entwickelt wurde, stimmt auch die schräge Ansicht gut mit dem Foto überein. Bestimmte Feinheiten, die erst durch das Formmodell und der Gewichtung der Eigenvektoren zustande kommen, sind noch nicht zu erkennen. Trotzdem ist das Modell durch die gesetzten Landmarken individuell verformt worden und ist ähnlich gut wie das Modell von Pighin und deutlich besser als das von Lee.

Zuletzt wird das Ergebnis der 3D Rekonstruktion aus 2D Daten von dieser Arbeit verglichen mit anderen Ergebnissen, die ebenfalls ein Formmodell als Grundlage für die Berechnungen verwenden. Wie bereits erwähnt, ist das Formmodell von Blanz und Vetter [BV99] das Standardmodell für dieses Verfahren. Daher wird es in gleicher oder leicht abgeänderter Variante auch in anderen Arbeiten verwendet [BKK⁺08] [JHY⁺05] [SE09].

Um die Ergebnisse zu vergleichen, werden aus den Arbeiten Bilder verwendet, die ohne Textur abgebildet sind. Durch die Textur, die mit Hilfe optischer Fluss Algorithmen angepasst wurde, sehen viele Modelle sehr gut aus, obwohl die Geometrie wenig Übereinstimmungen aufweist.

Die beste Geometrierekonstruktion liefern Blanz und Vetter in ihrem eigenen Artikel. Bild 1 auf Abbildung 46 zeigt das für die Rekonstruktion verwendete Ausgangsbild, Bild 2 das erstellte Geometriemodell und Bild 4 das texturierte Modell. Etwas weniger gut ist das gewählte Beispiel aus Jiang [JHY⁺05], zu sehen in Abbildung 47. Hier ist nur

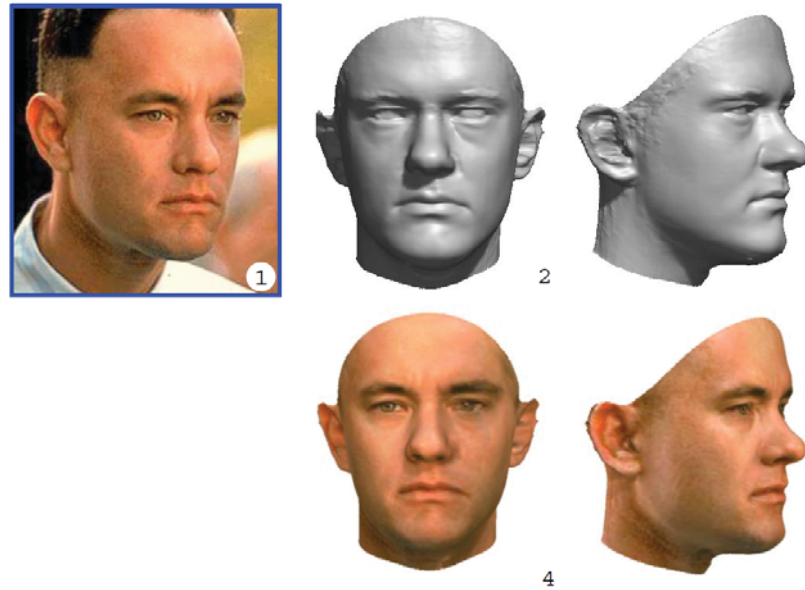


Abbildung 46: Formmodell Ergebnis von Blanz und Vetter. Die Anpassung des Formmodells an ein Foto (Bild 1) ergibt eine 3D Geometrie (Bild 2) und eine Textur für dieses Modell (Bild 4). vgl. Seite 6 [BV99]



Abbildung 47: Formmodell Ergebnis von Jiang. Links ist das 2D Bild für die Rekonstruktion mit verwendeten Landmarken zu sehen. Das rechte Bild zeigt das fertige 3D Modell für dieses Foto.(vgl. Seite 8, [JHY⁺05])

die Frontalansicht des Modells zu sehen und kein Vergleichsbild für neue Blickwinkel. Bei Breuer et. al. [BKK⁺08] wurden die Ergebnisse von Probanden zwischen “sehr gut” und “schlecht” eingeschätzt. Beispiele dafür mit entsprechender Geometrie sind in Abbildung 48 zu sehen. Es wurde das Modell mit Textur bewertet. Vergleicht man Geometrie mit Foto, fällt auf, dass zum einen die Frauenmodelle wenig weiblich sind und auch Merkmale Personen mit dunkler Hautfarbe nicht gut herausgearbeitet sind. Die Modelle sind dennoch sehr detailreich.

Ein großer Vorteil dieser Rekonstruktionen ist die Verwendung nur eines Fotos für die Erstellung der Geometrie. Denn hier ist die weitere Information nicht ein weiteres Bild sondern die Textur mit ihren Helligkeitsverläufen und Schattierungen. Mit Hilfe des optischen Flusses werden Korrespondenzen zwischen Bild und Geometrie gesucht und



Abbildung 48: Formmodell Ergebnis in Breuer et. al. Jeweils zwei Beispiele für die jeweilige Bewertung der Ergebnisse durch Probanden. Von links nach rechts: Originalbild, neuer Blickwinkel des Modells mit automatischer Merkmalsfindung, Vergleichsbild für den neuen Blickwinkel, das rekonstruierte 3D Modell mit manueller Merkmalsextraktion und das Geometriemodell der automatischen Rekonstruktion ohne Textur. vgl. Seite 8, [BKK⁺08]

für die Rekonstruktion verwendet. Breuer et. al. gehen sogar noch weiter und entwerfen eine automatische Merkmalsfindung in dem verwendeten Foto.

Es zeigt sich, dass beide entwickelten Algorithmen dieser Arbeit mit den bisherigen 3D Rekonstruktionen mithalten können. In der Geometrie weisen die Ergebnisse sogar bessere Ergebnisse auf, als oben beschriebene Arbeiten. Insbesondere bei der Verwendung des gesamten Kopfes anstelle des Gesichtes und der Individualisierung der Ohren heben sich die Algorithmen positiv hervor. Die Köpfe haben eine gute Ähnlichkeit zu den Fotos. Da der zweite Ansatz etwas bessere Ergebnisse liefert, finden sich im Anhang die Abbildungen 49 bis 52. Sie zeigen die Ergebnisse und den Vergleich mit neuen Blickwinkeln. Eine Ähnlichkeit der Geometrie zu den Bildern ist gut zu erkennen. Die Texturierung muss natürlich noch verbessert werden.

9 Ausblick

Die hier vorgestellten Methoden für die Rekonstruktion eines 3D Modells aus 2D Bildern ergeben in den meisten getesteten Datensätzen ein zufriedenstellendes Ergebnis. Trotzdem gibt es einige Schwachstellen, die in Zukunft verbessert werden können.

Dazu zählt erst einmal die Gewinnung der Landmarken aus den Bildern. Diese pasierte in dieser Arbeit manuell. Eine automatische Extraktion dieser Merkmale in den Bildern würde den Arbeitsaufwand deutlich verringern.

Weiterhin ist dieses Modell limitiert auf einen neutralen Gesichtsausdruck. Dieser könnte durch eine Ausbau der Datenbank um weitere Mimiken erweitert werden.

Der Algorithmus in dieser Form lässt ebenfalls wenig Toleranz im Hinblick auf die Kopfposition zu. Aufgrund der Silhouette kann eine Rotation des Kopfes um die Breitenachse berechnet werden, eine Rotation um Längs- oder Querachse jedoch nicht.

Ein geeigneter Texturalgorithmus für das Modell, der sowohl die Frontal- als auch die Seitenansicht berücksichtigt, kann die Darstellung des Endergebnisses noch verbessern.

Beide Algorithmen haben ihre Vor- und Nachteile. Eine Kombination beider Algorithmen könnte die Vorteile vereinigen, sodass nur die vorhandenen Daten verwendet werden und eine zusätzliche Glattheit und Genauigkeit durch die TPS-Verformung erreicht wird.

10 Fazit

In dieser Arbeit wurden zwei Algorithmen zur 3D Rekonstruktion des Kopfes aus zwei orthogonalen Bildern vorgestellt. Beide basieren auf der Anpassung eines Formmodells an manuell gesetzte Merkmalspunkte in den Fotos. Beide Algorithmen liefern individuelle, den Fotos ähnliche 3D Geometrien. Trotz relativ kleiner und einseitiger Datenbank können Männer, sowie Frauen und ethnische Besonderheiten modelliert werden. Der zweite Algorithmus ist aufgrund der TPS Modellierung etwas glatter und passt an den Landmarken besser an die exakten Positionen.

Die hier vorgestellten Algorithmen stehen in den Ergebnissen bisheriger Arbeiten in Bezug auf die reine Geometrie in nichts nach. Die Ergebnisse sind auch im Hinblick auf die Gesamtheit des Kopfes und der Detailliertheit deutlich besser.

In dieser Arbeit wurde besonderes Augenmerk auf die Geometrie gelegt. Viele Arbeiten haben sich auf die Texturierung und/oder der Veränderung der Gesichtszüge auf Grundlage eines 3D Modells spezialisiert, wobei das Objekt eher schlecht modelliert ist. Es ist denkbar, die hier entwickelten Modelle mit anderen Algorithmen zu kombinieren. Das kann eine große Verbesserung auf dem Gebiet der Gesichtserkennung und -rekonstruktion bringen.

Abschließend lässt sich sagen, dass es gelungen ist, eine qualitativ gute 3D Kopfrekonstruktion zu entwickeln, die eine gute Grundlage für 3D Filme, Spiele, zur Gesichtserkennung oder Emotionsveränderung liefert.

Literaturverzeichnis

- [BKK⁺08] BREUER, Pia ; KIM, Kwang I. ; KIENZLE, Wolf ; SCHÖLKOPF, Bernhard ; BLANZ, Volker: Automatic 3D face reconstruction from single images or video. In: *FG*, IEEE, 2008, 1-8
- [Boo89] BOOKSTEIN, F. L.: Principal Warps: Thin-Plate Splines and the Decomposition of Deformations. In: *IEEE Trans. Pattern Anal. Mach. Intell.* 11 (1989), Juni, Nr. 6, 567–585. <http://dx.doi.org/10.1109/34.24792>. – DOI 10.1109/34.24792. – ISSN 0162–8828
- [BV99] BLANZ, Volker ; VETTER, Thomas: A morphable model for the synthesis of 3D faces. In: *Proceedings of the 26th annual conference on Computer graphics and interactive techniques*. New York, NY, USA : ACM Press/Addison-Wesley Publishing Co., 1999 (SIGGRAPH '99). – ISBN 0–201–48560–5, 187–194
- [BV03] BLANZ, Volker ; VETTER, Thomas: Face Recognition Based on Fitting a 3D Morphable Model. In: *IEEE Trans. Pattern Anal. Mach. Intell.* 25 (2003), September, Nr. 9, 1063–1074. <http://dx.doi.org/10.1109/TPAMI.2003.1227983>. – DOI 10.1109/TPAMI.2003.1227983. – ISSN 0162–8828
- [DB03] DONATO, G. ; BELONGIE, S.: Approximation Methods for Thin Plate Spline Mappings and Principal Warps. In: HEYDEN, A. (Hrsg.) ; SPARR, G. (Hrsg.) ; NIELSEN, M. (Hrsg.) ; JOHANSEN, P. (Hrsg.): *Proceedings, Part III, Lecture Notes in Computer Science*. Copenhagen, Denmark, 2003 (Computer Vision – ECCV 2002: 7th European Conference on Computer Vision), S. 21–31
- [FH86] FAUGERAS, O. D. ; HEBERT, M.: The representation, recognition, and locating of 3-d objects. In: *Int. J. Rob. Res.* 5 (1986), September, Nr. 3, 27–52. <http://dx.doi.org/10.1177/027836498600500302>. – DOI 10.1177/027836498600500302. – ISSN 0278–3649
- [FOK02] FANANY, Mohamad I. ; OHNO, Masayoshi ; KUMAZAWA, Itsuo: *Face Reconstruction from Shading Using Smooth Projected Polygon Representation NN*. 2002
- [JHY⁺05] JIANG, Dalong ; HU, Yuxiao ; YAN, Shuicheng ; ZHANG, Lei ; ZHANG, Hongjiang ; GAO, Wen: Efficient 3D reconstruction for face recognition. In: *Pattern Recogn.* 38 (2005), Juni, Nr. 6, 787–798. <http://dx.doi.org/10.1016/j.patcog.2004.11.004>. – DOI 10.1016/j.patcog.2004.11.004. – ISSN 0031–3203
- [Koc96] KOCH, Reinhard: 3-D Modeling of Human Heads from Stereoscopic Image Sequences. In: JÄHNE, Bernd (Hrsg.) ; GEISLER, Peter (Hrsg.) ; HAUSSECKER, Horst (Hrsg.) ; HERING, Frank (Hrsg.): *DAGM-Symposium*, Springer, 1996 (Informatik Aktuell). – ISBN 3–540–61585–7, 169–178
- [LKT97] LEE, Won sook ; KALRA, Prem ; THALMANN, Nadia M.: Model Based Face Reconstruction For Animation. In: *In Proc. Multimedia Modeling (MMM'97)*, Singapore, Press, 1997, S. 323–338

- [LLS03] LAMECKER, Hans ; LANGE, Thomas ; SEEBASS, Martin: Erzeugung statistischer 3D-Formmodelle zur Segmentierung medizinischer Bilddaten. In: WITTENBERG, Thomas (Hrsg.) ; HASTREITER, Peter (Hrsg.) ; HOPPE, Ulrich (Hrsg.) ; HANDELS, Heinz (Hrsg.) ; HORSCH, Alexander (Hrsg.) ; MEINZER, Hans-Peter (Hrsg.): *Bildverarbeitung für die Medizin* Bd. 80, CEUR-WS.org, 2003 (CEUR Workshop Proceedings), 398-402
- [LMT98] LEE, Won-Sook ; MAGNENAT-THALMANN, Nadia: Head Modeling from Pictures and Morphing in 3D with Image Metamorphosis Based on Triangulation. In: MAGNENAT-THALMANN, Nadia (Hrsg.) ; THALMANN, Daniel (Hrsg.): *CAPTECH* Bd. 1537, Springer, 1998 (Lecture Notes in Computer Science). – ISBN 3-540-65353-8, 254-267
- [PHL⁺98] PIGHIN, Frédéric ; HECKER, Jamie ; LISCHINSKI, Dani ; SZELISKI, Richard ; SALESIN, David H.: Synthesizing Realistic Facial Expressions from Photographs. In: *Proceedings of the 25th Annual Conference on Computer Graphics and Interactive Techniques*. New York, NY, USA : ACM, 1998 (SIGGRAPH '98). – ISBN 0-89791-999-8, 75-84
- [PZ10] PATEL, Narendra ; ZAVERI, Mukesh: 3D Facial Model Construction and Expressions Synthesis using a Single Frontal Face Image. In: *International Journal of Graphics* 1 (2010), November, Nr. 1. <http://dx.doi.org/10.1109/ICCCT.2010.5640447>. – DOI 10.1109/ICCCT.2010.5640447
- [RBV02] ROMDHANI, Sami ; BLANZ, Volker ; VETTER, Thomas: Face Identification by Fitting a 3D Morphable Model Using Linear Shape and Texture Error Functions. In: *ECCV* (4), 2002, S. 3-19
- [San10] SANDER, Nick: *Erstellung und Evaluierung eines statistischen Formmodells des Kieferknochens*. GRIN Verlag, 2010 <http://books.google.de/books?id=wACIQrVfweYC>. – ISBN 9783640744404
- [SE09] SCHNEIDER, David ; EISERT, Peter: Fast Nonrigid Mesh Registration with a Data-Driven Deformation Prior. (2009), September, Nr. 9, S. 304-311
- [SE10] SCHNEIDER, David C. ; EISERT, Peter: Algorithms For Automatic And Robust Registration Of 3D Head Scans. In: *JVRB* 7 (2010). <http://dblp.uni-trier.de/db/journals/jvrb/jvrb7.html#SchneiderE10>
- [SHF09] SAJÓ, L. ; HOFFMANN, M. ; FAZEKAS, A.: A 3D head model from stereo images by a self-organizing neural network. In: *Journal for Geometry and Graphics* 13 (2009), Nr. 2, S. 209-220. – ISSN 1433-8157
- [SKHE11] SCHNEIDER, David C. ; KETTERN, Markus ; HILSMANN, Anna ; EISERT, Peter: A Global Optimization Approach to High-detail Reconstruction of the Head. In: EISERT, Peter (Hrsg.) ; HORNEGGER, Joachim (Hrsg.) ; POLTHIER, Konrad (Hrsg.): *VMV*, Eurographics Association, 2011. – ISBN 978-3-905673-85-2, 9-15
- [TP91] TURK, Matthew ; PENTLAND, Alex: Eigenfaces for recognition. In: *J. Cognitive Neuroscience* 3 (1991), Januar, Nr. 1, 71-86. <http://dx.doi.org/10.1162/jocn.1991.3.1.71>. – DOI 10.1162/jocn.1991.3.1.71. – ISSN 0898-929X

- [WD08] WIDANAGAMAACHCHI, W.N. ; DHARMARATNE, A.T.: 3D Face Reconstruction from 2D Images. In: *Digital Image Computing: Techniques and Applications (DICTA)*, 2008, 2008, S. 365–371
- [WJC12] WEON, SunHee ; JOO, SungIl ; CHOI, HyungIl: Individualized 3D Face Model Reconstruction using Two Orthogonal Face Images. In: AO, S. I. (Hrsg.) ; DOUGLAS, Craig (Hrsg.) ; GRUNDFEST, W. S. (Hrsg.) ; BURGSTONE, Jon (Hrsg.) ; International Association of Engineers (Veranst.): *Proceedings of the World Congress on Engineering and Computer Science 2011 Vol I, WCECS '12, October 24 - 26, 2012, San Francisco, USA* International Association of Engineers, Newswood Limited, 2012 (Lecture Notes in Engineering and Computer Science). – ISBN 978-988-19251-6-9, S. 680–684
- [Zha94] ZHANG, Zhengyou: Iterative point matching for registration of free-form curves and surfaces. In: *Int. J. Comput. Vision* 13 (1994), Oktober, Nr. 2, 119–152. <http://dx.doi.org/10.1007/BF01427149>. – DOI 10.1007/BF01427149. – ISSN 0920-5691

Anhang

Die hier aufgeführten Abbildungen zeigen einige Ergebnisse der Rekonstruktionen des zweiten Ansatzes mit Thin Plate Spline/ICP Verformung mit und ohne Texturierung. Die neuen Blickwinkel der Personen im Vergleich zu den originalen Fotos zeigen Übereinstimmung des 3D Modells auch aus anderen Richtungen.



Abbildung 49: 3D Modell mit anderen Vergleichsblickwinkeln (1)



Abbildung 50: 3D Modell mit anderen Vergleichsblickwinkeln (2)

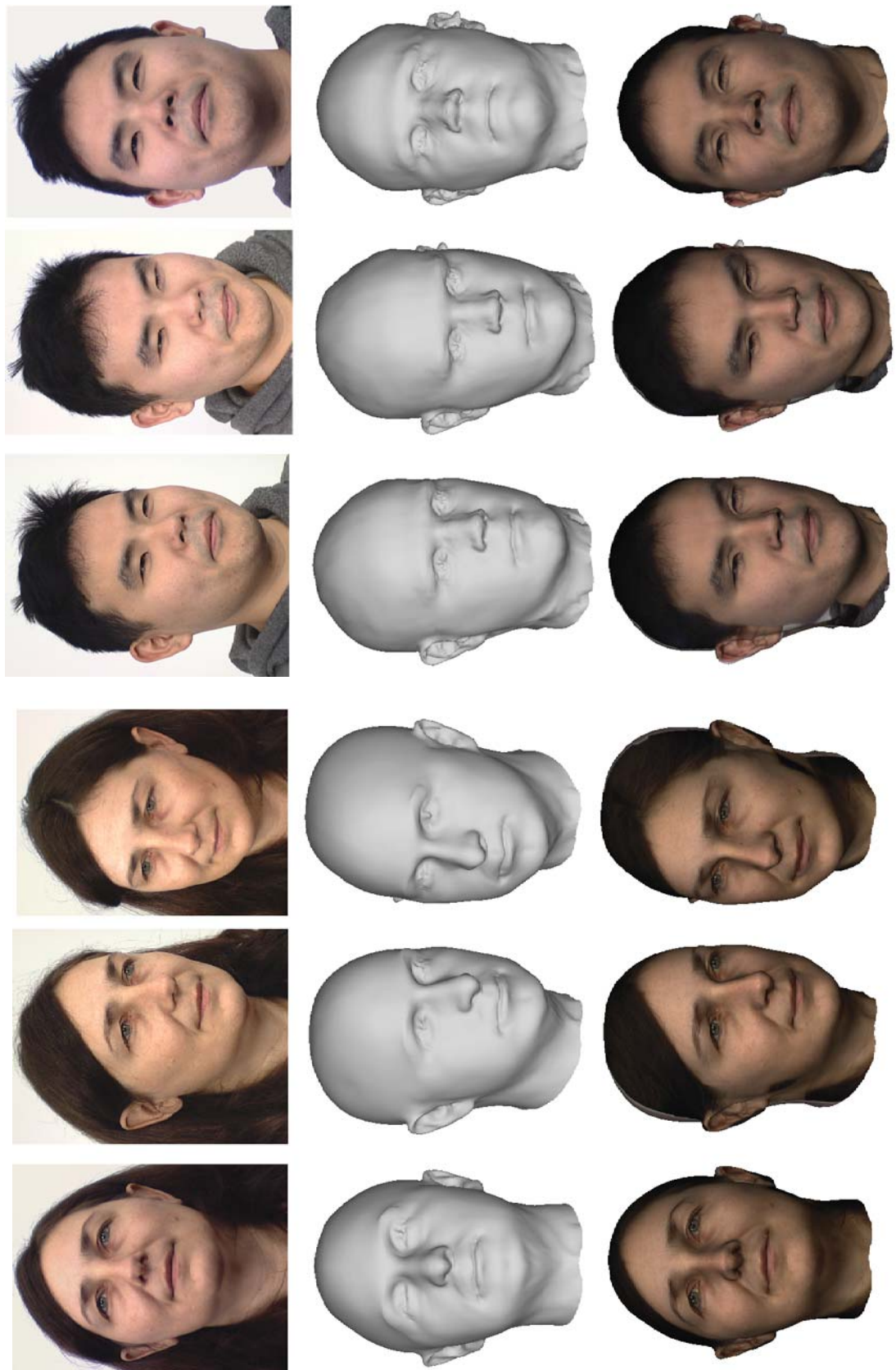


Abbildung 51: 3D Modell mit anderen Vergleichsblickwinkeln (3)



Abbildung 52: 3D Modell mit anderen Vergleichsblickwinkeln (4)