

Aus dem Institut für
Mathematische Methoden der Bildverarbeitung
der Universität zu Lübeck
Direktor: Prof. Dr. Bernd Fischer

Restringierte Medizinische Bildregistrierung

Inauguraldissertation
zur
Erlangung der Doktorwürde
der Universität zu Lübeck

Aus der Sektion Informatik/Technik

vorgelegt von
Björn Beuthien
aus Oldenburg i.H.
Lübeck 2012

-
1. Berichterstatter/Berichterstatterin: Prof. Dr. Jan Modersitzki
 2. Berichterstatter/Berichterstatterin: Prof. Dr. Heinz Handels

Tag der mündlichen Prüfung: 12. Februar 2014

Zum Druck genehmigt. Lübeck, den 14. Februar 2014

Eidesstattliche Erklärung

Hiermit erkläre ich, dass ich diese Dissertation selbstständig verfasst und keine anderen als die angegebenen Hilfsmittel genutzt habe. Ich versichere weiterhin, dass ich diese Dissertation nur in diesem und keinem anderen Promotionsverfahren eingereicht habe und dass diesem Promotionsverfahren keine endgültig gescheiterten Promotionsverfahren vorausgegangen sind.

Lübeck,

Danksagung

Viele Personen haben mich auf dem Weg zur Fertigstellung dieser Arbeit unterstützt. Hier anfangen möchte ich bei Bernd Fischer, meinem Betreuer. Vielen Dank für die nicht nur fachlichen Fachgespräche. Die gemeinsame Numerik-Zeit habe ich sehr genossen.

Jan Modersitzki ist ein weiterer dieser Kandidaten. Zusammen haben wir uns gut durch einige Lehrveranstaltungen geschlagen und ich habe sehr viel aus dieser Zeit mitgenommen. Auch über spontane Besuche in meinem Büro habe ich mich sehr gefreut.

Eine Hilfe, die ich zum Anfang nicht eingerechnet hatte, kam aus Amerika. Charles Guttman möchte ich für seine Einblicke in die medizinische Seite danken. Für die Hilfe bezüglich medizinischer Belange im Bereich der Leber möchte ich mich ebenfalls bei Thilo Schumacher und Christoph Logge bedanken.

In Lübeck ist natürlich noch der gesamten Gruppe zu danken. Ich habe immer wieder ein offenes Ohr gefunden, z. B. bei den Gesprächen über die Längenerhaltung mit Stefan Heldmann. Und mein Zimmerkollege Nils Papenberg. Danke für die super Zeit!

Nach einer langen Zeit des Aufschreibens wird man auch ein wenig blind. Deshalb möchte ich mich bei Wolfgang Erb, Judith Berger, Andreas Mang und meinem Vater Rüdiger Beuthien für das Korrekturlesen bedanken.

Wo ich schon bei meinem Vater bin, nehme ich auch gleich meine Mutter Marion Beuthien hinzu und bedanke mich für ihre Unterstützung in jeglicher Art.

Danke an alle!

Zusammenfassung

Nicht-parametrische Bildregistrierung ist eines der herausforderndsten Gebiete in der medizinischen Bildverarbeitung. Das Problem kann formuliert werden als

“Finde eine plausible Deformation φ , so dass korrespondierende Punkte im Referenzbild \mathcal{R} und im Templatebild \mathcal{T} möglichst gut aufeinander abgebildet werden.“

Typischerweise versucht man zu diesem Zweck ein Funktional, bestehend aus einem Distanzmaß und einem Regularisierer, zu minimieren und eine plausible Deformation zu erlangen. Hierzu bestraft der Regularisierungsterm z. B. nicht-glatte Deformationen. Das reicht in manchen Fällen allerdings nicht aus. Wir betrachten dazu in dieser Arbeit zwei Beispiele.

Die Tumorresektion ist eine der häufigsten Aufgaben in der Leberchirurgie. Dabei ist es zum einen von besonderer Wichtigkeit, den Tumor mit einem Sicherheitsabstand komplett zu entfernen. Zum anderen soll aber auch so viel heiles Gewebe wie möglich erhalten bleiben. Dafür wird präoperativ eine CT-Aufnahme gemacht und aus den gewonnenen Daten ein genauer Resektionsplan erstellt. Für den Chirurgen ist es nach der Operation wichtig das Ergebnis mit der Planung vergleichen zu können. Offensichtlich sind die prä- und postoperativen Daten nicht direkt vergleichbar, eine aussagekräftige Registrierung wird gebraucht. In der Literatur finden wir einen rigiden und einen landmarkenbasierten Ansatz, um das Problem zu bewältigen. Während die rigide Registrierung die nichtlinearen Deformationen nicht einfangen kann, kann der nichtlineare landmarkenbasierte Ansatz eine ungewollte Überregistrierung bzw. ein Nachwachsen der Leber nicht komplett verhindern.

Der zweite Anwendungsfall behandelt die Registrierung von MRT-Aufnahmen zur Kontrolle des Krankheitsverlaufs bei Multipler Sklerose. Um die Entwicklung von Läsionen zu beobachten, ist die Subtraktionsbildgebung von T_2 -gewichteten MRT-Aufnahmen ein wertvolles Hilfsmittel. Die kritische Komponente dieser Prozedur ist eine genaue Ausrichtung der Diagnosebilder, die zu unterschiedlichen Zeitpunkten aufgenommen wurden. Da die Größenänderung der Läsionen von besonderem Interesse ist, schlägt eine typische nicht-lineare Registrierung fehl. Bei diesem Ansatz würden auch die Läsionen direkt aufeinander abgebildet werden.

In beiden oben genannten Fällen betrachten wir Aufnahmen desselben Patienten, jedoch mit unterschiedlicher pathologischer Anatomie. Eine Möglichkeit die beiden Probleme unter Kontrolle zu bekommen ist es, zusätzliches Wissen in das mathematische Modell einfließen zu lassen. Dies wollen wir in dieser Arbeit tun. Hierzu verwenden wir den sogenannten diskretisieren-dann-optimieren Ansatz und binden

das Zusatzwissen über Nebenbedingungen in das Modell ein. Nach einer Einführung in die Grundlagen der dafür nötigen Optimierungstechniken und der Registrierung wenden wir uns den beiden Anwendungsfällen zu. Im Zuge der Registrierung prä- und postoperativer Daten entwickeln wir zwei unterschiedliche Ansätze und vergleichen diese. Der erste Ansatz verhindert ein Nachwachsen der Leber, indem eine punktweise Volumenerhaltung während der Registrierung gefordert wird. Der zweite Ansatz fordert eine Längenerhaltung der Lebergefäße. Während volumenerhaltende Nebenbedingungen schon in anderen Bereichen der intensitätsbasierten Registrierung genutzt worden sind, ist die Längenerhaltung neu in diesem Bereich. Bei der Längenerhaltung wird zudem eine zusätzliche Nebenbedingung entwickelt. Diese liefert punktuell Informationen über die Inverse des Deformationsfeldes. Diese Nebenbedingung wird, zusammen mit der volumenerhaltenden Nebenbedingung, im zweiten Anwendungsfall für eine regional eingeschränkte Volumenerhaltung wiederverwendet. In beiden Anwendungen stellt sich heraus, dass durch das Einbringen von Nebenbedingungen bessere Ergebnisse erzielt werden können als mit einem Standardansatz.

Inhaltsverzeichnis

1. Einleitung	1
1. Grundlagen	5
2. Optimierung	7
2.1. Unrestringierte Optimierung	7
2.1.1. Schrittweitenbestimmung	8
2.1.2. Suchrichtung	11
2.1.3. Das BFGS-Verfahren	12
2.1.4. Das L-BFGS-Verfahren	14
2.1.5. Das Gauss-Newton-Verfahren	15
2.1.6. Abbruchkriterien	17
2.1.7. Vergleich der Verfahren	17
2.2. Restringierte Optimierung	21
2.2.1. Grundlagen	22
2.2.2. Algorithmen	23
2.2.3. Augmented Lagrangian - Ein Beispiel	29
2.3. Zusammenfassung	32
3. Bildregistrierung	35
3.1. Bilder und Gitter	37
3.2. Interpolation	42
3.3. Numerische Integration und Differentiation	48
3.3.1. Numerische Integration	48
3.3.2. Numerische Differentiation	49
3.4. Transformationen	50
3.4.1. Eigenschaften	53
3.4.2. Diskretisierung	54
3.5. Distanzmaße	58
3.6. Plausibilität von Transformationen	66
3.6.1. Regularisierer	67
3.6.2. Zusatzwissen	72

3.7. Allgemeiner Registrierungsrahmen	75
3.8. Multiskalen Ansätze in der Bildregistrierung	78
3.8.1. Grundlagen	78
3.8.2. Ein Multilevel Ansatz	80
3.9. Zusammenfassung	84
II. Anwendung	85
4. Registrierung Prä- und Postoperativer Leberdaten	87
4.1. Grundlagen der Leberresektion	87
4.1.1. Leberanatomie	88
4.1.2. Durchführung der Leberresektion	89
4.2. Motivation	90
4.3. Stand der Technik	91
4.4. Datenlage	92
4.5. Vorregistrierung	97
4.5.1. Schwerpunktregistrierung	97
4.5.2. VCI-Vorregistrierung	99
4.5.3. Rigide Landmarkenregistrierung	100
4.5.4. Ergebnisse der Vorregistrierung	102
4.6. Volumenerhaltende Registrierung	104
4.6.1. Kontinuierliche Formulierung	105
4.6.2. Vorverarbeitung	107
4.6.3. Distanzmaß	107
4.6.4. Regularisierer	108
4.6.5. Diskretisierung	108
4.6.6. Ein Multilevel Augmented Lagrangian	110
4.6.7. Anwendung und Vergleich von Volumenstraftermen	112
4.6.8. Vergleich der Optimierungsverfahren	114
4.6.9. Fazit und Ausblick	119
4.7. Längenerhaltende Registrierung	120
4.7.1. Stand der Technik	120
4.7.2. Modellierung der Gefäße	121
4.7.3. Längenerhaltungsnebenbedingungen	125
4.7.4. Inversitätsforderung	126
4.7.5. Vorverarbeitung	127
4.7.6. Distanzmaß	127
4.7.7. Regularisierung des Funktionals	127
4.7.8. Kontinuierliche Formulierung der Längenerhaltenden Registrierung	128

4.7.9. Diskretisierung	128
4.7.10. Vergleich der Unrestringierten Optimierungsverfahren	133
4.7.11. Fazit und Ausblick	137
4.8. Ausblick der Prä/Post-Registrierung	137
5. Subtraktionsbildgebung bei Multipler Sklerose	141
5.1. Grundlagen der Multiplen Sklerose	143
5.2. Stand der Technik	143
5.3. Läsions-Volumenerhaltende Registrierung	146
5.4. Anwendung und Vergleich von Distanzmaßen	155
5.5. Vergleich der Optimierungsverfahren	156
5.6. Ausblick	158
6. Diskussion und Ausblick	161

1. Einleitung

Die Bildgebung spielt in der Medizin eine wichtige Rolle. Durch sie wird es möglich von außen in den Patienten hineinzuschauen. Dies ermöglicht unter anderem bessere Diagnosen, die Beobachtung des Verlaufs einer Krankheit oder auch die Überprüfung eines Operationsverlaufs. Die Bilddaten sind allerdings durch die Lage des Patienten, Atmung, Herzschlag, unterschiedliche Aufnahmeverfahren (Modalitäten) und viele weitere mögliche Einflüsse nicht immer direkt vergleichbar. An diesem Punkt kommt die Bildregistrierung ins Spiel, deren Aufgabe die Ausrichtung korrespondierender Strukturen ist. Viele verschiedene Methoden wurden zur Lösung dieses Problems vorgeschlagen, siehe z.B. [97]. Ein Großteil dieser basiert auf einer Minimierung eines Funktionals, das zum einen darauf achtet, dass die Bilder möglichst ähnlich werden. Zum anderen wird dabei darauf geachtet, dass nur plausible Transformationen zur Ausrichtung der Bilder genutzt werden. Diese einfache Modellannahme hat jedoch ihre Grenzen. Betrachten wir beispielsweise die Aufnahmen einer Leber vor und nach einer Resektion. In diesem Fall fehlen zu einem Teil der Leber in der präoperativen Aufnahme, deren Gegenpart in der postoperativen Leber. Führen wir nun eine (Standard-)Registrierung auf diesen Daten aus, wird das Verfahren versuchen beide Bilder möglichst ähnlich zu machen. Dies führt jedoch typischerweise zu einem Nachwachsen der postoperativen oder zu einem Schrumpfen der präoperativen Leber im betrachteten Bild. Bei beiden Fällen werden nun jedoch nicht mehr korrespondierende Strukturen aufeinander abgebildet. Dieses Beispiel ist in Abbildung 1.1 zu betrachten.

Ähnliche Probleme treten in vielen weiteren Anwendungen auf. Ein Beispiel ist hier die Beobachtung eines Krankheitsverlaufs bei einem Patienten (siehe z.B. [84, 69]). Aber auch gerade bei der Registrierung zwischen unterschiedlichen Patienten tritt das Problem fehlender Korrespondenzen auf. Registrierungsprobleme dieser Art werden beispielsweise in [9] behandelt. Ein Ansatz dieses Problem unter Kontrolle zu bringen, ist die Entwicklung eines feineren Modells durch Einbringen von Vorwissen. Im Beispiel der Leberdaten können wir z. B. explizit fordern, dass die postoperative Leber bei der Registrierung nicht nachwachsen darf. An diesem Punkt setzt die vorliegende Dissertation an. Das Vorwissen soll anhand von Nebenbedingungen in das bestehende Modell eingebunden werden. Hierzu wird ein diskretisieren-dann-optimieren Ansatz verwendet, bei dem das kontinuierliche Problem zuerst diskre-

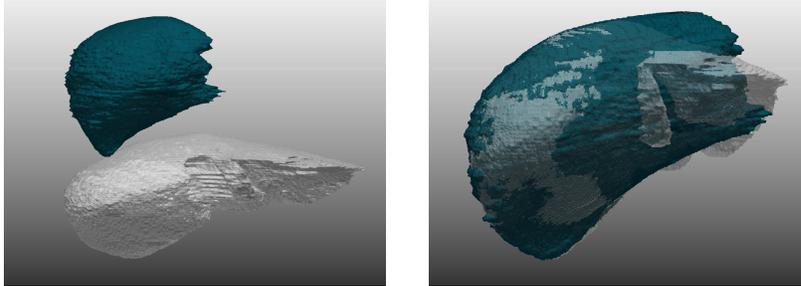


Abbildung 1.1.: Problem bei einer standardmäßigen nichtlinearen Registrierung einer post- auf eine präoperative Leber. Die initiale Situation ist links zu sehen, rechts das Ergebnis. Das Verfahren versucht beide Daten möglichst ähnlich zu machen und unterstützt damit ein Nachwachsen der postoperativen Leber. Dadurch werden allerdings nicht die korrespondierenden Strukturen aufeinander ausgerichtet. Für eine Operationskontrolle ist dieses Ergebnis beispielsweise unbrauchbar.

tisiert und dann optimiert wird. Ein großer Vorteil dabei besteht darin, dass die numerische Optimierung einen sehr guten Rahmen zur Behandlung von Problemen mit Nebenbedingungen bietet. Diesen möchten wir in dieser Arbeit vorstellen, sowie zwei spezielle Anwendungen näher betrachten, bei denen eine feinere Modellierung mit Einbringung von Vorwissen deutlich bessere und plausible Ergebnisse mit sich bringt. Ziel dieser Arbeit ist einerseits eine Vorstellung der Möglichkeiten, die eine feinere Modellierung des Problems mit sich bringen. Andererseits soll auch deutlich werden, dass dieser Schritt nicht zu viel Mehraufwand bedeutet und am Ende deutlich bessere Ergebnisse aufweist.

Die weitere Arbeit ist wie folgt aufgeteilt.

Kapitel 2: Optimierung Die Optimierung ist eines der Kernelemente der Registrierung in einem diskretisieren-dann-optimieren Rahmen. In diesem Kapitel werden wir nach einem Einblick in die unrestringierte Optimierung und einige für uns interessante Methoden aus diesem Gebiet auf die restringierte Optimierung kommen. Diese ist essentiell für das Einbringen von zusätzlichem Vorwissen in dem benutzten Rahmen. Hierbei liegt der Schwerpunkt auf dem Augmented Lagrangian Ansatz.

Kapitel 3: Registrierung Das Registrierungskapitel stellt eine Zusammenfassung der Registrierung im diskretisieren-dann-optimieren Rahmen. Angefangen bei den Bildern, geht es über Interpolation und Diskretisierungswerkzeuge zu Distanzmaßen,

Regularisierern und das Einbringen von Vorwissen. Zum Abschluss wird der generelle Rahmen einer Registrierung vorgestellt. Der Abschluss des Kapitels ist einem Multiskalen/-level Ansatz in der Registrierung gewidmet zur Stärkung der Robustheit und Performance des Verfahrens.

Kapitel 4: Registrierung Prä- und Postoperativer Leberdaten Mit Kapitel 4 beginnt der zweite Teil der Arbeit. Nach Erklärung der Grundlagen beschäftigen wir uns mit einer Anwendung aus der Leberchirurgie. Hierbei sollen Leberdaten, aufgenommen mit einem CT-Scan vor und nach einer Resektion, zur Analyse der Operation miteinander verglichen werden. Das oben erwähnte Beispiel, dargestellt in Abbildung 1.1, kommt aus dieser Anwendung. Wir stellen hier zwei verschiedene Möglichkeiten vor ein Nachwachsen der Leber zu verhindern. Zum einen wird eine Volumenerhaltung auf dem gesamten Gebiet gefordert, zum anderen eine Längenerhaltung der Gefäße. Die zweite Nebenbedingung ist im Bereich der intensitätsbasierten Registrierung neu und wird daher ausführlicher behandelt. Für beide Modelle werden unrestringierte Optimierungsverfahren innerhalb des Augmented Lagrangian Ansatzes getestet und gegeneinander verglichen.

Kapitel 5: Subtraktionsbildgebung für MS Die zweite Anwendung kommt aus dem Bereich der Beobachtung von Krankheitsverläufen bzw. Therapieverläufen bei Multipler Sklerose. Zum Vergleich zweier Aufnahmen eines Patienten zu unterschiedlichen Zeitpunkten soll ein Subtraktionsbild erstellt werden. In diesem Bild sollen bis auf Läsionsänderungen keine weiteren Informationen vorhanden sein. Obwohl sich das Gehirn beinahe rigide verhält, stellt sich heraus, dass durch unzureichende Ausrichtung der grauen und weißen Substanz ein Rauschen in diesem Subtraktionsbild entsteht. Damit eine nichtlineare Registrierung nicht die Läsionsvolumina und damit die Aussagekraft der Subtraktionsbilder zerstört, fügen wir eine Volumenerhaltungsforderung im Bereich der Läsionen des Referenz- und Templatebildes ein. Im Vergleich zur Volumenerhaltung im kompletten Gebiet müssen hier zusätzlich die Änderungen der Läsionsbereiche mitverfolgt werden.

Kapitel 6: Diskussion und Ausblick Zum Abschluss der Dissertation resümieren wir die vorgestellten Methoden und Ergebnisse und geben einen Ausblick auf weitere Arbeitsschritte und Anwendungsbereiche.

Teil I.
Grundlagen

2. Optimierung

Die nachfolgenden Registrierungsprobleme werden in einem diskretisieren-dann-optimieren Rahmen angegangen. Wie der Name bereits sagt, wird das kontinuierliche Registrierungsfunctional dabei zuerst diskretisiert und dann optimiert. Dieses Kapitel soll die Grundlagen des zweiten Schritts näher beleuchten. Das Problem kann in der allgemeinen Form

$$\min_{\mathbf{x}} f(\mathbf{x}) \quad \text{u. d. NB} \quad c_i(\mathbf{x}) = 0, \quad i = 1, \dots, m, \quad (2.1)$$

beschrieben werden, wobei $f : \mathbb{R}^n \rightarrow \mathbb{R}$ ZIELFUNKTION und $c_i : \mathbb{R}^n \rightarrow \mathbb{R}$, $i = 1, \dots, m$ NEBENBEDINGUNGEN genannt werden. Im Zuge dieser Arbeit werden nur Gleichheitsnebenbedingungen betrachtet. Im Folgenden werden wir die generellen Ideen zur unrestringierten und restringierten Optimierung wiederholen und konkrete Algorithmen erklären, um solche Probleme zu lösen. Dabei sei angemerkt, dass die Algorithmen auf die Bestimmung lokaler Minima abzielen. Aussagen über das Erreichen globaler Minima sind dagegen schwierig und werden hier nicht weiter verfolgt. Der Hauptfokus des Kapitels liegt dabei auf der restringierten Optimierung, die wir später nutzen werden, um Zusatzinformationen mit in das Registrierungsproblem einzuarbeiten. Die Ausführungen beziehen sich dabei hauptsächlich auf [72, 5].

2.1. Unrestringierte Optimierung

Für den Anfang lassen wir die Nebenbedingungen c_i , $i = 1, \dots, m$ weg und betrachten das unrestringierte Optimierungsproblem

$$\min_{\mathbf{x}} f(\mathbf{x}), \quad (2.2)$$

für hinreichend glatte Funktionen $f : \mathbb{R}^n \rightarrow \mathbb{R}$. Abstiegsverfahren sind eine typische Wahl, um eine Lösung des Problems (2.2) zu finden. Algorithmus 2.1 zeigt deren generellen Aufbau.

Startend mit einer initialen Schätzung der Lösung $\mathbf{x}^{(0)}$, generiert der Algorithmus eine Folge von Iterierten $\{\mathbf{x}^{(k)}\}$, die zu einem Minimierer \mathbf{x}^* konvergieren soll. Um

Quelltext 2.1: Modell-Algorithmus für Abstiegsverfahren

```
1 % Zielfunktion  $f: \mathbb{R}^n \rightarrow \mathbb{R}$ 
2 % Startschätzung  $x^{(0)} \in \mathbb{R}^n$ 
3 function  $\mathbf{x}^{\text{ende}} = \text{Abstiegsverfahren}(f, x^{(0)})$ 
4  $k = 0$ 
5 while (Abbruchbedingungen nicht erfüllt)
6     Bestimme Suchrichtung  $\mathbf{s}^{(k)}$ 
7     Bestimme Schrittweite  $t^{(k)}$ 
8      $\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} + t^{(k)}\mathbf{s}^{(k)}$ 
9      $k = k + 1$ 
10 end
11 end
```

dabei Konvergenz zu einem lokalen Minimierer abzusichern, müssen die SUCHRICHTUNG $\mathbf{s}^{(k)}$ und die SCHRITTWEITE $t^{(k)}$ entsprechend gewählt werden. Genügen diese den in den folgenden Abschnitten aufgeführten Bedingungen ,

- bricht der Algorithmus bei $\nabla f(x^{(k)}) = 0$ nach endlich vielen Schritten ab,
- konvergiert die Folge $\nabla f(x^{(k)})$ gegen Null oder
- ist f nach unten unbeschränkt.

Besonders für die letzten beiden Fälle ist es wichtig dem Algorithmus mitzuteilen, wann er abbrechen soll. Dazu dienen die Abbruchkriterien, die ebenfalls noch genauer beschrieben werden. Wir starten hier mit den Bedingungen an die Schrittweite.

2.1.1. Schrittweitenbestimmung

Nehmen wir vorerst an, dass eine Suchrichtung $\mathbf{s} \in \mathbb{R}^n \setminus \{\mathbf{0}\}$ gegeben ist. Auf einem Halbstrahl entlang dieser Suchrichtung, suchen wir einen Punkt, der unsere Zielfunktion möglichst gut minimiert. Wir betrachten also die eindimensionale Funktion

$$\phi(t) = f(\mathbf{x}^{(k)} + t\mathbf{s}^{(k)})$$

in der Variablen t , unserer Schrittweite. Eine optimale Schrittweite t zu bestimmen wäre wünschenswert, ist allerdings eine schwierige und teure Aufgabe. Daher beschäftigen wir uns mit der Bestimmung einer EFFIZIENTEN SCHRITTWEITE, die

ein gewisses Maß an Abstieg sicherstellt. Für diese Schrittweite $t^{(k)}$ soll

$$f(\mathbf{x}^{(k+1)}) = \phi(t^{(k)}) \leq \delta \left(\frac{\nabla f(\mathbf{x}^{(k)}) \mathbf{s}^{(k)}}{\|\mathbf{s}^{(k)}\|_2} \right)^2, \quad (2.3)$$

$\delta > 0$, gelten. Mit dieser Bedingung können Konvergenzaussagen zu dem Abstiegsverfahren (Quelltext 2.1), bei sinnvoll gewählter Suchrichtung $\mathbf{s}^{(k)}$, gezeigt werden. Da in der Registrierung die Auswertungen der Zielfunktion teuer sind, ist eine gute Balance zwischen Reduktion der Zielfunktion und der Anzahl der Funktionsauswertungen von besonderem Interesse. Wir stellen zwei Bedingungen vor, die dieses Kriterium erfüllen.

Die ARMIJO BEDINGUNG

$$\phi(t^{(k)}) \leq f(\mathbf{x}^{(k)}) + c_1 t^{(k)} \nabla f(\mathbf{x}^{(k)})^T \mathbf{s}^{(k)}, \quad c_1 \in (0, 1) \quad (C1)$$

achtet darauf, dass die Reduktion der Zielfunktion unterhalb einer Geraden liegt, die von der Steigung im Punkt der aktuellen Iterierten abhängt. Der Parameter c_1 gewichtet die Steigung der Geraden als Balance zwischen dem Wert der Richtungsableitung und der Konstanten mit dem Wert an der aktuellen Iterierten. Eine typische Wahl ist $c_1 = 10^{-4}$. Die Schrittweitenbestimmung nach dieser Bedingung ist allerdings nicht effizient. Eine Konvergenzaussage kann aber ebenfalls gezeigt werden, siehe [94]. Das Problem dieser Bedingung ist, dass zu kleine Schrittweiten ausgewählt werden können. Die WOLFE BEDINGUNGEN fügen

$$\phi'(t^{(k)}) \geq c_2 \phi'(0), \quad c_2 \in (c_1, 1), \quad (C2)$$

als weitere Bedingung hinzu. Durch die Wahl von $c_2 < 1$ werden kleine Schrittweiten durch die Bedingung (C2) vermieden. Wichtig ist zudem, dass durch die Bedingung lokale Minima als zulässige Wahl erhalten bleiben. Eine typische Wahl in der Praxis ist $c_2 = 0.9$. Für Punkte $t^{(k)}$ mit einer zu großen positiven Steigung, und damit typischerweise weit von einem stationären Punkt entfernt liegend, werden häufig auch die STARKEN WOLFE BEDINGUNGEN

$$|\phi'(t_k)| \leq -c_2 \phi'(0), \quad c_2 \in (c_1, 1), \quad (C3)$$

genutzt. Die (starken) Wolfe Bedingungen sind effiziente Schrittweiten (2.3) [32]. Abbildung 2.1 visualisiert die Bedingungen mit den verschiedenen resultierenden zulässigen Bereichen. Die Wahl der Bedingungen und des damit verbundenen Algorithmus zur Schrittweitenbestimmung hängt stark von der jeweiligen Anwendung ab. Eine Armijo-Schrittweitenbestimmung, basierend auf einem Backtracking-Algorithmus, kommt normalerweise in einer Iteration mit wenig Funktionsauswertungen aus. Beim Backtracking-Algorithmus starten wir mit einem relativ großen

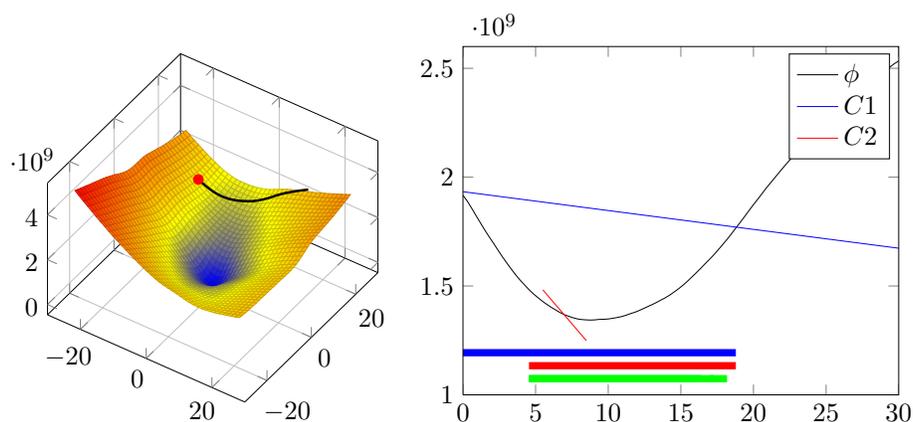


Abbildung 2.1.: Visualisierung der Schrittweiten-Bedingungen. Links ist die Zielfunktion $f : \mathbb{R}^2 \rightarrow \mathbb{R}$ zu sehen. Die aktuelle Iterierte $\mathbf{x}^{(k)}$ ist durch den roten Punkt markiert. Von hier startet der Halbstrahl (schwarz) definiert durch die aktuelle Suchrichtung $\mathbf{s}^{(k)}$. Die rechte Seite zeigt die Funktion ϕ sowie die Armijo (C1) (dünne blaue Gerade) und die linke Seite der Bedingung (C2) an die Steigung für ein spezielles t (dünnes rotes Segment). Die resultierenden zulässigen Bereiche für die Armijo (dick, blau), die Wolfe (dick rot), als auch die starken Wolfe (dick, grün) sind markiert. Man kann sehen, dass die starken Wolfe Bedingungen die restriktivsten sind. Die Parameter der Bedingung sind dabei $c_1 = 0.1$ und $c_2 = 0.9$.

Wert t und verkleinern dann diesen Wert nach jedem missglückten Versuch um einen festen Faktor bis eine den Bedingungen entsprechende Schrittweite gefunden ist. Durch die nicht effiziente Schrittweite müssen eventuell viele Iterationen durchgeführt werden. Von einem auf den (starken) Wolfe Bedingungen beruhenden Algorithmus erhoffen wir uns, bei höherem Aufwand in einer Iteration, wenige Iterationsschritte zur Konvergenz gegen einen (lokalen) Minimierer. Für genaue Ausführungen zu den Konvergenzsätzen und den jeweiligen Algorithmen möchten wir auf [72, 94] verweisen. Nun gilt es eine geeignete Suchrichtung \mathbf{s} zu wählen.

2.1.2. Suchrichtung

Bei der Betrachtung von Minimierungsproblemen erscheint es sinnvoll eine Richtung zu betrachten, in der die Zielfunktion abnimmt. Wir betrachten Abstiegsrichtungen $\mathbf{s} \in \mathbb{R}^n$, also Richtungen für deren Richtungsableitung $\nabla f^T \mathbf{s} < 0$ gilt. Wir stellen die Forderung

$$-\frac{\nabla f^T \mathbf{s}}{\|\nabla f\|_2 \|\mathbf{s}\|_2} \geq \delta > 0 \quad (2.4)$$

an unsere Suchrichtung. Die sogenannte WINKELBEDINGUNG fordert, dass der Winkel zwischen der Suchrichtung und dem negativen Gradienten $-\nabla f$ kleiner als 90° sein soll. Im Folgenden betrachten wir Suchrichtungen, die der Winkelbedingung genügen.

Der Winkelbedingung (2.4) folgend ist die RICHTUNG DES STEILSTEN ABSTIEGS, $\mathbf{s} = -\nabla f(\mathbf{x}^{(k)})$, eine intuitive Wahl. Allerdings ist schon bei der Betrachtung von quadratischen Funktionen zu beobachten, dass diese Wahl eine langsame Konvergenz aufweist (siehe z. B. [72]). Eine andere häufig genutzte Wahl ist die NEWTON-RICHTUNG, die aus dem Gleichungssystem

$$\nabla^2 f(\mathbf{x}^{(k)}) \mathbf{s}^{(k)} = -\nabla f(\mathbf{x}^{(k)}) \quad (2.5)$$

bestimmt wird. Die Idee der Richtung kommt aus der Minimierung des quadratischen Modells $f(\mathbf{x}^{(k)} + \mathbf{s}^{(k)}) \approx f(\mathbf{x}^{(k)}) + \nabla f(\mathbf{x}^{(k)})^T \mathbf{s}^{(k)} + \frac{1}{2} \mathbf{s}^{(k)T} \nabla^2 f(\mathbf{x}^{(k)}) \mathbf{s}^{(k)}$ an der aktuellen Iterierten $\mathbf{x}^{(k)}$. Im Gegensatz zum Verfahren des steilsten Abstiegs, das aus einem linearen Modell hervorgeht, wählen wir eine natürliche Schrittweite von 1, welche uns zum Extremum der quadratischen Approximation führt. Das Verfahren weist lokal quadratische Konvergenz auf, was darauf beruht, dass in der Nähe des Minimums das quadratische Modell die eigentliche Zielfunktion sehr gut repräsentiert. Es sei angemerkt, dass die Newton-Richtung nur eine Abstiegsrichtung ist, wenn die Hesse-Matrix positiv definit ist. Führen wir zusätzlich eine bei $t^{(k)} = 1$ startende Schrittweitenbestimmung ein, sprechen wir vom gedämpften Newton-Verfahren. Dieses weist ein verbessertes globales Konvergenzverhalten auf

Quelltext 2.2: Modell-Algorithmus für Quasi-Newton-Verfahren

```
1 % Zielfunktion  $f: \mathbb{R}^n \rightarrow \mathbb{R}$ 
2 % Startschätzung  $\mathbf{x}^{(0)} \in \mathbb{R}^n$ 
3 function  $\mathbf{x}^{\text{ende}} = \text{QuasiNewton}(f, \mathbf{x}^{(0)})$ 
4  $k = 0$ 
5  $H^{(0)}$  positiv definit
6 while (Abbruchbedingungen nicht erfüllt)
7      $\mathbf{s}^{(k)} = -H^{(k)} \nabla f(\mathbf{x}^{(k)})$ 
8     Bestimme Schrittweite  $t^{(k)}$ 
9      $\mathbf{x}^{k+1} = \mathbf{x}^{(k)} + t^{(k)} \mathbf{s}^{(k)}$ 
10     $H^{(k+1)}$  = Aufdatierung von  $H^{(k)}$ 
11     $k = k + 1$ 
12 end
13 end
```

[32]. Der große Nachteil dieser Suchrichtung ist, dass die Berechnung der Hesse-Matrix schwierig bzw. rechenintensiv sein kann. QUASI-NEWTON Ansätze bieten eine Möglichkeit, diesen Nachteil zu umgehen. Hierbei wird eine Approximation der Hesse-Matrix oder ihrer Inversen, $B^{(k)}$ bzw. $H^{(k)}$ sowie Aufdatierungsregeln dieser, hauptsächlich basierend auf dem Gradienten, berechnet. Die Verfahren weisen eine superlineare Konvergenz auf. Durch die Vermeidung der Berechnung der Hesse-Matrix, sind Quasi-Newton-Verfahren häufig sogar effizienter. In dieser Arbeit betrachten wir das sogenannte BFGS-Verfahren.

2.1.3. Das BFGS-Verfahren

Das BFGS-Verfahren, benannt nach seinen Entwicklern Broyden, Fletcher, Goldfarb und Shanno, ist die wohl populärste Variante der Quasi-Newton-Verfahren. Nachdem wir kurz die Grundlagen der Quasi-Newton-Verfahren vorgestellt haben, werden wir die spezielle BFGS-Aufdatierungsregel motivieren. Dabei betrachten wir den Fall der Approximation der Inversen $H^{(k)}$. Dies erspart uns in der Anwendung das Lösen eines linearen Gleichungssystems zur Bestimmung der Suchrichtung. Algorithmus 2.2 zeigt den Rahmen eines typischen Quasi-Newton-Verfahrens. Der Unterschied zum Modell-Algorithmus 2.1 für Abstiegsverfahren besteht in der zusätzlichen Aufdatierung der Approximation $H^{(k)}$. Da wir eine Abstiegsrichtung suchen, müssen sowohl $H^{(0)}$ als auch alle weiteren Aufdatierungen $H^{(k)}$ symmetrisch positiv definit sein. Es sei nochmals angemerkt, dass dies beim Newton-Verfahren nicht immer der Fall sein muss. Da wir die Hesse-Matrix approximieren, verwenden wir, im Gegensatz zum Newton-Verfahren, wiederum eine Schrittweiten-

Bestimmung. Setzen wir $H^{(k)} = \nabla^2 f(\mathbf{x}^{(k)})^{-1}$, sind wir wieder beim (gedämpften) Newton-Verfahren. Die Aufdatierung beruht im Allgemeinen auf den Iterierten $\mathbf{x}^{(k+1)}, \mathbf{x}^{(k)}$ und deren Funktionswerten $f(\mathbf{x}^{(k+1)}), f(\mathbf{x}^{(k)})$ sowie den Gradienten $\nabla f(\mathbf{x}^{(k+1)}), \nabla f(\mathbf{x}^{(k)})$. Eine Taylorentwicklung von $\nabla f(\mathbf{x}^{(k+1)})$ liefert die QUASI-NEWTON-BEDINGUNG

$$\begin{aligned} H^{(k+1)} \left(\nabla f(\mathbf{x}^{(k+1)}) - \nabla f(\mathbf{x}^{(k)}) \right) &= \mathbf{x}^{(k+1)} - \mathbf{x}^{(k)} \\ \Leftrightarrow H^{(k+1)} \mathbf{y}^{(k)} &= \mathbf{d}^{(k)} \\ \Leftrightarrow \mathbf{y}^{(k)} &= B^{(k+1)} \mathbf{d}^{(k)}, \end{aligned}$$

mit $H^{(k+1)} = (B^{(k+1)})^{-1}$, für die Aufdatierungsregeln. Hierbei ist

$$\mathbf{y}^{(k)} = \nabla f(\mathbf{x}^{(k+1)}) - \nabla f(\mathbf{x}^{(k)}) \quad \text{und} \quad \mathbf{d}^{(k)} = \mathbf{x}^{(k+1)} - \mathbf{x}^{(k)}.$$

Sie ist jedoch dadurch nicht eindeutig bestimmt. Die hier vorgestellte BFGS-Aufdatierung ist eine mögliche Wahl und basiert auf dem Ansatz einer symmetrischen Rang-2-Aufdatierung

$$B^{(k+1)} = B^{(k)} + \gamma^{(k)} \mathbf{u}\mathbf{u}^T + \delta^{(k)} \mathbf{v}\mathbf{v}^T,$$

mit $\mathbf{u}, \mathbf{v} \in \mathbb{R}^n$ und $\gamma, \delta \in \mathbb{R}$. Sie resultiert aus der Bedingung, dass die aufdatierte Matrix $B^{(k+1)}$ bezüglich der gewichteten Frobenius-Norm möglichst nah an der Matrix $B^{(k)}$ liegt (siehe [72]). Dabei wird

$$\mathbf{u} = \mathbf{y}^{(k)}, \quad \mathbf{v} = B^{(k)} \mathbf{d}^{(k)}, \quad \gamma^{(k)} = \frac{1}{(\mathbf{y}^{(k)})^T \mathbf{d}^{(k)}} \quad \text{und} \quad \delta^{(k)} = -\frac{1}{(\mathbf{d}^{(k)})^T B^{(k)} \mathbf{d}^{(k)}}$$

gewählt. Eine Invertierung von $B^{(k+1)}$ mit der Sherman-Morrison-Woodbury-Formel führt zu der populären Aufdatierungsregel

$$\begin{aligned} H^{(k+1)} &= (I - \gamma^{(k)} \mathbf{d}^{(k)} (\mathbf{y}^{(k)})^T) H^{(k)} (I - \gamma^{(k)} \mathbf{y}^{(k)} (\mathbf{d}^{(k)})^T) + \gamma^{(k)} \mathbf{d}^{(k)} (\mathbf{d}^{(k)})^T \\ &= (V^{(k)})^T H^{(k)} V^{(k)} + \gamma^{(k)} \mathbf{d}^{(k)} (\mathbf{d}^{(k)})^T, \end{aligned} \quad (2.6)$$

mit

$$V^{(k)} = I - \gamma^{(k)} \mathbf{y}^{(k)} (\mathbf{d}^{(k)})^T.$$

Damit haben wir eine erste Aufdatierungsstrategie für den in Quelltext 2.2 angeführten generellen Algorithmus. $H^{(k+1)}$ bzw. $B^{(k+1)}$ sind positiv definit, wenn $\mathbf{d}^T \mathbf{y} > 0$. Sollte dies nicht der Fall sein, kann die Aufdatierung z. B. übersprungen werden. In diesem Fall wird die Suchrichtung mit Hilfe der Approximation aus der letzten Iteration berechnet. Im nächsten Schritt wird dann für die aktuelle Iterierte eine Aufdatierung versucht. Als Startapproximation wird häufig

$$H^{(0)} = \gamma^{(0)} I$$

gewählt. Im Gegensatz zur Rang-2-Aufdatierung haben Rang-1-Aufdatierungen den Nachteil, dass im Allgemeinen nicht die positive Definitheit erhalten bleibt. Die hier angeführte Aufdatierungsstrategie hat einen großen Nachteil. Die Approximationsmatrix kann dichtbesetzt werden. Bei Problemen mit vielen Unbekannten, wie in der Registrierung, wird die Berechnung teuer oder die Matrix kann erst gar nicht aufgebaut werden. Aus diesem Problem heraus entstand die Idee, Approximationen zu finden, die einfach umsetzbar sind, gleichzeitig wenig Speicher benötigen und trotzdem eine akzeptable Konvergenzrate aufweisen. Methoden, die diesen Aspekten gerecht werden, sind z. B. die Limited-Memory Quasi-Newton-Verfahren. Wir stellen hier stellvertretend den L-BFGS Algorithmus vor.

2.1.4. Das L-BFGS-Verfahren

Betrachten wir die Bestimmung der Suchrichtung $\mathbf{s}^{(k+1)}$ beim Quasi-Newton-Verfahren (Quelltext 2.2) mit der BFGS-Aufdatierungsregel (2.6), d. h.

$$\begin{aligned}
 -H^{(k+1)}\nabla f(\mathbf{x}^{(k+1)}) &= -(V^{(k)})^T H^{(k)} V^{(k)} \nabla f(\mathbf{x}^{(k)}) \\
 &\quad + \gamma^{(k)} \mathbf{d}^{(k)} (\mathbf{d}^{(k)})^T \nabla f(\mathbf{x}^{(k)}) \\
 &= -(V^{(k)})^T (V^{(k-1)})^T H^{(k-1)} V^{(k-1)} V^{(k)} \nabla f(\mathbf{x}^{(k)}) \\
 &\quad - \left(\gamma^{(k)} \mathbf{d}^{(k)} (\mathbf{d}^{(k)})^T + \gamma^{(k-1)} \mathbf{d}^{(k-1)} (\mathbf{d}^{(k-1)})^T \right) \nabla f(\mathbf{x}^{(k)}) \\
 &= \dots \\
 &= -(V^{(k)})^T \dots (V^{(0)})^T H^{(0)} V^{(0)} \dots V^{(k)} \nabla f(\mathbf{x}^{(k)}) \\
 &\quad - \sum_{t=0}^k \gamma^{(t)} \mathbf{d}^{(t)} (\mathbf{d}^{(t)})^T \nabla f(\mathbf{x}^{(k)}),
 \end{aligned}$$

sehen wir, dass - bis auf den Fall $H^{(0)}$ - nur Berechnungen von Skalarprodukten notwendig sind. Hierbei ist die Struktur von $V^{(j)}$, $\forall j = 1, \dots, k$, zu beachten. Um dies auszunutzen, müssten wir jedoch die Vektorenpaare $\{\mathbf{d}^{(j)}, \mathbf{y}^{(j)}\}$ für alle Iterationsschritte mitführen und wir bekämen wiederum ein Speicherproblem. Durch das Ausführen aller Rechenschritte in jedem Iterationsdurchlauf ist außerdem kein zeitlicher Gewinn erreicht. Die Idee des L-BFGS ist es, nicht alle dieser Vektorpaare mitzuführen, sondern nur die $p \ll n$ aktuellsten. Damit liegen wir zur Speicherung der Approximation der Hesse-Matrix bei einer Speicherplatzkomplexität von $O(pn)$

Quelltext 2.3: Zwei-Schleifen-Rekursion zur Berechnung der aktuellen Schrittweite $\mathbf{s}^{(k+1)}$ beim L-BFGS-Verfahren nach [72]

```

1  $\mathbf{q} = \nabla f(\mathbf{x}^{(k)})$ 
2 for  $i = p - k + 1 : 1 : k$ 
3      $\alpha_i = \gamma^{(i)}(\mathbf{d}^{(i)})^T \mathbf{q}$ 
4      $\mathbf{q} = \mathbf{q} - \alpha_i \mathbf{y}^{(i)}$ 
5 end
6  $\mathbf{s}^{(k+1)} = -H^{(k-p+1)} \mathbf{q}$ 
7 for  $i = p - k + 1 : -1 : k$ 
8      $\beta = \gamma^{(i)} \mathbf{y}^{(i)} \mathbf{s}^{(k+1)}$ 
9      $\mathbf{s}^{(k+1)} = \mathbf{s}^{(k+1)} + \mathbf{d}^{(i)}(\alpha_i - \beta)$ 
10 end

```

ansatz bei $O(n^2)$. Wir erhalten für die Berechnung der Suchrichtung $\mathbf{s}^{(k+1)}$

$$\begin{aligned}
 -H^{(k+1)} \nabla f(\mathbf{x}^{(k)}) &= -(V^{(k)})^T \dots (V^{(k-p+1)})^T H^{(k-p+1)} V^{(k-p+1)} \dots V^{(k)} \nabla f(\mathbf{x}^{(k)}) \\
 &\quad - \sum_{t=k-p+1}^k \gamma^{(t)} \mathbf{d}^{(t)} (\mathbf{d}^{(t)})^T \nabla f(\mathbf{x}^{(k)}).
 \end{aligned}$$

Zu klären bleibt, woher die Matrix $H^{(k-p+1)}$ kommt, unter der Bedingung, dass wir trotzdem speicherplatzeffizient bleiben. Hier wird häufig in der Praxis eine initiale Schätzung gewählt, z. B.

$$H^{(k-p+1)} = \gamma^{(k-p+1)} I = \frac{\mathbf{d}_{k-p+1}^T \mathbf{y}_{k-p+1}}{\mathbf{y}_{k-p+1}^T \mathbf{y}_{k-p+1}} I.$$

Zur Berechnung der aktuellen Suchrichtung verwenden wir eine Zwei-Schleifen-Rekursion dargestellt in Quelltext 2.3 (siehe auch [72]). Bis auf der Berechnung von $-H^{(k-p+1)} \mathbf{q}$ treten nur günstige Skalarprodukte und Vektoradditionen auf. Durch die Diagonalstruktur von $-H^{(k-p+1)}$ liegt die Berechnungskomplexität der Matrix-Vektor-Addition auch im Bereich der eines Skalarproduktes. Somit haben wir ein Verfahren, das sowohl speicher- als auch rechentechnisch eine geringe Komplexität ($O(n)$) besitzt. Es sei angemerkt, dass wir für den Fall $p \geq k$ das ursprüngliche BFGS-Verfahren betrachten.

2.1.5. Das Gauss-Newton-Verfahren

Das L-BFGS-Verfahren hat große Vorteile bezüglich der Speicher- und Rechenkomplexität, zeigt jedoch in der Praxis nicht immer ein robustes Verhalten. Daher

betrachten wir ein weiteres Verfahren, dessen Vorteile in diesem Bereich liegen, um später problemabhängig das bessere Verfahren zu identifizieren. Im Rahmen von Registrierungsproblemen betrachten wir typischerweise Least-Squares-Probleme, d. h. $\min_{\mathbf{x}} f(\mathbf{x})$ mit

$$f(\mathbf{x}) = \frac{1}{2} \|r(\mathbf{x})\|_2^2, \quad (2.7)$$

wobei $r : \mathbb{R}^n \rightarrow \mathbb{R}^m$, $r(\mathbf{x}) = (r_1(\mathbf{x}) \dots r_m(\mathbf{x}))^T$ hinreichend glatt ist. Wie das BFGS-Verfahren, basiert das Gauss-Newton-Verfahren auf dem Newton-Verfahren und approximiert die Hesse-Matrix geeignet. Dabei gehen wir hier auf die spezielle Form der Zielfunktion ein. Untersuchen wir die erste und zweite Ableitung der Zielfunktion (2.7),

$$\begin{aligned} \nabla f(\mathbf{x}) &= \sum_{i=1}^m r_i(\mathbf{x}) \nabla r_i(\mathbf{x}) = \nabla r(\mathbf{x})^T r(\mathbf{x}), \\ \nabla^2 f(\mathbf{x}) &= \nabla r(\mathbf{x})^T \nabla r(\mathbf{x}) + \sum_{i=1}^m r_i(x) \nabla^2 r_i(x), \end{aligned}$$

zeigt sich, dass der erste Term der zweiten Ableitung direkt aus der ersten Ableitung berechenbar ist. In der Praxis stellt sich heraus, dass dieser Term oft der wichtigere ist. Dies liegt daran, dass entweder die Zielfunktion fast linear und damit $\nabla^2 r_i(\mathbf{x}) \approx 0$ oder aber die Residuen $r_i(\mathbf{x})$ nahe eines lokalen Minimums sehr klein werden. Wir trennen uns also von dem zweiten Term, d. h.

$$\nabla^2 f(\mathbf{x}) \approx \nabla r(\mathbf{x})^T \nabla r(\mathbf{x}).$$

Setzen wir diese Approximation anstelle der eigentlichen Hesse-Matrix in die Berechnung der Newton-Richtung ein, führt dies zur Gauss-Newton-Richtung

$$\nabla r(\mathbf{x}^{(k)})^T \nabla r(\mathbf{x}^{(k)}) \mathbf{s}^{\text{GN}} = \nabla r(\mathbf{x}^{(k)})^T r(\mathbf{x}^{(k)}). \quad (2.8)$$

Man kann leicht nachrechnen, dass die Gauss-Newton-Richtung eine Abstiegsrichtung ist. Wie beim BFGS-Verfahren, bei dem die Approximation immer symmetrisch positiv definit ist, umgehen wir auch hier den Nachteil des Newton-Verfahrens, dass nicht immer eine Abstiegsrichtung vorhanden ist. Die Effizienz des Verfahrens ist in der Praxis ähnlich dem Newton-Verfahren. Die Konvergenz hängt davon ab, wie stark der Term $\nabla r(\mathbf{x}^{(k)})^T \nabla r(\mathbf{x}^{(k)})$ in der zweiten Ableitung dominiert. Bei einer Dominanz von $\nabla r(\mathbf{x}^{(k)})^T \nabla r(\mathbf{x}^{(k)})$ können wir eine rasante Konvergenz erwarten (bis hin zu lokal quadratisch für $\nabla^2 r_i(\mathbf{x}^*) = 0$), sonst können wir im Allgemeinen lineare Konvergenz zeigen [72].

2.1.6. Abbruchkriterien

Bisher haben wir die Zutaten des Algorithmus kennengelernt, die uns effizient in Richtung eines lokalen Minimums bringen und zwar die Suchrichtung und die Schrittweite. Allerdings bricht der Algorithmus, wie wir bei der Konvergenzaussage auf Seite 8 gesehen haben, nicht immer nach endlich vielen Schritten in einem lokalen Minimum ab. In diesem Abschnitt geben wir Abbruchkriterien, vorgeschlagen von Gill, Murray und Wright [34], für hinreichend glatte Zielfunktionen f an. Zusätzlich werden wir diese motivieren.

Fünf Bedingungen werden für den unrestringierten Fall genannt, nämlich

$$|f(\mathbf{x}^{(k-1)}) - f(\mathbf{x}^{(k)})| \leq \tau \left(1 + |f(\mathbf{x}^{(k)})|\right) \quad (\text{S1})$$

$$\|\mathbf{x}^{(k-1)} - \mathbf{x}^{(k)}\| \leq \sqrt{\tau} \left(1 + \|\mathbf{x}^{(k)}\|\right) \quad (\text{S2})$$

$$\|\nabla f(\mathbf{x}^{(k)})\| \leq \sqrt[3]{\tau} \left(1 + \|\mathbf{x}^{(k)}\|\right) \quad (\text{S3})$$

$$\|\nabla f(\mathbf{x}^{(k)})\| \leq \text{eps} \quad (\text{S4})$$

$$k \leq k_{\max}. \quad (\text{S5})$$

Der Algorithmus soll bei $((\text{S1}) \wedge (\text{S2}) \wedge (\text{S3})) \vee (\text{S4}) \vee (\text{S5})$ stoppen. Wenn die ersten drei Kriterien erfüllt sind, haben wir das gewünschte Ergebnis erreicht. Die Idee ist, einen Funktionswert zu erreichen, der nur in den letzten r Dezimalstellen variiert. Diese Forderung wird sichtbar in Bedingung (S1). Daraus folgt, dass $\tau = 10^{-r}$. Die Toleranz ist dabei, aufgrund der numerischen Probleme bei $f(\mathbf{x}^{(k)}) \approx 0$, als eine Kombination von relativem und absolutem Fehler gegeben. Da für schlechtgestellte Probleme kleine Änderungen des Funktionswertes nicht immer mit kleinen Änderungen der Iterierten Hand in Hand gehen, wurde Bedingung (S2) hinzugefügt. Die Wurzel in dieser Bedingung kommt aus der linearen Taylorapproximation, bei der die Differenz der Funktionswerte mit der Norm der Betrachtungspunkte einhergeht. Bedingung (S3) bezieht die notwendige Bedingung erster Ordnung mit ein. Die dritte Wurzel ist dabei eine heuristische Wahl. Die Bedingung (S5) ist ein Notstopp für den Fall, dass der Algorithmus nicht nach endlich vielen Schritten abbricht. Die Bedingung (S4) kontrolliert, ob die Ableitung für weitere Berechnungen zu klein wird und kann daher auch als Notstopp angesehen werden.

2.1.7. Vergleich der Verfahren

Zum Vergleich der Algorithmen wählen wir ein Beispiel aus der Registrierung. Wir betrachten dafür zwei identische 2D-Bilder und verschieben eines der beiden um

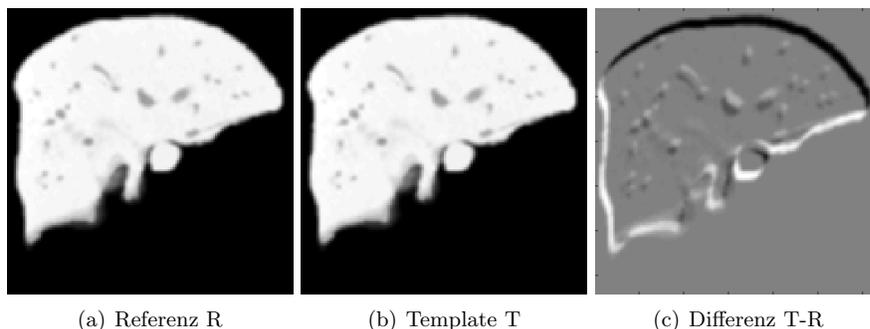


Abbildung 2.2.: Bildliche Ausgangslage des registrierungsartigen Problems $\min_{\mathbf{x}} f(\mathbf{x})$ mit f gegeben über Gleichung (2.9). Links sehen wir das Referenzbild, das um $\mathbf{x} = (-4, 2)^T$ transliert wurde (Template, mittig). Das daraus resultierende Differenzbild ist rechts zu sehen. Ziel ist es, den Translationsvektor zu finden, der die Zielfunktion f und damit die Differenz der beiden Bilder minimiert.

den Vektor \mathbf{x} . Die Ausgangslage ist in Abbildung 2.2 visualisiert. Die diskrete Zielfunktion kann angegeben werden als

$$f(\mathbf{x}) = \|\mathbf{R} - \mathbf{R} \circ t(\mathbf{x})\|_2^2, \quad (2.9)$$

wobei $\mathbf{x} \in \mathbb{R}^2$ der Translationsvektor ist und $\mathbf{R} \circ t(\mathbf{x})$ die Translation des Bildes \mathbf{R} um \mathbf{x} darstellt. Für eine genauere Darstellung des Problems verweisen wir auf das folgende Kapitel über Bildregistrierung (Kapitel 3). Die graphische Darstellung der Funktion sehen wir in Abbildung 2.3 für den interessanten Bereich der Optimierung. Als Startwert wählen wir $\mathbf{x} = (-4, 2)^T$. Das Minimum liegt durch die Identität der Bilder bei $\mathbf{x}^* = (0, 0)^T$. Wir betrachten das Verfahren des steilsten Abstiegs, das Gauss-Newton- und das BFGS-Verfahren. Die Betrachtung des L-BFGS entfällt hier, da das Problem nur zwei Unbekannte besitzt und somit die für $p \geq 2$ das L-BFGS identisch mit dem BFGS-Verfahren ist. Zur Schrittweitenbestimmung wird ein Armijo Backtracking Algorithmus benutzt, zum Abbruch die von Gill, Murray und Wright vorgeschlagenen Kriterien. Die Iterationsverläufe der drei Verfahren können wir in Abbildung 2.4 betrachten. Alle Verfahren brechen aufgrund der Erfüllung von (S1) \wedge (S2) \wedge (S3) ab und finden den Minimierer \mathbf{x}^* . Der Iterationsverlauf der Methoden ist jedoch deutlich unterschiedlich. Während das Verfahren des steilsten Abstiegs häufiger hin- und herspringt, verhalten sich die anderen beiden Methoden zielstrebiger. Dies macht sich auch bei der Anzahl der Funktionsauswertungen bemerkbar. Um den Minimierer zu finden benötigt das Gradientenverfahren 838, das Gauss-Newton-Verfahren 21 und das BFGS-Verfahren 50

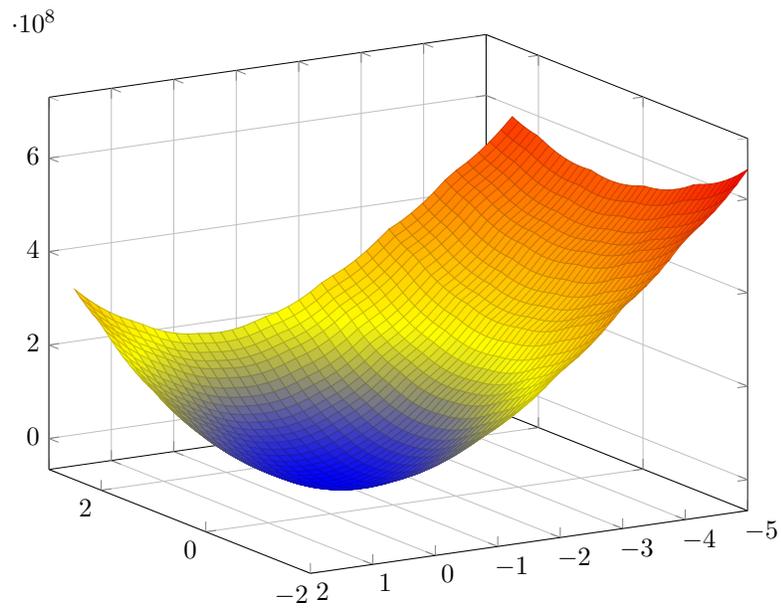


Abbildung 2.3.: Darstellung der Funktion f aus Gleichung 2.9 für kleine Translationen \mathbf{x} . Ein eindeutiges Minimum ist bei $\mathbf{x}^* = (0, 0)^T$ zu erkennen.

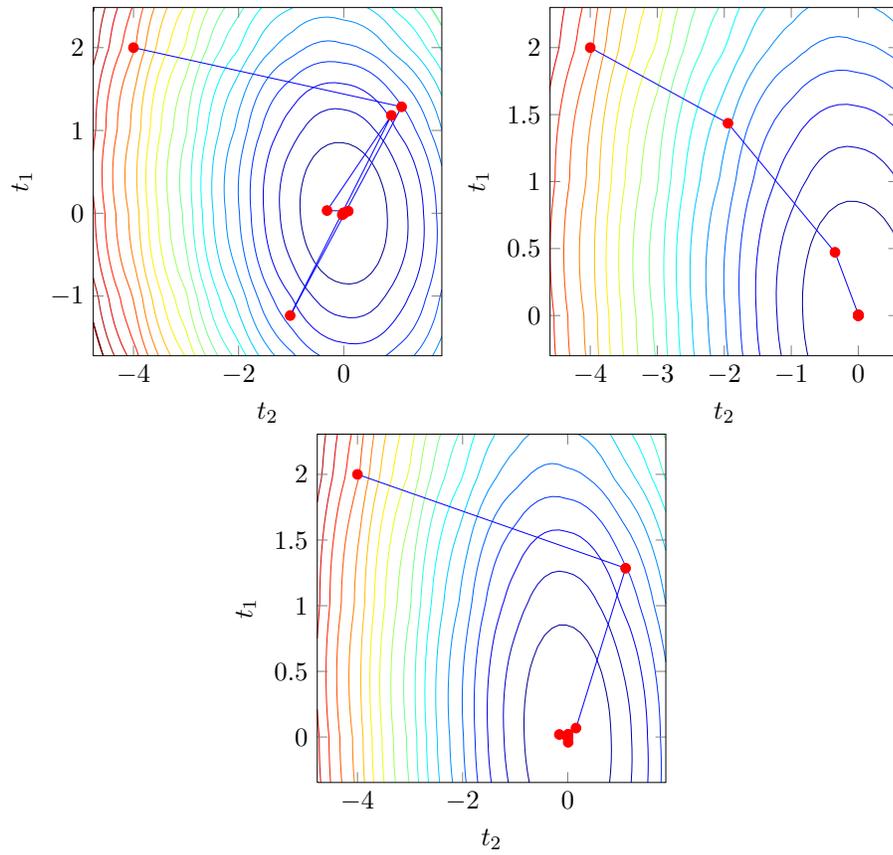


Abbildung 2.4.: Iterationsverläufe der drei Optimierungsverfahren, d. h. Gradientenverfahren (links oben), Gauss-Newton-Verfahren (rechts oben) und BFGS-Verfahren (unten).

Funktionsauswertungen. Beim Gradientenverfahren liegt dies unter anderem daran, dass wir keine Normierung des Gradienten durchgeführt haben, die bei der Berechnung der anderen beiden Methoden implizit durch das Lösen des Gleichungssystems passiert. Dadurch braucht das Verfahren viele Funktionsauswertungen bei der Schrittweitenbestimmung. Wird eine Normierung durchgeführt liegt die Anzahl der Funktionsauswertungen allerdings immer noch bei 369. Der Vorteil, weitere Informationen über den Funktionsverlauf einzubeziehen, wird deutlich sichtbar. Beim Iterationsverlauf des Gradientenverfahrens sehen wir zudem klar, dass keine optimale Schrittweiten-Bestimmung genutzt wird.

Mit dieser abschließenden Vorstellung der Algorithmen wollen wir den Abschnitt über die unrestringierte nichtlineare Optimierung abschließen und uns der restringierten Optimierung widmen.

2.2. Restringierte Optimierung

Wie wir später sehen werden, besitzt unser unrestringierter Registrierungsansatz viele lokale Minima. Die Verfahren können dann zu ungewollten Lösungen führen. Um dieses Problem anzugehen, wollen wir dem Verfahren mehr Information zukommen lassen. Eine Möglichkeit dies zu tun, ist das Hinzufügen der Informationen durch NEBENBEDINGUNGEN $c_i : \mathbb{R}^n \rightarrow \mathbb{R}$, $i = 1, \dots, m$, die an die Zielfunktion f gestellt werden. Dadurch wird der Raum der zulässigen Lösungen eingeschränkt und wir betrachten nur Lösungen, die gewisse zusätzliche Eigenschaften erfüllen. Durch die Einschränkung werden auch ungewollte lokale Minima als mögliche Lösungen ausgeschlossen. Dieser Ansatz wird für die Registrierungsprobleme in dieser Arbeit verfolgt. Daher beschäftigen wir uns auf den nächsten Seiten mit der Optimierung unter Nebenbedingungen, der RESTRINGIERTEN OPTIMIERUNG. Das Optimierungsproblem kann jetzt im Allgemeinen durch

$$\min_{\mathbf{x}} f(\mathbf{x}), \quad \text{u. d. NB } c_i(\mathbf{x}) = 0, \quad i = 1, \dots, m, \quad (2.10)$$

formuliert werden, wobei wir zweimal stetig differenzierbare Funktionen betrachten, d. h. $f, c_i \in C^2(\mathbb{R}^n, \mathbb{R})$. Die Nebenbedingungen fassen wir zusammen als Funktion $c : \mathbb{R}^n \rightarrow \mathbb{R}^m$ mit $c(\mathbf{x}) = (c_1(\mathbf{x}), \dots, c_m(\mathbf{x}))^T$. Wir betrachten in dieser Arbeit ausschließlich GLEICHHEITSNEBENBEDINGUNGEN ($c_i(\mathbf{x}) = 0$). UNGLEICHHEITSNEBENBEDINGUNGEN ($c_i(\mathbf{x}) \geq 0$) werden nur in Ausblicken erwähnt und finden sich daher nicht in diesem Abschnitt wieder. Im Folgenden wenden wir uns kurz den Grundlagen der restringierten Optimierung zu, d. h. einigen Begrifflichkeiten und den notwendigen Bedingungen für Minimierer. Gefolgt wird dies von (Standard-)Algorithmen zum Lösen restringierter Probleme, bevor wir mit einem abschließenden Beispiel enden.

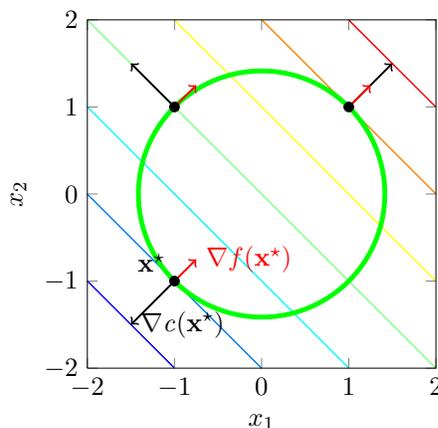


Abbildung 2.5.: Visualisierung des Beispielsproblems (2.11). Die Zielfunktion ist mittels Höhenlinien, die zulässige Menge durch den dicken, grünen Kreis dargestellt. Eingetragen sind die Gradienten der Zielfunktion (rote Pfeile) und Nebenbedingungen (schwarze Pfeile) an ausgewählten Orten. Es ist zu sehen, dass an den stationären Punkten, $(-1, -1)^T, (1, 1)^T$ die Gradienten linear abhängig sind

2.2.1. Grundlagen

Bei der Betrachtung restringierter Optimierungsprobleme sind Minimierer typischerweise nicht wie im unrestringierten Fall charakterisierbar. Dieser Abschnitt soll die formalen Grundlagen zur Charakterisierung lokaler Minima und damit der vorgestellten Algorithmen legen. Wir starten mit einem Beispiel aus [72], in dem wir erste Begrifflichkeiten einführen.

Betrachten wir das Problem

$$\min_{\mathbf{x}} x_1 + x_2 \quad \text{u. d. NB} \quad x_1^2 + x_2^2 - 2 = 0, \quad (2.11)$$

mit $f(\mathbf{x}) = x_1 + x_2$ und $c(\mathbf{x}) = x_1^2 + x_2^2 - 2$, dargestellt in Abbildung 2.5. Im unrestringierten Fall hätte die Zielfunktion f keinen Minimierer auf \mathbb{R} . Sie ist nach unten unbeschränkt. Für den Gradienten gilt $\nabla f(\mathbf{x}) = (1, 1)^T \neq 0 \forall \mathbf{x} \in \mathbb{R}$. Betrachten wir jetzt jedoch nur Punkte \mathbf{x} der ZULÄSSIGEN MENGE \mathcal{F} , d.h. $\mathcal{F} = \{\mathbf{x} : c(\mathbf{x}) = 0\}$, hat das resultierende Problem 2.11 einen Minimierer $\mathbf{x}^* = (-1, -1)^T$. Offensichtlich gilt $\nabla f(\mathbf{x}^*) \neq 0$. Allerdings zeigt sich, dass eine lineare Abhängigkeit zwischen dem Gradienten der Nebenbedingungen $\nabla c(\mathbf{x}^*)$ und dem der Zielfunktion $\nabla f(\mathbf{x}^*)$ existiert. Es gilt $\nabla f(\mathbf{x}^*) = (1, 1)^T = -\frac{1}{2}(-2, -2)^T = \nabla c(\mathbf{x}^*)$. Diese Eigenschaft spielt

eine wichtige Rolle in der Charakterisierung stationärer Punkte bei restringierten Optimierungsproblemen. Zu diesem Zwecke führen wir eine Funktion ein, die die Zielfunktion f mit den Nebenbedingungen c vereint.

Sei $\mathcal{L} : \mathbb{R}^n \times \mathbb{R}^m \rightarrow \mathbb{R}$ mit

$$\mathcal{L}(\mathbf{x}, \boldsymbol{\lambda}) := f(\mathbf{x}) + \boldsymbol{\lambda}^T c(\mathbf{x}) \quad (2.12)$$

die LAGRANGE FUNKTION mit LAGRANGE-VEKTOR $\boldsymbol{\lambda} = (\lambda_1, \dots, \lambda_m)^T \in \mathbb{R}^m$. Die Skalare λ_i , $i = 1, \dots, m$ heißen LAGRANGE-MULTIPLIKATOREN. Mit dieser Hilfsfunktion lassen sich notwendige Bedingungen erster Ordnung übersichtlich definieren.

Sei \mathbf{x}^* ein lokaler Minimierer des Problems 2.10 und seien $\nabla c_1(\mathbf{x}^*), \dots, \nabla c_m(\mathbf{x}^*)$ linear unabhängig. Dann existiert ein eindeutiger Lagrange-Vektor $\boldsymbol{\lambda}^* \in \mathbb{R}^m$, so dass

$$\nabla_{\mathbf{x}} \mathcal{L}(\mathbf{x}^*, \boldsymbol{\lambda}^*) = 0 \quad \text{und} \quad (2.13)$$

$$\nabla_{\boldsymbol{\lambda}} \mathcal{L}(\mathbf{x}^*, \boldsymbol{\lambda}^*) = 0. \quad (2.14)$$

Die lineare Abhängigkeit der Gradienten aus dem Einführungsbeispiel (2.11) spiegelt sich in Gleichung (2.13) wider. Gleichung (2.14) gibt an, dass $\mathbf{x}^* \in \mathcal{F}$, da $\nabla_{\boldsymbol{\lambda}} \mathcal{L}(\mathbf{x}^*, \boldsymbol{\lambda}^*) = c(\mathbf{x}^*)$. Die notwendige und hinreichende Bedingung zweiter Ordnung soll uns für die Algorithmenbetrachtungen hier nicht weiter interessieren. Für eine Übersicht zu diesen sei auf [72] verwiesen. Wir wenden uns nun zwei Algorithmen zu, um Probleme mit Gleichheitsnebenbedingungen zu lösen.

2.2.2. Algorithmen

Eine Idee, Probleme der Form 2.10 zu lösen, ist das restringierte Problem durch eine Folge unrestringierter Probleme zu ersetzen. Diese können dann mit den bekannten Verfahren aus dem vorherigen Abschnitt gelöst werden. Dazu konstruieren wir eine Funktion, die aus der Zielfunktion f und einem Term besteht, der Null bei Erfüllung der Nebenbedingungen wird. Im Folgenden betrachten wir zwei dieser zusätzlichen Terme und die daraus resultierenden Ansätze: Der quadratische Strafterm und der Augmented Lagrangian Ansatz. Letzterer wird später bei unseren speziellen Registrierungsproblemen zur Anwendung kommen. Der quadratische Strafterm Ansatz dient als Motivation für den Augmented Lagrangian Ansatz und ist bei der Betrachtung aktueller Verfahren in der Registrierung mit sogenannten weichen Nebenbedingungen interessant.

Quelltext 2.4: Modell-Algorithmus für den quadratischen Strafterm Ansatz

```

1 % Zielfunktion  $f: \mathbb{R}^n \rightarrow \mathbb{R}$ 
2 % Nebenbedingungen  $c: \mathbb{R}^m \rightarrow \mathbb{R}$ 
3 % Startschätzung  $\mathbf{x}^{(0)} \in \mathbb{R}^n$ ,  $\mu^{(0)} \in \mathbb{R}$ 
4 function  $\mathbf{x}^{\text{ende}} = \text{QP}(f, c, \mathbf{x}^{(0)}, \mu^{(0)})$ 
5  $k = 0$ 
6 while (Abbruchbedingungen nicht erfüllt)
7     Approximiere Minimierer  $\mathbf{x}^{(k+1)}$  von  $Q(\mathbf{x}, \mu^{(k)})$ 
8     mit Startwert  $\mathbf{x}^{(k)}$  und  $\|Q(\mathbf{x}^{(k+1)}, \mu^{(k)})\| \leq \tau^{(k)}$ 
9     Aufdatierung:  $\mu^{(k+1)} \geq \mu^{(k)}$ 
10    Wähle Toleranz  $\tau^{(k+1)}$ 
11     $k = k + 1$ 
12 end
13 end

```

Quadratischer Strafterm Ansatz Der quadratische Strafterm Ansatz ist aufgrund seiner Einfachheit und Intuitivität in der Praxis, trotz der später noch erläuterten Probleme, ein häufig gewähltes Mittel. Der Zielfunktion wird als Strafterm ein Vielfaches der quadrierten Nebenbedingungen angehängt. Es ergibt sich die QUADRATISCHE STRAFTERM FUNKTION $Q: \mathbb{R}^n \times \mathbb{R} \rightarrow \mathbb{R}$ mit

$$Q(\mathbf{x}, \mu) := f(\mathbf{x}) + \frac{\mu}{2} \|c(\mathbf{x})\|_2^2,$$

mit dem STRAFFPARAMETER $\mu > 0$.

Wie oben angesprochen, betrachten wir eine Folge von unrestringierten Problemen, $\min_{\mathbf{x}} Q(\mathbf{x}, \mu^{(k)})$, die als Grenzwert eine Lösung des restringierten Problems 2.10 besitzen soll. Damit die Lösung im zulässigen Bereich \mathcal{F} liegt, erhöhen wir dafür in jedem Schritt den Strafparameter $\mu^{(k)}$, so dass $\mu^{(k)} \rightarrow \infty$, $k \rightarrow \infty$. Der approximierete Minimierer $\mathbf{x}^{(k)}$ des Problems $\min_{\mathbf{x}} Q(\mathbf{x}, \mu^{(k)})$ dient dabei als Startwert für das Problem $\min_{\mathbf{x}} Q(\mathbf{x}, \mu^{(k+1)})$. Der Ansatz ist in Quelltext 2.4 mittels Pseudo-Code dargestellt.

Durch die Glattheit des Strafterms $\|c(\mathbf{x})\|_2^2$ können wir die in Abschnitt 2.1 betrachteten Verfahren der unrestringierten Optimierung direkt anwenden. Durch die Least-Squares Form des Strafterms ist auch das Gauß-Newton-Verfahren wieder einsetzbar. Die Wahl des Strafparameters $\mu^{(k)}$ ist anwendungsspezifisch und hängt in der Regel von der Schwierigkeit (z. B. Anzahl der Iterationsschritte) des vorherigen Problems ab. Eine typische Aufdatierungsregel ist

$$\mu^{(k+1)} = \beta \mu^{(k)}$$

mit $\beta \in [1.5, 10]$. Für die Toleranz zur Bestimmung des Abbruchs des unrestringierten Problems, $\|Q(\mathbf{x}^{(k+1)}, \mu^{(k)})\| \leq \tau^{(k)}$, soll gelten $\tau^{(k)} \rightarrow 0, k \rightarrow \infty$. Dabei bietet sich eine große Spannweite von möglichen Wahlen an. Entscheidend ist $\tau^{(k)} \rightarrow 0$.

Es kann gezeigt werden, dass das Verfahren, unter bestimmten Voraussetzungen, gegen einen Punkt \mathbf{x}^* konvergiert, der die notwendigen Bedingungen (2.13) und (2.14) mit den Lagrange-Multiplikatoren $\lambda_i^* = \lim_{k \rightarrow \infty} -\mu^{(k)} c_i(\mathbf{x}^{(k)})$ erfüllt oder ein stationärer Punkt von $\|c(\mathbf{x})\|^2$ ist [72]. Da nicht für jeden stationären Punkt \mathbf{x}^* mit $\nabla (\|c(\mathbf{x}^*)\|^2) = 0$ auch $c(\mathbf{x}^*) = 0$ gelten muss, erhalten wir als Lösung also eventuell einen Punkt, der nicht in der zulässigen Menge \mathcal{F} liegt und damit keine Lösung des Problems (2.10) ist. Zusätzlich zu diesem Aspekt wird die Hesse-Matrix, bzw. die Approximation der Hessematrix beim Gauß-Newton oder (L-)BFGS-Verfahren, für große Strafparameter μ schlecht konditioniert. Dies führt zu numerischen Instabilitäten und das Lösen des Gleichungssystems zur Bestimmung der Suchrichtung wird drastisch erschwert.

Diese Nachteile fordern eine Alternative, die sicherstellt, dass eine Lösung des Verfahrens auch eine Lösung des betrachteten Problems liefert und die gegen die schlechte Kondition der Hesse-Matrix lenkt. Zu diesem Zwecke betrachten wir den sogenannten Augmented Lagrangian Ansatz.

Augmented Lagrangian Ansatz Wie beim quadratischen Strafterm Ansatz fügen wir einen zusätzlichen Term der Zielfunktion hinzu. Wir behalten $\|c(\mathbf{x})\|_2^2$ und fügen den Term $-\boldsymbol{\lambda}^T c(\mathbf{x})$ hinzu, wobei $\boldsymbol{\lambda} \in \mathbb{R}^m$ eine Schätzung des Lagrange-Vektors ist.

Die AUGMENTED LAGRANGIAN FUNKTION $\mathcal{L}_A : \mathbb{R}^n \times \mathbb{R}^m \times \mathbb{R} \rightarrow \mathbb{R}$ ist dann gegeben durch

$$\mathcal{L}_A(\mathbf{x}, \boldsymbol{\lambda}, \mu) := f(\mathbf{x}) - \boldsymbol{\lambda}^T c(\mathbf{x}) + \frac{\mu}{2} \|c(\mathbf{x})\|_2^2. \quad (2.15)$$

Wie der Name impliziert, kann sie als Erweiterung der Lagrange Funktion (2.12) um den quadratischen Strafterm der Nebenbedingungen gesehen werden. Diese einfache Erweiterung stellt sich als hervorragende Idee heraus. Es kann gezeigt werden, dass ein Minimierer \mathbf{x}^* von (2.15), mit bekanntem Lagrange-Vektor $\boldsymbol{\lambda} = \boldsymbol{\lambda}^*$, ein strikter lokaler Minimierer des ursprünglichen Problems (2.10) ist [72]. Da der Lagrange-Vektor $\boldsymbol{\lambda}^*$ allerdings nicht bekannt ist, können wir in dem Ansatz nur mit einer Schätzungen $\boldsymbol{\lambda}^{(0)}$ beginnen. Wir benötigen daher eine Möglichkeit unsere Schätzungen im Laufe des Verfahrens zu verbessern.

Quelltext 2.5: Modell-Algorithmus für den Augmented Lagrangian Ansatz

```

1 % Zielfunktion  $f: \mathbb{R}^n \rightarrow \mathbb{R}$ 
2 % Nebenbedingungen  $c: \mathbb{R}^m \rightarrow \mathbb{R}$ 
3 % Startschätzung  $\mathbf{x}^{(0)} \in \mathbb{R}^n$ ,  $\boldsymbol{\lambda}^{(0)} \in \mathbb{R}$ ,  $\mu^{(0)} \in \mathbb{R}$ 
4 function  $\mathbf{x}^{\text{ende}} = \text{AL}(f, c, \mathbf{x}^{(0)}, \boldsymbol{\lambda}^{(0)}, \mu^{(0)})$ 
5  $k = 0$ 
6 while (Abbruchbedingungen nicht erfüllt)
7     Approximiere Minimierer  $\mathbf{x}^{(k+1)}$  von  $\mathcal{L}_A(\mathbf{x}, \boldsymbol{\lambda}^{(k)}, \mu^{(k)})$ 
8     mit Startwert  $\mathbf{x}^{(k)}$  und  $\|\mathcal{L}_A(\mathbf{x}^{(k+1)}, \boldsymbol{\lambda}^{(k)}, \mu^{(k)})\| \leq \tau^{(k)}$ 
9     Aufdatierung:  $\mu^{(k+1)} \geq \mu^{(k)}$ 
10    Aufdatierung:  $\boldsymbol{\lambda}^{(k+1)}$ 
11    Wähle Toleranz  $\tau^{(k+1)}$ 
12     $k = k + 1$ 
13 end
14 end

```

Betrachten wir dazu den Gradienten der Augmented Lagrangian Funktion

$$\begin{aligned} \nabla_{\mathbf{x}} \mathcal{L}_A(\mathbf{x}, \boldsymbol{\lambda}, \mu) &= \nabla f(\mathbf{x}) - \sum_{i=1}^m \lambda_i \nabla c_i(\mathbf{x}) + \mu \sum_{i=1}^m c_i(\mathbf{x}) \nabla c_i(\mathbf{x}) \\ &= \nabla f(\mathbf{x}) - \sum_{i=1}^m (\lambda_i - \mu c_i(\mathbf{x})) \nabla c_i(\mathbf{x}). \end{aligned} \quad (2.16)$$

Wir wissen zudem, dass für einen Minimierer \mathbf{x}^* mit eindeutig bestimmten Lagrange-Multiplikatoren λ_i^* , $i = 1, \dots, m$

$$\nabla_{\mathbf{x}} \mathcal{L} = \nabla f(\mathbf{x}^*) - \sum_{i=1}^m \lambda_i^* \nabla c_i(\mathbf{x}^*) = 0 \quad (2.17)$$

gilt. Vergleichen wir die Gleichungen (2.16) und (2.17) suggeriert dies eine Aufdatierung der Lagrange-Vektoren mittels der Regel

$$\boldsymbol{\lambda}^{(k+1)} = \boldsymbol{\lambda}^{(k)} - \mu^{(k)} c(\mathbf{x}^{(k)}). \quad (2.18)$$

Wir können nun einen Algorithmus zur Lösung des Problems (2.10) in Quelltext 2.5 angeben. Es kann unter gewissen Annahmen gezeigt werden [72], dass mit der Aufdatierungsregel (2.18) der Abstand zwischen der exakten Lösung \mathbf{x}^* und der Iterierten $\mathbf{x}^{(k)}$ beschränkt werden kann durch das Verhältnis von der Güte der Schätzung des Lagrange-Vektors $\boldsymbol{\lambda}^{(k)}$ und dem Strafparameter $\mu^{(k)}$, d. h.

$$\|\mathbf{x}^{(k)} - \mathbf{x}^*\| \leq M \frac{\boldsymbol{\lambda}^{(k)} - \boldsymbol{\lambda}^*}{\mu^{(k)}}, \quad (2.19)$$

$M > 0$. Dies zeigt, dass im Gegensatz zum quadratischen Strafterm Ansatz, bei dem die Einhaltung der Nebenbedingungen nur durch, $\mu^{(k)} \rightarrow \infty, k \rightarrow \infty$ gewährt werden konnte, es hier eine zweite Option gibt. Wir approximieren eine Lösung des Problems (2.10) auch, wenn die Schätzungen des Lagrange-Vektors gut sind. Im gleichen Zuge kann auch gezeigt werden, dass die Hesse-Matrix $\nabla_{\mathbf{x}}^2 \mathcal{L}_A(\mathbf{x}^{(k)}, \boldsymbol{\lambda}^{(k)}, \mu^{(k)})$ positiv definit ist. Dies motiviert den Einsatz der vorgestellten unrestringierten Optimierungsverfahren. Einige Punkte sind jetzt allerdings noch offen geblieben, auf die wir näher eingehen wollen.

Im Gegensatz zum (L-)BFGS-Verfahren, welches auf beliebige, hinreichend glatte Funktionen anwendbar ist, setzt das Gauß-Newton-Verfahren auf einem Least-Squares Problem auf. Beim quadratischen Strafterm Ansatz führte dies zu keinem Problem, da $Q(\mathbf{x}, \mu^{(k)})$ die entsprechende Form aufweist. Durch Hinzufügen des Terms $\boldsymbol{\lambda}^T c(\mathbf{x})$ geht diese Form verloren und die Approximation der Hessematrix von \mathcal{L}_A ist unklar. Da wir weiterhin dieses robuste Verfahren nutzen wollen, beleuchten wir diese Problematik näher.

Die Gauß-Newton-Richtung (2.8) kann neben dem vorgestellten Ansatz auch über ein lineares Modell der Residuenfunktion motiviert werden, d. h.

$$f(\mathbf{x}^{(k)} + \mathbf{s}) = \frac{1}{2} \|r(\mathbf{x}^{(k)} + \mathbf{s})\|^2 \approx \frac{1}{2} \|r(\mathbf{x}^{(k)}) + \nabla r(\mathbf{x}^{(k)})\mathbf{s}\|^2.$$

Führen wir das lineare Modell auch bei unseren Nebenbedingungen ein, $c(\mathbf{x}^{(k)} + \mathbf{s}) \approx c(\mathbf{x}^{(k)}) + \nabla c(\mathbf{x}^{(k)})\mathbf{s}$ und setzen dies in \mathcal{L}_A ein, erhalten wir

$$\begin{aligned} \mathcal{L}_A(\mathbf{x}^{(k)} + \mathbf{s}, \boldsymbol{\lambda}^{(k)}, \mu^{(k)}) &= f(\mathbf{x}^{(k)} + \mathbf{s}) - (\boldsymbol{\lambda}^{(k)})^T c(\mathbf{x}^{(k)} + \mathbf{s}) + \frac{\mu^{(k)}}{2} \|c(\mathbf{x}^{(k)} + \mathbf{s})\|^2 \\ &\approx \frac{1}{2} \|r(\mathbf{x}^{(k)}) + \nabla r(\mathbf{x}^{(k)})\mathbf{s}\|^2 - (\boldsymbol{\lambda}^{(k)})^T (c(\mathbf{x}^{(k)}) + \nabla c(\mathbf{x}^{(k)})\mathbf{s}) \\ &\quad + \frac{\mu^{(k)}}{2} \|c(\mathbf{x}^{(k)}) + \nabla c(\mathbf{x}^{(k)})\mathbf{s}\|^2. \end{aligned}$$

Differenzierung bezüglich \mathbf{s} liefert als erste Ableitung

$$\begin{aligned} \nabla_{\mathbf{s}} \mathcal{L}_A(\mathbf{x}^{(k)} + \mathbf{s}, \boldsymbol{\lambda}^{(k)}, \mu^{(k)}) &\approx \nabla r(\mathbf{x}^{(k)})^T (r(\mathbf{x}^{(k)})^T + \nabla r(\mathbf{x}^{(k)})\mathbf{s}) - (\boldsymbol{\lambda}^{(k)})^T \nabla c(\mathbf{x}^{(k)}) \\ &\quad + \mu^{(k)} \nabla c(\mathbf{x}^{(k)})^T (c(\mathbf{x}^{(k)}) + \nabla c(\mathbf{x}^{(k)})\mathbf{s}) \end{aligned}$$

und als zweite Ableitung

$$\nabla_{\mathbf{s}}^2 \mathcal{L}_A(\mathbf{x}^{(k)} + \mathbf{s}, \boldsymbol{\lambda}^{(k)}, \mu^{(k)}) \approx \nabla r(\mathbf{x}^{(k)})^T \nabla r(\mathbf{x}^{(k)}) + \mu^{(k)} \nabla c(\mathbf{x}^{(k)})^T \nabla c(\mathbf{x}^{(k)}).$$

Es ergibt sich die Suchrichtung des Gauß-Newton-Verfahrens für die Approximation des Augmented Lagrangian Ansatz als

$$\nabla_{\mathbf{s}}^2 \mathcal{L}_A(\mathbf{x}^{(k)} + \mathbf{s}, \boldsymbol{\lambda}^{(k)}, \mu^{(k)})\mathbf{s} = -\nabla_{\mathbf{s}} \mathcal{L}_A(\mathbf{x}^{(k)} + \mathbf{s}, \boldsymbol{\lambda}^{(k)}, \mu^{(k)}).$$

2. Optimierung

Ein weiterer offener Punkt ist die Aufdatierung des Strafparameters μ . In [5] werden dazu folgende zwei Strategien vorgeschlagen. Die erste ist, den Parameter, wie beim quadratischen Strafterm Ansatz vorgeschlagen, monoton zu erhöhen, d. h.

$$\mu^{(k+1)} = \beta\mu^{(k)}.$$

Dabei ist in der Praxis eine typische Wahl $\beta \in [4, 10]$. Um große Parameter μ und damit eine einhergehende schlechte Kondition der Hessematrix zu vermeiden, kann die Wahl auch abhängig zur aktuellen Verletzung der Nebenbedingungen erfolgen, z. B. durch

$$\mu^{(k+1)} = \begin{cases} \beta\mu^{(k)} & c(\mathbf{x}^{(k)}) > \gamma c(\mathbf{x}^{(k-1)}) \\ \mu^{(k)}, & \text{sonst.} \end{cases}$$

Eine in der Praxis gebräuchliche Wahl ist $\beta = 10$ und $\gamma = \frac{1}{4}$. Unter der Annahme, dass die Folge $\{\boldsymbol{\lambda}^{(k)}\}$ beschränkt ist, kann die Beschränktheit der $\{\mu^{(k)}\}$ nachgewiesen werden.

Es bleibt die Wahl der Startwerte $\mu^{(0)}, \boldsymbol{\lambda}^{(0)}$. Ungleichung (2.19) suggeriert eine große Auswirkung des Startparameters $\boldsymbol{\lambda}^{(0)}$. Es zeigt sich, dass die Effizienz des Verfahrens tatsächlich stark von der Wahl dieser Parameter abhängt. Eine Wahl von $\boldsymbol{\lambda}^{(0)}$ nahe $\boldsymbol{\lambda}^*$ erhöht drastisch die Effizienz der Methode. Leider ist uns keine Vorschrift zur Wahl von $\boldsymbol{\lambda}^{(0)}$ bekannt. Da für einen lokalen Minimierer \mathbf{x}^* mit Lagrange-Vektor $\boldsymbol{\lambda}^*$ die Gleichheit $\nabla f(\mathbf{x}^*) = -\boldsymbol{\lambda}^* \nabla c(\mathbf{x}^*)$ gilt und die Vorzeichen stark variieren, bietet es sich an, eine Startschätzung von

$$\boldsymbol{\lambda}^{(0)} = 0$$

zu wählen. Dadurch wird die Abweichung der einzelnen Komponenten im Mittel verringert.

Bei der Wahl von $\mu^{(0)}$ gilt es eine Balance zwischen verschiedenen Faktoren zu erlangen. Zum einen sollte die Startwahl nicht zu hoch ausfallen, um nicht in den Bereich der schlechten Konditionierung zu kommen. Zum anderen sollte die Wahl nicht zu niedrig ausfallen, um nicht zu weit von den Nebenbedingungen abzuweichen und nach Ungleichung (2.19) eine gute Startapproximation zu gewinnen. Der Abbruch des Verfahrens erfolgt nach den vorgestellten Kriterien von Gill, Murray und Wright, (S1) - (S5) auf \mathcal{L}_A , erweitert um die Bedingung

$$c(\mathbf{x}^k) \leq \tau_c, \tag{S6}$$

$\tau_c \in \mathbb{R}$. Es ergibt sich das Abbruchkriterium

$$((S1) \wedge (S2) \wedge (S3) \wedge (S6)) \vee (S4) \vee (S5).$$

Zum Abschluss des Abschnitts über restringierte Optimierung betrachten wir das leicht modifizierte Beispiel aus den Grundlagen, um uns mit dem Verfahren vertraut zu machen.

2.2.3. Augmented Lagrangian - Ein Beispiel

Wir untersuchen im Folgenden den Augmented Lagrangian Ansatz mit dem Gauß-Newton- und BFGS-Verfahren. Da diese auf Approximationen der Hesse-Matrix arbeiten, verwenden wir nicht das exakte Beispielpfad (2.11) mit der linearen Zielfunktion aus den Grundlagen, sondern

$$\min_{\mathbf{x}} x_1^2 + x_2^2 \quad \text{u. d. NB} \quad (x_1 - 2)^2 + (x_2 - 2)^2 - 2 = 0. \quad (2.20)$$

Der eindeutige globale Minimierer der Zielfunktion $f(\mathbf{x}) = x_1^2 + x_2^2$ liegt dabei nicht im zulässigen Bereich. Man sieht leicht, dass der Minimierer des Problems (2.20) $\mathbf{x}^* = (1, 1)^T$ mit Lagrange-Multiplikator $\lambda^* = -1$ ist. Im Folgenden wollen wir die Startwahl des Lagrange-Multiplikators und des Strafparameters sowie das Verhalten der beiden unrestringierten Algorithmen untersuchen. Wir erweitern die Abbruchkriterien aus Quelltext 2.5 und brechen die unrestringierte Optimierung bei Eintreten der Abbruchkriterien nach Gill, Murray und Wright ((S1)-(S5)) ab. Für die äußere Schleife fügen wir das Kriterium (S6) den Abbruchbedingungen hinzu. Von den strengeren Abbruchkriterien erwarten wir eine schnellere Konvergenz. Die Toleranzen der Abbruchkriterien sind $\tau = 10^7, \tau_c = 10^{-9}$ bei einer maximalen Anzahl an Iterationsschritten von 500. Die Iterationsverläufe für festes $\mu^{(0)}$ und variierendes $\lambda^{(0)}$ sind Abbildung 2.6 zu entnehmen. Es zeigt sich, dass ein guter Startwert sich hier sehr positiv auf das Verfahren auswirkt. Besonders deutlich zeigt sich dies bei der Wahl des optimalen Lagrange-Multiplikators $\lambda^{(0)} = \lambda^* = -1$ in Verbindung mit dem BFGS-Verfahren. Aber auch für die Startwahl $\mu^{(0)}$ zeigen sich die angesprochenen Punkte (siehe Abbildung 2.7). Ist die Startwahl zu niedrig, entfernen wir uns zum Anfang deutlich von den Nebenbedingungen. Ist die Wahl zu hoch sind wir durchgehend dicht an den Nebenbedingungen, allerdings ist die Optimierung deutlich schwieriger und viele Schritte werden benötigt.

Das Beispiel zeigt deutlich, wie wichtig die Startwahl $\lambda^{(0)}, \mu^{(0)}$ ist. Wir sollten jedoch beachten, dass es in unseren Anwendungen schwierig ist, diese von vornherein zu finden.

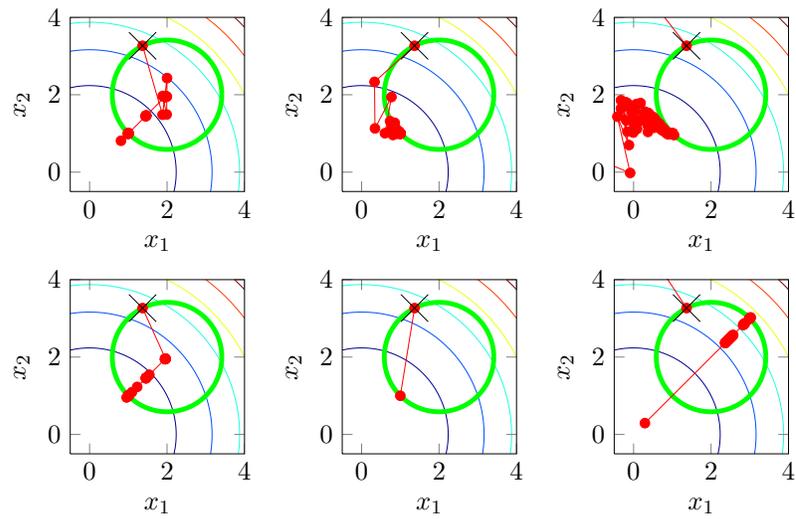


Abbildung 2.6.: Testdurchläufe für das Problem (2.20) mit $\mu^{(0)} = 1$ und $\lambda^{(0)} = -50, -1, 50$ (von links nach rechts). In der oberen Reihe sehen wir die Ergebnisse mit dem Gauß-Newton-Verfahren als unrestringierter Optimierer innerhalb des Augmented Lagrangian Rahmens, in der unteren Reihe das BFGS-Verfahren.

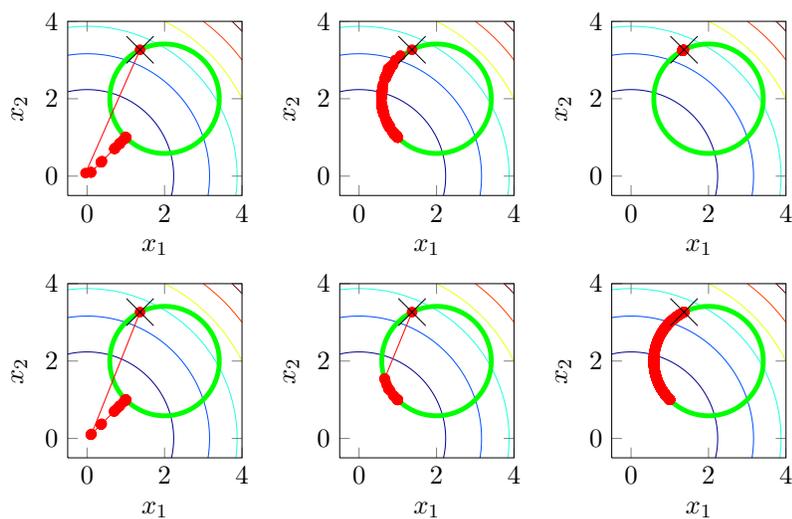


Abbildung 2.7.: Testdurchläufe für das Problem (2.20) mit $\lambda^{(0)} = 0$ und $\mu^{(0)} = 10^{-2}, 10^2, 10^6$ (von links nach rechts). In der oberen Reihe sehen wir die Ergebnisse mit dem Gauß-Newton-Verfahren als unrestringierten Optimierer innerhalb des Augmented Lagrangian Rahmens, in der unteren Reihe das BFGS-Verfahren.

2.3. Zusammenfassung

Übersicht - Grundlagen der Optimierung	
Begriff	Notation
Zielfunktion	$f : \mathbb{R}^n \rightarrow \mathbb{R}$
Minimierer	$\mathbf{x}^* \in \mathbb{R}^n$
Nebenbedingungen	$c_i : \mathbb{R}^n \rightarrow \mathbb{R}, i = 1, \dots, m$ $c : \mathbb{R}^n \rightarrow \mathbb{R}^m,$ $c(\mathbf{x}) = (c_1(\mathbf{x}), \dots, c_m(\mathbf{x}))^T$
Zulässige Menge	$\mathcal{F} = \{\mathbf{x} : c(\mathbf{x}) = 0\}$
Schrittweite	$t \in \mathbb{R}$
Suchrichtung	$\mathbf{s} \in \mathbb{R}^n$
Toleranzparameter Abbruch	$\tau \in \mathbb{R}$
Toleranzparameter NB	$\tau_c \in \mathbb{R}$
Lagrange-Multiplikatoren	$\lambda_i \in \mathbb{R}, i = 1, \dots, m$
Lagrange-Vektor	$\boldsymbol{\lambda} \in \mathbb{R}^m$ $\boldsymbol{\lambda} = (\lambda_1, \dots, \lambda_m)^T$
Strafparameter	$\mu \in \mathbb{R}$
Lagrange Funktion	$\mathcal{L} : \mathbb{R}^n \times \mathbb{R}^m \rightarrow \mathbb{R},$ $\mathcal{L}(\mathbf{x}, \boldsymbol{\lambda}) = f(\mathbf{x}) + \boldsymbol{\lambda}^T c(\mathbf{x})$
Quadratische Strafterm	$Q : \mathbb{R}^n \times \mathbb{R} \rightarrow \mathbb{R},$
Funktion	$Q(\mathbf{x}, \mu) = f(\mathbf{x}) + \frac{\mu}{2} \ c(\mathbf{x})\ ^2$
Augmented Lagrangian	$\mathcal{L}_A : \mathbb{R}^n \times \mathbb{R}^m \times \mathbb{R} \rightarrow \mathbb{R},$
Funktion	$\mathcal{L}_A(\mathbf{x}, \boldsymbol{\lambda}, \mu) = f(\mathbf{x}) + \boldsymbol{\lambda}^T c(\mathbf{x}) + \frac{\mu}{2} \ c(\mathbf{x})\ ^2$
Unrestringiertes Problem	$\min_{\mathbf{x}} f(\mathbf{x})$
Restringiertes Problem	$\min_{\mathbf{x}} f(\mathbf{x})$ u. d. NB $c(\mathbf{x}) = \mathbf{0}$

Wir haben im vorangegangenen Kapitel die Grundlagen der unrestringierten und restringierten Optimierung für Gleichheitsnebenbedingungen dargestellt. Dabei haben wir den Schwerpunkt auf die numerischen Verfahren zur Lösung der vorgestellten Probleme gelegt. Bei der unrestringierten Optimierung sind das Gauß-Newton- und das (L-)BFGS-Verfahren betrachtet worden. Diese finden auch wieder ihren Platz in der restringierten Optimierung im Rahmen des Augmented Lagrangian Ansatzes, der eine Folge von unrestringierten Problemen betrachtet. Zum Abschluss fassen wir kurz die Notation des Kapitels zusammen. Sprechen wir von restringier-

ten Problemen, fällt die Betrachtung von Ungleichheitsnebenbedingungen weg, es sei denn, diese sind explizit erwähnt.

3. Bildregistrierung

Bildregistrierung ist ein herausforderndes Thema im Bereich der Bildverarbeitung und häufig ein kritischer Schritt in einer Bildverarbeitungskette. Eine Bildregistrierung wird notwendig, wenn zwei Bilder verglichen werden sollen. Korrespondierende Punkte liegen bei zwei Aufnahmen allerdings nicht immer an derselben Stelle. Dies kann daran liegen, dass eine Aufnahme eines Patienten mit der eines anderen Patienten verglichen werden soll. Diese Bilder unterscheiden sich schon alleine dadurch, dass das betrachtete Objekt nicht gleich ist. Doch auch bei einem Vergleich zweier Aufnahmen eines Patienten kommt die Bildregistrierung ins Spiel. Liegen die Aufnahmen zeitlich auseinander, wird der Patient zum einen nicht die gleiche Position einnehmen, zum anderen kann auch eine andere Modalität ins Spiel kommen. So kann die erste Aufnahme eine Computertomographie (CT), die zweite hingegen eine Magnetresonanztomographie (MRT) sein. Wir sprechen dann von einer multimodalen Aufnahme, bei der sowohl die Position als auch die Darstellung des aufgenommenen Bereichs sich unterscheidet. In manchen Fällen unterscheidet sich sogar die Anatomie, z. B. bei Aufnahmen vor und nach einer Leberresektion. Abbildung 3.1 veranschaulicht zwei dieser Fälle. Hierbei ist vor allem interessant, dass es nicht zu jedem Punkt in dem einen Bild einen korrespondierenden Punkt in dem anderen Bild gibt. Bei der Bildregistrierung wird nun eines der Bilder, bezeichnet als REFERENZBILD, fixiert. Dann suchen wir eine Transformation, die das andere Bild, das TEMPLATEBILD, deformiert, so dass korrespondierende Punkte an derselben Stelle liegen und die Bilder auf eine sinnvolle Weise möglichst ähnlich werden. Wir können das Registrierungsproblem also folgendermaßen formulieren:

“Finde eine plausible Deformation φ , so dass korrespondierende Punkte im Referenzbild \mathcal{R} und im Templatebild \mathcal{T} möglichst gut aufeinander abgebildet werden.“

Hierfür werden wir für die auftretenden Probleme jeweils ein spezielles Modell aufstellen. Die Zielsetzung werden wir dann als variationelles Problem formulieren, bei dem wir einen Minimierer eines Funktionals $\mathcal{J} : \mathbb{F} \rightarrow \mathbb{R}$ suchen, d. h.

$$\min_{\varphi \in \mathbb{F}} \mathcal{J}(\varphi), \tag{3.1}$$

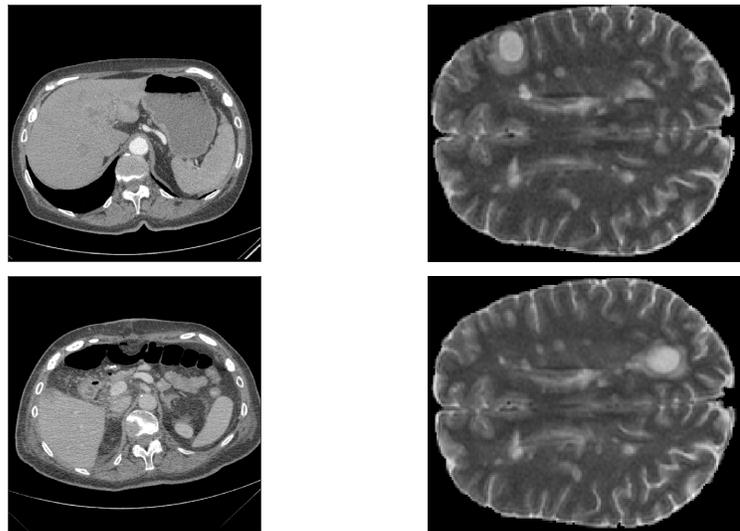


Abbildung 3.1.: CT-Aufnahme des Abdomens (links) und MRT-Aufnahmen des Gehirns (rechts). Die Aufnahmen stammen jeweils von einem Patienten, sind allerdings zu unterschiedlichen Zeitpunkten aufgenommen. Beim linken Patienten liegt eine Leberresektion zwischen den Aufnahmen, beim rechten Patienten sehen wir durch Multiple Sklerose verursachte Läsionen an zwei Zeitpunkten. Ziel der Registrierung ist es nun, korrespondierende Punkte der Aufnahmen aufeinander abzubilden.

wobei wir eine Menge \mathbb{F} von zulässigen, plausiblen Funktionen betrachten. Um dieses Problem zu lösen, werden wir nach einer kontinuierlichen Formulierung eine Diskretisierung durchführen. Das diskretisierte Problem können wir dann mit wohlbekannten Methoden aus der numerischen Optimierung (siehe Kapitel 2) lösen. Wir verfolgen einen diskretisieren-dann-optimieren Ansatz (siehe hierzu [68]). Für andere Ansätze verweisen wir beispielsweise auf [97, 66]. Um diesem Weg zu folgen, benötigen wir einige Konzepte. Neben einem Überblick über die Diskretisierungswerkzeuge, sind dies eine Definition eines Bildes und der Deformation. Die wörtliche Problemformulierung lässt zudem Fragen offen, was “möglichst gut“ für die Korrespondenzen und “plausibel“ im Kontext einer Deformation bedeutet. Dies wird uns zu Distanzmaßen und Regularisierern bringen, sowie das Einbringen von Zusatzwissen motivieren. Dadurch bekommen wir die Bausteine für die Konstruktion unserer problemangepassten Modelle im Anwendungsteil dieser Arbeit. Folgend der generellen Idee, werden wir zuerst die kontinuierliche Formulierung der Sachverhalte, gefolgt von der Diskretisierung dieser vorstellen. Abschließend betrachten wir noch spezielle Registrierungstechniken, um robuste Verfahren zu ermöglichen.

3.1. Bilder und Gitter

Zum Anfang widmen wir uns einem zentralen Punkt in der Bildregistrierung. Dies ist die Frage:

“Was ist ein Bild?“

Dabei wollen wir ausschließlich Grauwertbilder behandeln. Der Anschaulichkeit halber beschäftigen wir uns im Folgenden mit dem zweidimensionalen Fall ($d = 2$). Die Übertragung in den dreidimensionalen Fall ($d = 3$) verläuft analog.

Bilder liegen typischerweise diskret, in Form einer Matrix \mathbf{T} vor. Dies wird in Abbildung 3.2 veranschaulicht. Eine Röntgenaufnahme ist für die Entstehung des diskreten Bildes ein gut zugängliches Beispiel. Wir betrachten dafür ein stark vereinfachtes Modell. Für nähere Informationen verweisen wir z. B. auf [11].

Von einer Röntgenquelle geht Röntgenstrahlung in Richtung des Objektes. Auf dem Weg durch das Objekt wird Strahlung absorbiert. Die Stärke der Absorption hängt dabei von dem durchdrungenen Material ab. Auf der anderen Seite des Objektes befindet sich ein Detektor. Dieser misst die Stärke der verbleibenden Strahlung. Der Detektor ist in Zellen unterteilt. Für jede Zelle bekommen wir einen Wert für die Absorption. Wir erhalten also die Matrix \mathbf{T} , die unser Bild darstellt. In diesem Fall handelt es sich um eine Projektion des dreidimensionalen Objektes auf einen

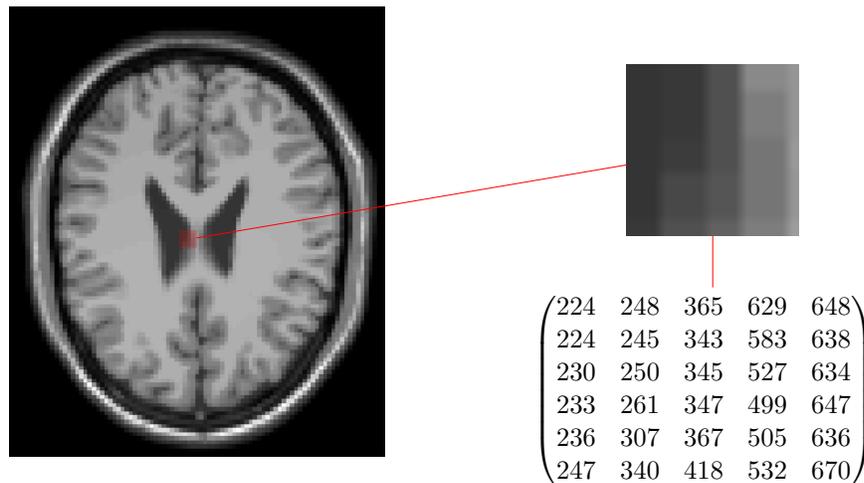


Abbildung 3.2.: Ein diskretes (zweidimensionales) Bild kann als Matrix aufgefasst werden, in der Grauwerteinträge stehen. Hier sehen wir eine Schicht einer MRT-Aufnahme eines Gehirns (links). Eine Vergrößerung des roten Kästchens ist rechts oben abgebildet. Hier sehen wir Flächen, deren Wert konstant ist, die Voxel. Der Grauwert für die einzelnen Voxel ist in der Matrix unten rechts eingetragen. Durch die Matrix können wir also unser Bild darstellen.

zweidimensionalen Detektor. Die Geometrie des Detektors erlaubt es uns, das Bild mit weiteren Attributen zu versehen. Wir bezeichnen den Bereich des Detektors (im Zweidimensionalen gesehen) als **URBILDBEREICH** $\Omega = (\omega_1, \omega_2) \times (\omega_3, \omega_4) \subset \mathbb{R}^2$ des Bildes \mathbf{T} . Ist eine derartige Information nicht gegeben, kann dieser Bereich beliebig gewählt werden, beispielsweise mit $\Omega = (0, 1) \times (0, 1)$. Wie wir gesehen haben, ist dieser Bereich in unserem Beispiel in $m_1 \cdot m_2$ Zellen der Größe $\bar{h} = h_1 \cdot h_2$ unterteilt. Bei der Zelle sprechen wir im Zusammenhang mit Bildern von einem **VOXEL**, unabhängig von der Dimension. Wir nennen $m = (m_1, m_2)$ die **BILDGRÖSSE** und $h = (h_1, h_2)$ die **VOXELGRÖSSE**. Abbildung 3.3 visualisiert diese Konzepte. Wir bezeichnen die Matrix $\mathbf{T} \in \mathbb{G}^{m_1 \times m_2}$ als **DISKRETES BILD**. Diskrete Bilder werden häufig auch als Vektoren $\mathbf{T} \in \mathbb{G}^{m_1 m_2}$ aufgefasst. Dazu wird die Matrix spaltenweise in einen Vektor übertragen. Aus dem Kontext geht hervor, ob das Bild in dieser Arbeit in Matrix- oder Vektorform genutzt wird. Die Menge $\mathbb{G} \subset \mathbb{R}$ ist dabei die Menge der **GRAUWERTE**, z. B. $\mathbb{G} = \{0, 1, \dots, 255\}$.

Im Zuge der Registrierung sollen Bilder aneinander ausgerichtet werden, z. B. durch eine Rotation. Was dies für unser Bild bzw. die Matrix bedeutet, ist nicht klar. Zur Modellierung führen wir deshalb ein kontinuierliches Modell ein. Dafür benötigen wir auch ein kontinuierliches Modell unseres Bildes.

Das **KONTINUIERLICHE BILD** \mathcal{T} ist dann definiert durch

$$\begin{aligned} \mathcal{T} : \Omega &\rightarrow \mathbb{G} \\ \mathbf{x} &\mapsto \mathcal{T}(\mathbf{x}). \end{aligned} \tag{3.2}$$

Abbildung 3.4 zeigt ein Bild, das in seiner natürlichen (bildlichen) Form und als Funktion dargestellt wird.

Um eine kontinuierliche Darstellung \mathcal{T} des diskreten Bildes \mathbf{T} zu gewinnen, werden wir eine Interpolation nutzen. Dafür benötigen wir jedoch eine Zuordnung unserer Matrixeinträge zu Positionen im Urbildbereich Ω . Mit dieser beschäftigen wir uns bevor wir zur Interpolation kommen.

Betrachten wir wieder den Detektor aus dem Beispiel. Die Zellen können dabei z. B. durch die Zellmittel- oder Eckpunkte beschrieben werden. Für die Zuordnung der Matrixeinträge zu Positionen im Urbildbereich wollen wir die Zellmittelpunkte nutzen. Die Eckpunkte der Zellen spielen dagegen später bei den Transformationen eine wichtige Rolle und werden daher auch hier schon eingeführt.

Die Menge

$$X^{cc}(\Omega, m) = \{\mathbf{x}_{i_1, i_2}^{cc} \mid i_1 = 0, \dots, m_1 - 1, i_2 = 0, \dots, m_2 - 1\} \tag{3.3}$$

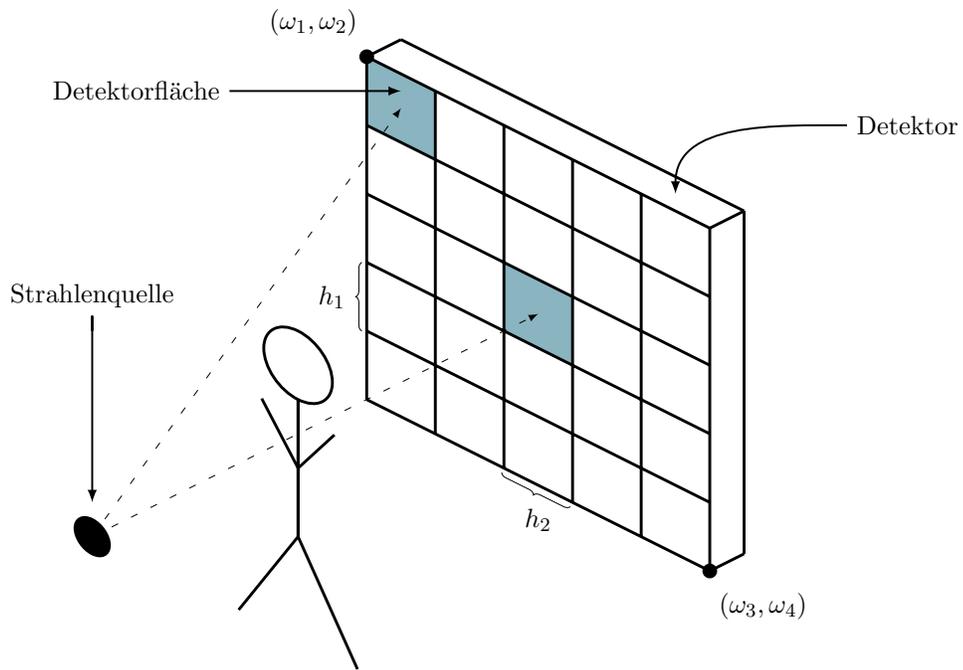


Abbildung 3.3.: Entstehung eines Bildes durch eine Projektion auf einen Detektor. Die in der Strahlenquelle erzeugt Strahlung wird auf ihrem Weg durch den Patienten abgeschwächt. Wie stark die Abschwächung ist, hängt von dem durchquerten Material ab. Die abgeschwächte Strahlung wird dann am Detektor gemessen. Dabei werden Werte für die einzelnen Detektorflächen in der Bildmatrix gespeichert. Diese ergeben die Grauwerte. Durch den Detektor ist auch eine örtliche Zuordnung der Grauwerte möglich. So ist der Urbildbereich Ω durch die Eckpunkte (ω_1, ω_2) und (ω_3, ω_4) gegeben.

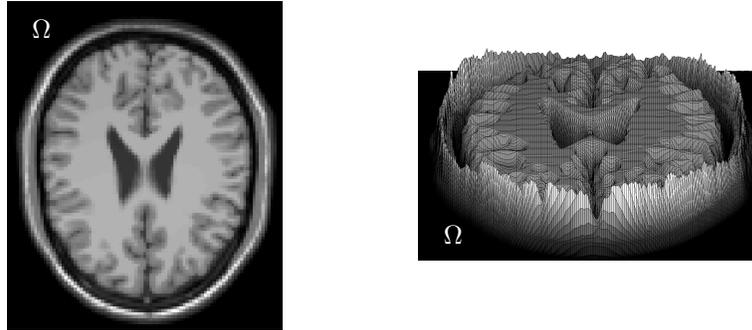


Abbildung 3.4.: Darstellung eines Bildes, gegeben im Bildbereich Ω . Links sehen wir die natürliche Form des Bildes, rechts eine kontinuierliche Darstellung als Funktion $\mathcal{T} : \mathbb{R}^2 \rightarrow \mathbb{R}$.

nennen wir ZELLENZENTRIERTES GITTER. Dabei sind die Zellmittelpunkte $\mathbf{x}_{i_1, i_2}^{cc}$ gegeben durch

$$\mathbf{x}_{i_1, i_2}^{cc} = \left(x_1^{(0)} + i_1 h_1, x_2^{(0)} + i_2 h_2 \right) \in \mathbb{R}^2 \quad (3.4)$$

mit $\mathbf{x}^{(0)} = (x_1^{(0)}, x_2^{(0)}) = (\omega_1 + \frac{h_1}{2}, \omega_3 + \frac{h_2}{2})$.

Die Menge

$$X^n(\Omega, m) = \{ \mathbf{x}_{i_1, i_2}^n \mid i_1 = 0, \dots, m_1, i_2 = 0, \dots, m_2 \} \quad (3.5)$$

der Zelleckpunkte nennen wir NODALES GITTER, wobei

$$\mathbf{x}_{i_1, i_2}^n = (\omega_1 + i_1 h_1, \omega_3 + i_2 h_2) \in \mathbb{R}^2. \quad (3.6)$$

Wir können Abbildung 3.5 für ein Beispiel dieser beiden Gitter mit $\Omega = (1, 3) \times (2, 8)$, $m = (2, 3)$ und $h = (1, 2)$ betrachten.

Für die spätere Verarbeitung der Gitter sollen die Gitter als Vektoren $\vec{\mathbf{x}}^{cc}$ bzw. $\vec{\mathbf{x}}^n$ dargestellt werden. Wir stellen die Idee anhand des zellzentrierten Gitters unseres Beispiels (Abbildung 3.5) vor. Dafür bringen wir die Elemente der Menge $X^{cc}(\Omega, m)$ in lineare Ordnung und schreiben die Komponenten der Dimension untereinander

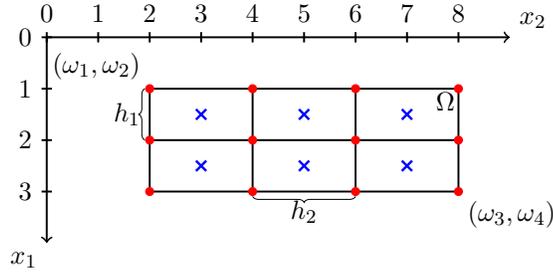


Abbildung 3.5.: Eine Unterteilung des Gebietes $\Omega = (1, 3) \times (2, 8)$ in 2×3 Zellen. Die Voxelgröße h ist dabei gegeben durch $h_1 = 1, h_2 = 2$. Das zellzentrierte Gitter ist visualisiert durch die blauen Kreuze, das nodale Gitter durch die roten Punkte.

weg. Für unserer Beispiel ergibt sich

$$X^{cc}(\Omega, m) \rightarrow \begin{matrix} (1.5, 3) \\ (2.5, 3) \\ (1.5, 5) \\ (2.5, 5) \\ (1.5, 7) \\ (2.5, 7) \end{matrix} \rightarrow \begin{pmatrix} 1.5 \\ 2.5 \\ 1.5 \\ 2.5 \\ 3 \\ 3 \\ 5 \\ 5 \\ 7 \\ 7 \end{pmatrix} = \bar{\mathbf{x}}^{cc}. \quad (3.7)$$

Damit haben wir die nötigen Notationen eingeführt und können uns nun mit der Interpolation beschäftigen, um eine kontinuierliche Repräsentante des diskreten Bildes zu erlangen.

3.2. Interpolation

Anstatt uns direkt der Interpolation von zweidimensionalen Bildern zu widmen, starten wir der Übersicht halber mit einem eindimensionalen Beispiel. Dafür betrachten wir beispielhaft eine Zeile $\mathbf{t} = (t_0, \dots, t_{m_2-1})^T \in \mathbb{G}^{m_2}$ des diskreten Bildes $\mathbf{T} \in \mathbb{G}^{m_1 \times m_2}$ mit $\Omega = (0, m_1) \times (0, m_2)$ gegeben auf einem zellzentrierten Gitter

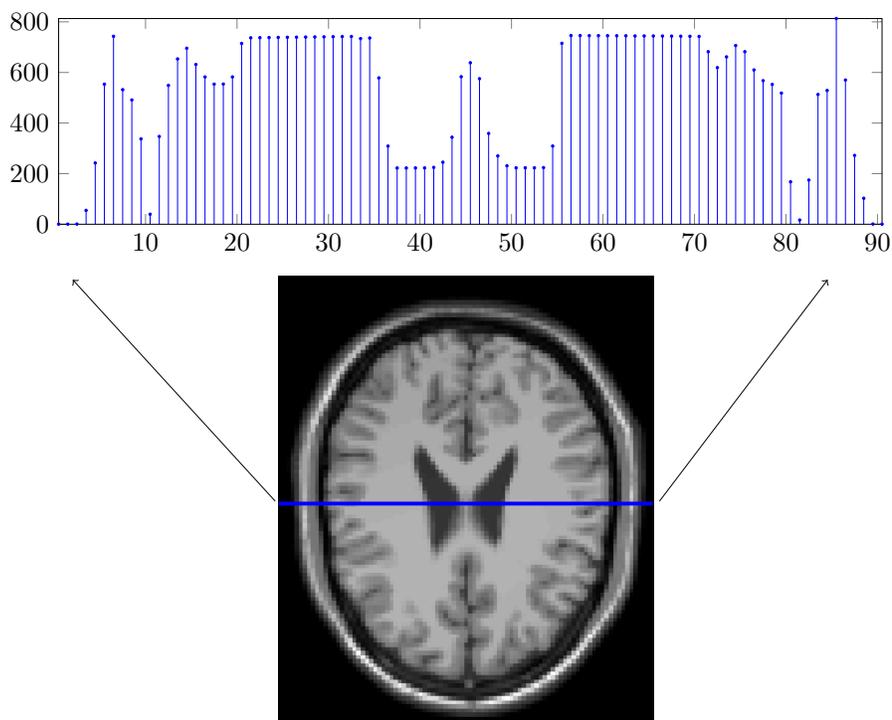


Abbildung 3.6.: Zeile $\mathbf{t} \in \mathbb{R}^{91}$ (oben) aus einem zweidimensionalen diskreten Bild $\mathbf{T} \in \mathbb{G}^{109 \times 91}$ (unten) mit $\Omega = (0, 109) \times (0, 91)$. Die Bildzeile \mathbf{t} ist dabei gegeben auf dem zellzentrierten Gitter $X^{cc}((0, 91), 91)$.

$X^{cc}(\Omega, m)$ mit $h = (h_1, h_2) = (1, 1)$. Die betrachteten Daten \mathbf{t} sind also gegeben auf dem zellzentrierten Gitter $X^{cc}(\omega, m_2)$ mit $\omega = (0, m_2)$. Hierbei ist $X^{cc}(\omega, m_2)$ das eindimensionale Äquivalent zu (3.3). Die Elemente der Menge sind mit x_k^{cc} gegeben. Ein Beispiel hierfür ist in Abbildung 3.6 zu betrachten. Die folgenden Ausführungen orientieren sich an [51].

Das INTERPOLATIONSPROBLEM besteht darin, eine Funktion $\mathcal{T} : \mathbb{R} \rightarrow \mathbb{R}$ zu finden, für die die INTERPOLATIONSBEDINGUNGEN $I(x_k^{cc}) = t_k$ für alle $k = 0, \dots, m_2 - 1$ erfüllt sind.

Die INTERPOLATIONSFUNKTION \mathcal{T} ist definiert als gewichtete Summation von Basisfunktionen $b_k : \mathbb{R} \rightarrow \mathbb{R}$, d. h.

$$\mathcal{T} = \sum_k c_k b_k, \quad (3.8)$$

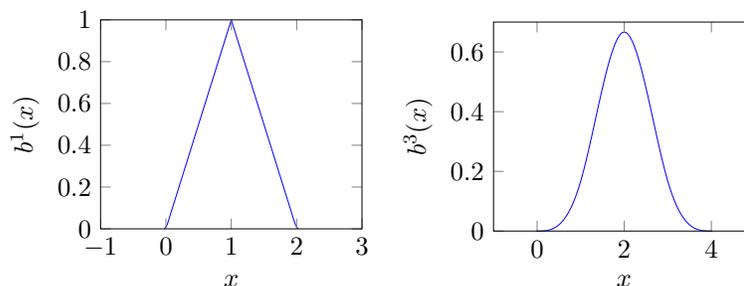


Abbildung 3.7.: Visualisierung des linearen b^1 (links) und kubischen b^3 (rechts) B-Splines nach Gleichung (3.2) bzw. (3.2). Man sieht, dass b^n einen lokalen Träger von auf $[0, n + 1]$ besitzt und im Falle des linearen B-Splines 0-mal stetig differenzierbar ist

mit Gewichten $c_k \in \mathbb{R}$.

Es gibt viele mögliche Basisfunktionen, siehe z. B. [46, 74]. Im Zuge dieser Arbeit betrachten wir hierfür die B-Spline Interpolation vom Grad $n = 1, 3$ (linear und kubisch), deren Stärken im Verlauf sichtbar werden und die typischerweise in der Registrierung genutzt werden. Die entsprechenden Basisfunktionen ergeben sich aus noch zu definierenden Translierten von

$$b^1(x) = \begin{cases} x & x \in [0, 1) \\ 2 - x & x \in [1, 2) \\ 0 & \text{sonst} \end{cases}$$

für den linearen Fall ($n = 1$) und durch

$$b^3(x) = \frac{1}{6} \begin{cases} x^3, & [0, 1) \\ 1 + 3(x - 1) + 3(x - 1)^2 - 3(x - 1)^3, & [1, 2) \\ 4 - (x - 2)^2 + 3(x - 2)^3, & [2, 3) \\ 1 - 3(x - 3) + 3(x - 3)^2 - (x - 3)^3, & [3, 4) \\ 0, & \text{sonst.} \end{cases}$$

für die kubische ($n = 3$) B-Spline Interpolation. Die beiden B-Splines sind in Abbildung 3.7 zu betrachten. Da die vorgestellten Splines nur auf dem Intervall $[0, n + 1]$ Einfluss haben, führen wir die KARDINALEN B-SPLINES b_k^n als Translierte der B-Splines b^n ein, d. h.

$$b_{k,h}^n(x) = b^n \left(\frac{x - x^{(0)}}{h} - k + \frac{n + 1}{2} \right),$$

mit $x^{(0)}$ definiert wie in (3.4) und $k \in \mathbb{Z}$. Bei Betrachtung der Bildzeile \mathbf{t} ist $k \in \{0, m_2 - 1\}$. Mit diesen Basisfunktionen und den Interpolationsbedingungen

$$t_j = I(\bar{\mathbf{x}}_j^{cc}) = \sum_{k=0}^m c_k b_{k,h}^n(\bar{\mathbf{x}}_j^{cc}), \quad j = 0, \dots, m_2 - 1$$

kommen wir auf das lineare Gleichungssystem $A\mathbf{c} = \mathbf{t}$ mit

$$\mathbf{c} = (c_0, \dots, c_{m_2-1})^T$$

$$A = (b_{k,h}^n(\bar{\mathbf{x}}_j^{cc}))_{j,k=0}^{m_2-1}.$$

Somit sind die Koeffizienten der linearen B-Spline-Interpolation gegeben durch

$$\begin{pmatrix} 1 & & & \\ & \ddots & & \\ & & & 1 \end{pmatrix} \mathbf{c} = \mathbf{t} \quad (3.9)$$

und für die kubische B-Spline-Interpolation durch

$$\begin{pmatrix} 4 & 1 & & & \\ 1 & 4 & 1 & & \\ & \ddots & \ddots & \ddots & \\ & & 1 & 4 & 1 \\ & & & 1 & 4 \end{pmatrix} \mathbf{c} = \mathbf{y}. \quad (3.10)$$

Hier wird auch eine Stärke der B-Spline Interpolation deutlich. Die Matrix A ist symmetrisch positiv definit. Das Gleichungssystem ist eindeutig und stabil lösbar. Ein visueller Vergleich der beiden Interpolationsarten ist in Abbildung 3.8 zu finden.

Die Ableitungen der Interpolationsfunktionen sind einfach zu berechnen und werden hier nicht weiter aufgeführt. Es sei bemerkt, dass b^1 eine Sprungstelle an den Stützstellen in der ersten Ableitung aufweist. Während des Gebrauchs der ableitungsbasierten Optimierungsmethoden liegt es also nahe, auf die kubische Interpolation zurückzugreifen. Ein Vorteil der linearen B-Spline-Interpolation dagegen liegt darin, dass das min-max-Prinzip eingehalten wird, d. h. durch die Interpolation bekommen wir keine Grauwerte außerhalb des durch die Daten gegebenen Intervalls. Solange wir also keine Ableitungen brauchen, verwenden wir die linearen B-Splines. Im Zuge der Optimierung greifen wir dagegen auf die kubischen B-Splines zurück. Da wir in dieser Arbeit Bilder der Dimension $d = 2, 3$ betrachten, kommen wir nun auf die höherdimensionale B-Spline-Interpolation.

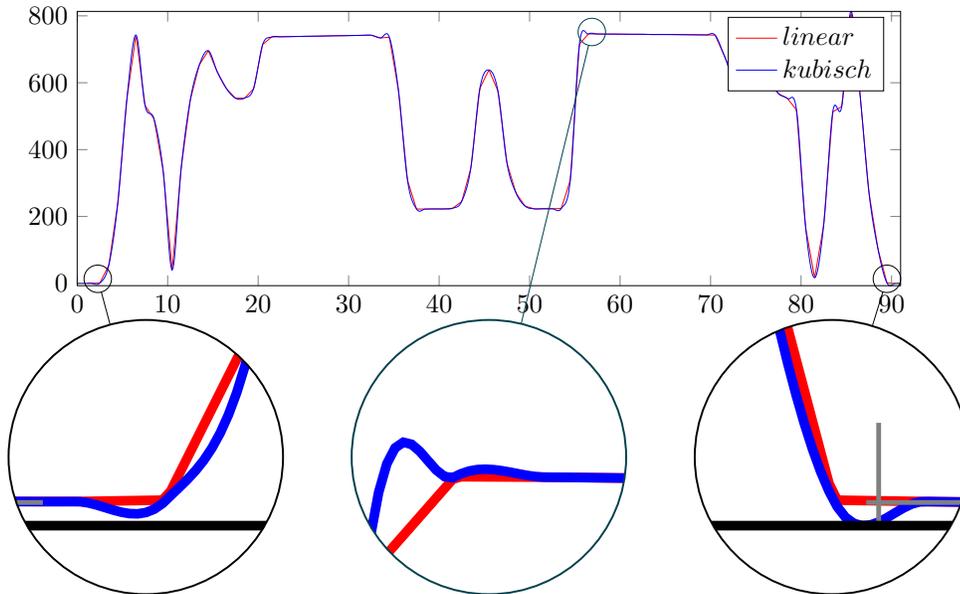


Abbildung 3.8.: Vergleich der linearen und kubischen Interpolation im eindimensionalen Fall auf dem Intervall $\Omega = [0, 10]$. Während die kubische Interpolation (grün) eine glatte, differenzierbare Lösung bereitstellt hat die lineare Interpolation den Vorteil, dass das min-max-Prinzip eingehalten wird und konstante Gebiete auch nach der Interpolation konstant bleiben. Die kubische Interpolation auf dem Intervall $[6, 8]$ weist leichte Überschwinger auf und wird im Intervall $[4, 5]$ negativ, was z. B. bei detektierter Strahlung nicht auftreten sollte.

Höherdimensionale B-Spline Interpolation Die Erweiterung der eindimensionalen Interpolation auf den d -dimensionalen Fall kann auf verschiedene Art und Weise erfolgen. Wir wählen den Tensor-Ansatz und nehmen dabei an, dass wir in alle Richtungen dieselben Basisfunktionen nutzen. Dadurch ist die höherdimensionale Interpolation einfach auf die eindimensionale zurückzuführen. Der d -dimensionale kardinale B-Spline $b_{\mathbf{k},h}^n : \mathbb{R}^d \rightarrow \mathbb{R}$, $\mathbf{k} = (k_1, \dots, k_d)$, $h = (1, \dots, h_d)$ ist gegeben durch

$$b_{\mathbf{k},h}^n(\mathbf{x}) = \prod_{i=1}^d b_{k_i, h_i}^n(x_i). \quad (3.11)$$

Wie im eindimensionalen Fall gilt der Zusammenhang zwischen der Bildgröße m und \mathbf{k} durch $k_i \in \{0, m_i - 1\}$. Die partiellen Ableitungen lassen sich analog zum

eindimensionalen Fall berechnen. Auch die Koeffizienten werden im mehrdimensionalen Fall analog zum eindimensionalen berechnet. Um dies zu verdeutlichen, betrachten wir den zweidimensionalen Fall beispielhaft.

Sei das diskrete Bild $\mathbf{T} \in \mathbb{G}^{m_1 \times m_2}$ im Urbildbereich $\Omega = (0, m_1) \times (0, m_2)$ auf einem zellzentrierten Gitter $X^{cc}(\Omega, m)$ mit $h = (h_1, h_2) = (1, 1)$ gegeben. Für die Interpolationsbedingungen gilt dann

$$\mathbf{T}_{ij} = \mathcal{T}(x_i, y_j) = \sum_{k=0}^{m_1-1} \underbrace{\sum_{l=0}^{m_2-1} c_{kl} b_{l,h}^n(x_i)}_{=: c_{il}} b_{k,h}^n(y_j) = \sum_{k=0}^{m_1} c_{il} b_k^n(y_j), \quad (3.12)$$

mit $x_i = \bar{\mathbf{x}}_{j(m_1-1)+i}^{cc}$ und $y_j = \bar{\mathbf{x}}_{(m_1 m_2)+j(m_1-1)+i}^{cc}$. Wir bekommen also m_1 eindimensionale Interpolationsprobleme in einem ersten Schritt und weitere m_2 für jeweils fixiertes $l \in \{0, \dots, m_2 - 1\}$ in einem zweiten Schritt. Gleiches Schema ist auf den Fall $d = 3$ zu übertragen. Der Übersicht halber betrachten wir es hier nicht weiter. Als Essenz nehmen wir mit, dass wir mit den Methoden der eindimensionalen B-Spline-Interpolation auch die höherdimensionalen Interpolationen lösen können.

Zur späteren Hilfe bei der Notation führen wir den INTERPOLATIONSOPERATOR $\text{inter} : \mathbb{R}^n \times \mathbb{R}^{dm} \times \{1, 3\} \rightarrow \mathbb{R}^m$ ein, mit der Dimension d der Daten. Aus dem Kontext gehen die korrespondierenden Koordinaten zu den Daten hervor (z. B. nodales oder zellzentriertes Gitter).

Seien die Daten \mathbf{T} , durch die die Interpolante berechnet wird, sowie die Koordinaten Y , an denen die kontinuierliche Repräsentante ausgewertet werden soll, gegeben. Wir schreiben dann

$$[\mathbf{T}_Y, d\mathbf{T}_Y] = \text{inter}(\mathbf{T}, Y, 1) \quad \text{und} \quad (3.13)$$

$$[\mathbf{T}_Y, d\mathbf{T}_Y] = \text{inter}(\mathbf{T}, Y, 3) \quad (3.14)$$

für die Auswertung der linearen respektive kubischen Interpolation der Daten $\mathbf{T} \in \mathbb{R}^m$ und deren DISKRETE ABLEITUNG $d\mathbf{T} \in \mathbb{R}^{m \times dm}$ an den Punkten Y . Die Daten $\mathbf{T} \in \mathbb{R}^{m_1 m_2}$ können beispielsweise ein diskretes Bild sein, die Koordinaten $Y \in \mathbb{R}^{2m_1 m_2}$ ein rotiertes zellzentriertes Gitter, wie bereits oben beschrieben. Eine Möglichkeit der Berechnung der diskreten Ableitung werden wir in Abschnitt 3.3.2 betrachten.

Bisher haben wir uns keine Gedanken gemacht, wie Punkte am Rand des Gebietes Ω behandelt werden. Hier fließen Informationen ein, die außerhalb von Ω liegen. Hierzu werden Randbedingungen gestellt [94]. Im Zuge dieser Arbeit verwenden wir Dirichlet-Null-Randbedingungen, d. h. das Bild wird am Rand gleich 0 gesetzt. Dies

bietet sich bei den vorliegenden diskreten Bildern an, da diese nach einer Vorverarbeitung am Rand typischerweise den Wert 0 aufweisen. Zudem erhalten wir durch die Null-Randbedingungen und den kompakten Träger der Basisfunktionen einen kompakten Träger für unsere Bilder, wodurch später z. B. auftretende Integrale wohldefiniert sind.

3.3. Numerische Integration und Differentiation

Wie wir im Folgenden sehen werden, enthalten die vorgestellten kontinuierlichen Distanzmaße, Regularisierer und Nebenbedingungen Integral- und Differentialoperatoren. Da wir in dem diskretisieren-dann-optimieren Rahmen arbeiten, müssen diese Operatoren diskretisiert werden. Daher führen wir ein kurzes Resümee über numerische Integration und Differentiation. Für weitere Hinweise können z. B. [74, 46] herangezogen werden.

3.3.1. Numerische Integration

Wir haben im vorangegangenen Abschnitt das nodale und zellzentrierte Gitter kennengelernt, auf denen unsere Daten gegeben sein können. Je nachdem welches der Gitter verwendet wird, bietet sich eine Regel zur numerischen Integration an. Wir werden daher zwei Methoden vorstellen, eine für zellzentrierte und eine für nodale Gitter. Dazu starten wir im eindimensionalen Fall und erläutern zum Abschluss die Erweiterung auf den mehrdimensionalen Fall. Die grundlegende Idee ist dabei, die betrachtete Funktion durch ein Polynom eines gewissen Grades $n \in \mathbb{N}$ anzunähern. Das Integral des Polynoms ist dann leicht zu berechnen. Wir betrachten die Fälle $n = 0, 1$.

Starten wir mit dem Fall $n = 0$ und einem zellzentrierten Gitter $X^{cc}(\Omega, m)$ auf dem die kontinuierliche Funktion \mathcal{T} ausgewertet ist. Betrachten wir eine Zeile des diskreten Bildes $\mathbf{T} \in \mathbb{G}^{m_1, m_2}$. Dann erhalten wir für die Approximation des analytischen Integrals

$$\int_{\omega_1}^{\omega_2} \mathcal{T}(x) \, dx = h \sum_{x \in X^{cc}} \mathcal{T}(x) + \mathcal{O}(h^2).$$

Diese Approximation wird ZUSAMMENGESETZTE RECHTECKREGEL genannt.

Betrachten wir hingegen ein nodales Gitter $X^n(\Omega, m)$, bietet sich die ZUSAMMENGESETZTE TRAPEZREGEL

$$\int_{\omega_1}^{\omega_2} \mathcal{T}(x) \, dx = h \left(\frac{\mathcal{T}(\omega_1)}{2} + \sum_{x \in X^n \setminus \{\omega_1, \omega_2\}} \mathcal{T}(x) + \frac{\mathcal{T}(\omega_2)}{2} \right) + \mathcal{O}(h^2)$$

an. Da beide Approximationen dieselbe Güte ($\mathcal{O}(h^2)$) aufweisen, hängt die Wahl wirklich nur von dem gegebenen Gitter ab.

Die Erweiterung in den d -dimensionalen Fall ist unkompliziert. Wir betrachten ihn am Beispiel der Rechteckregel für den zweidimensionalen Fall. Die weiteren Fälle verlaufen analog. Seien Daten $\mathbf{T} \in \mathbb{R}^{m_1 \times m_2}$ der kontinuierlichen Funktion $\mathcal{T} : \mathbb{R}^2 \rightarrow \mathbb{R}$ auf einem zellzentrierten Gitter $X^{cc}(\Omega, m)$ gegeben. Betrachten wir das Integral

$$\int_{\Omega} \mathcal{T}(\mathbf{x}) \, d\mathbf{x} = \int_{\omega_3}^{\omega_4} \int_{\omega_1}^{\omega_2} \mathcal{T}(x_1, x_2) \, dx_1 \, dx_2.$$

Bei einer Fixierung von x_2 führt die Rechteckregel zu

$$\int_{\Omega} \mathcal{T}(\mathbf{x}) \, d\mathbf{x} = \int_{\omega_3}^{\omega_4} h_1 \sum_{i_1=0}^{m_1-1} \mathcal{T}(x_1^{(0)} + i_1 h_1, x_2) \, dx_2 + \mathcal{O}(h_1^2).$$

Aufgrund der Linearität des Integraloperators können wir für hinreichend glatte \mathcal{T} die Reihenfolge der Integration und Summation vertauschen, lösen die Fixierung von x_2 und wenden die Rechteckregel erneut an. Wir enden bei

$$\int_{\Omega} \mathcal{T}(\mathbf{x}) \, d\mathbf{x} = \bar{h} \sum_{i_2=0}^{m_2-1} \sum_{i_1=1}^{m_1-1} \mathcal{T}(x_1^{(0)} + i_1 h_1, x_2^{(0)} + i_2 h_2) + \mathcal{O}(\max\{h_1^2, h_2^2\}). \quad (3.15)$$

3.3.2. Numerische Differentiation

Um die Ableitungen approximieren zu können, die später in unseren Zielfunktionalen auftauchen, verwenden wir finite Differenzen. Wie bei der Integration ist dabei die Wahl der Regel abhängig von dem jeweiligen Gitter, auf dem unsere Daten gegeben sind und auf welchem Gitter die Ableitungen gegeben sein sollen. Da der höherdimensionale Fall direkt mit dem eindimensionalen Fall einhergeht, wenden wir uns diesem direkt zu. Im Zuge dieser Arbeit betrachten wir für hinreichend glatte $f : \mathbb{R}^n \rightarrow \mathbb{R}$ den ZENTRALEN DIFFERENZENQUOTIENTEN

$$\frac{\partial}{\partial x_i} f(\mathbf{x}) = \frac{f(\mathbf{x} + h_i \mathbf{e}_i) - f(\mathbf{x} - h_i \mathbf{e}_i)}{h_i} + \mathcal{O}(h_i^2) \quad (3.16)$$

für die Approximation der ersten Ableitung in Richtung der Einheitsvektoren \mathbf{e}_i , $i \in \{1, \dots, d\}$.

Eine Approximation der zweiten Ableitung in Richtung \mathbf{e}_i , $i \in \{1, \dots, d\}$ ist gegeben durch

$$\frac{\partial^2}{\partial^2 x_i} f(\mathbf{x}) = \frac{f(\mathbf{x} + h_i \mathbf{e}_i) - 2f(\mathbf{x}) + f(\mathbf{x} - h_i \mathbf{e}_i)}{h_i^2} + \mathcal{O}(h_i^2). \quad (3.17)$$

3.4. Transformationen

Durch die Interpolation haben wir ein Mittel kennengelernt, diskret gegebene Bilder kontinuierlich darzustellen. Dies ermöglicht uns eine Transformationen φ einfach auf Bilder anzuwenden. Wegen der besseren Übersicht betrachten wir den zweidimensionalen Fall.

Wir schreiben für das durch die TRANSFORMATION $\varphi : \mathbb{R}^2 \rightarrow \mathbb{R}^2$ DEFORMIERTE BILD \mathcal{T} ,

$$\mathcal{T}_\varphi := \mathcal{T} \circ \varphi, \quad \mathcal{T}_\varphi(\mathbf{x}) = \mathcal{T}(\varphi(\mathbf{x})). \quad (3.18)$$

Die Transformation wird in der Bildregistrierung auch DEFORMATION genannt.

Häufig wird $\varphi = id + \mathbf{u}$ geschrieben, wobei $\mathbf{u} : \mathbb{R}^2 \rightarrow \mathbb{R}^2$ auch VERRÜCKUNGSFUNKTION genannt wird. Wir nehmen an, dass die Transformation invertierbar ist und somit $\varphi^{-1} : \mathbb{R}^2 \rightarrow \mathbb{R}^2$ existiert. Wir können bei Transformationen zwischen zwei Betrachtungsweisen unterscheiden, dem Euler- und dem Lagrange-Ansatz (siehe z.B. [38]). Im Folgenden sei $\mathbf{x} \in \Omega$ und $\varphi(\mathbf{x}) = \bar{\mathbf{x}}$.

Beim EULER-ANSATZ betrachten wir, woher ein fester Punkt $\bar{\mathbf{x}}$ im deformierten Bild kommt. Da dieser Punkt $\mathbf{x} = \varphi^{-1}(\bar{\mathbf{x}})$ ist, betrachten wir folglich am Punkt $\bar{\mathbf{x}}$ den Grauwert $\mathcal{T}(\varphi^{-1}(\bar{\mathbf{x}})) = \mathcal{T}(\mathbf{x})$. Das deformierte Bild kann also auf einem vorgegebenen Gitter direkt interpoliert werden. Mit dieser Betrachtungsweise einher geht ein kontraintuitives Verhalten. Beschreibt nun die Transformation beispielsweise eine Ausdehnung unseres Urbildbereiches Ω und betrachten wir die Punkte $\bar{\mathbf{x}}$ im deformierten Bild. Die Grauwerte werden an den Positionen $\varphi^{-1}(\bar{\mathbf{x}})$ des Ursprungbildes abgegriffen. Da $\varphi(\varphi^{-1}(\mathbf{x})) = \mathbf{x}$ gilt, ist es einleuchtend dass die φ^{-1} ein schrumpfendes Verhalten aufweist. Die Objekte werden kleiner anstatt größer dargestellt.

Der LAGRANGE-ANSATZ beschreibt dagegen, wohin der Punkt \mathbf{x} geht. Wir finden also den Grauwert $\mathcal{T}(\mathbf{x})$ an der Stelle $\bar{\mathbf{x}} = \varphi(\mathbf{x})$ wieder. Im Gegensatz zum Euler-Ansatz tritt hierbei das erwartete Verhalten ein, da wir die Punkte direkt verfolgen,

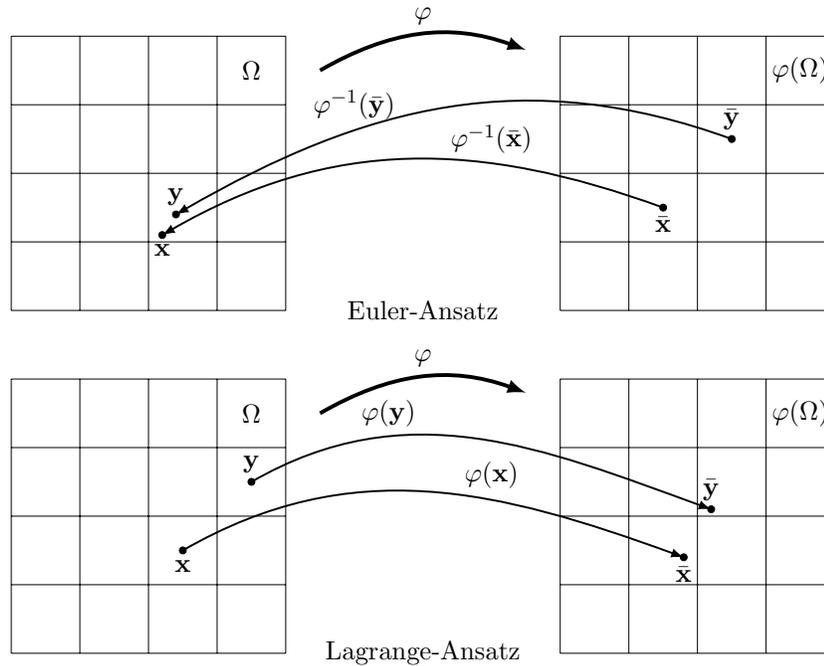


Abbildung 3.9.: Unterschied des Euler- (oben) und Lagrange-Ansatz (unten). Beim Euler-Ansatz betrachten wir einen Punkt $\bar{\mathbf{x}}$ im deformierten Bild (rechts) und gucken, woher dieser Punkt kommt. Hier ist der Einsatz der Inversen φ^{-1} notwendig. Im Lagrange-Ansatz gucken wir hingegen direkt, wohin ein Punkt \mathbf{x} sich bewegt.

allerdings erfolgt die Interpolation nicht so einfach wie beim Euler-Ansatz, da wir im deformierten Bild kein regelmäßiges Gitter mehr betrachten.

Abbildung 3.9 veranschaulicht diese beiden Ansätze. Im Zuge dieser Arbeit werden wir den Euler-Ansatz verfolgen. Dies begründet sich auf den Vorteilen in der Berechnung des transformierten Bildes. So können wir zur Auswertung unseres deformierten Bildes auf einem zellzentrierten Gitter mittels Interpolation direkt, wie oben bereits erwähnt, die entsprechenden Grauwerte des Ursprungsbildes ansprechen. Wie wir später feststellen werden, müssen wir dies bei der Formulierung der Nebenbedingungen im Auge behalten. Für eine nähere Beleuchtung dieses Themas verweisen wir auf [38].

Kann die Transformation nicht durch eine endliche Anzahl an Parametern beschrieben werden, sprechen wir von einer NICHTPARAMETRISCHEN TRANSFORMATION.

3. Bildregistrierung

Wird die Transformation allerdings von einer endlichen Menge von $p \in \mathbb{N}$ Parametern $\boldsymbol{\gamma} = (\gamma_1, \dots, \gamma_p)^T \in \mathbb{R}^p$ beschrieben, sprechen wir von PARAMETRISCHEN TRANSFORMATIONEN. Wir schreiben dann für die Transformation

$$\varphi(\mathbf{x}) = Q(\mathbf{x})\boldsymbol{\gamma}, \quad (3.19)$$

wobei $Q : \mathbb{R}^2 \rightarrow \mathbb{R}^{2 \times p}$ eine Menge von p Basisfunktionen an der Stelle \mathbf{x} auswertet, siehe hierzu auch z. B. [76]. Häufig soll in einem ersten Schritt der Registrierung eine grobe und vertrauenswürdige Ausrichtung der beiden Bilder erfolgen. Im Rahmen dieser Arbeit sind daher für uns zwei parametrische Transformationen von Interesse, die rigiden und affin-linearen Transformationen. Diese behandeln alle Punkte gleich und erhalten somit die Strukturen der im Bild enthaltenen Objekte. Wir betrachten hierbei den zweidimensionalen Fall, eine Erweiterung in den dreidimensionalen Fall verläuft analog und kann z.B. in [76] nachvollzogen werden.

Rigide Transformationen Eine RIGIDE TRANSFORMATION ermöglicht Rotation und Translation, d. h. $\boldsymbol{\gamma} \in \mathbb{R}^3$. Sie kann beschrieben werden durch

$$\varphi(\mathbf{x}) = \mathbf{y} = \begin{pmatrix} y_1 \\ y_2 \end{pmatrix} = \begin{pmatrix} \cos \gamma_1 & -\sin \gamma_1 \\ \sin \gamma_1 & \cos \gamma_1 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} + \begin{pmatrix} \gamma_2 \\ \gamma_3 \end{pmatrix}, \quad (3.20)$$

wobei γ_1 der Rotationsparameter ist und $(\gamma_2, \gamma_3)^T$ die Translationsparameter enthält.

Die Transformation ist allerdings nichtlinear in $\boldsymbol{\gamma}$. Eine lineare Darstellung erhalten wir durch

$$Q(\mathbf{x})\mathbf{q} = \begin{pmatrix} x_1 & x_2 & 1 & 0 & 0 & 0 \\ & & & x_1 & x_2 & 1 \end{pmatrix} \mathbf{q}(\boldsymbol{\gamma}), \quad (3.21)$$

mit $\mathbf{q}(\boldsymbol{\gamma}) = (\cos \gamma_1, -\sin \gamma_1, \gamma_2, \sin \gamma_1, \cos \gamma_1, \gamma_3)^T \in \mathbb{R}^6$. Für eine detailliertere Ausführung verweisen wir auf [76].

Rigide Transformationen sind volumen-, längen- und winkelerhaltend. Es wird lediglich eine Reorientierung eines starren Objektes durchgeführt.

Manchmal ist die Betrachtung einer Kombination aus Rotation und Translation nicht ausreichend, um eine gute Vorregistrierung, d. h. erste grobe Ausrichtung zweier Bilder, zu ermöglichen. Betrachten wir beispielsweise MR-Aufnahmen eines Gehirns von Patient A und Patient B, die sich stark in ihrer Größe unterscheiden. In diesem Fall macht es Sinn, zusätzlich z. B. Skalierungen zu erlauben.

Affin-Lineare Transformationen Eine AFFIN-LINEARE TRANSFORMATION kann durch die Parameter $\gamma \in \mathbb{R}^6$ mit

$$\varphi(\mathbf{x}) = \mathbf{y} = \begin{pmatrix} y_1 \\ y_2 \end{pmatrix} = \begin{pmatrix} \gamma_1 & \gamma_2 \\ \gamma_4 & \gamma_5 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} + \begin{pmatrix} \gamma_3 \\ \gamma_6 \end{pmatrix}, \quad (3.22)$$

beschrieben werden. Wir erhalten die Darstellung

$$Q(\mathbf{x})\gamma = \begin{pmatrix} x_1 & x_2 & 1 & 0 & 0 & 0 \\ & & & x_1 & x_2 & 1 \end{pmatrix} \gamma, \quad (3.23)$$

die linear in γ ist. Mit affin-linearen Transformationen ist es möglich, Rotationen, Translationen, Skalierungen und Scherungen zu beschreiben.

3.4.1. Eigenschaften

In der Bildregistrierung sind wir häufig an Eigenschaften von Transformationen wie der Invertierbarkeit interessiert. Mit den rigiden und affin-linearen Transformationen haben wir globale Transformationen, d. h. Transformationen, die sich für alle Punkte $\mathbf{x} \in \mathbb{R}^2$ gleich verhalten, kennengelernt. Eigenschaften dieser Transformationen sind dadurch auch global. Rigide Transformationen sind zudem invertierbar, bei affin-linearen Transformationen ist dies abhängig von der Matrix bestehend aus den Parametern $\gamma_1, \gamma_2, \gamma_4, \gamma_5$. Diese Fälle sollen uns darum nicht weiter interessieren. Betrachten wir nun eine beliebige nichtparametrische Transformation φ .

Viele wichtige Informationen über die Transformationen stecken in der FUNKTIONALDETERMINANTE

$$\det(D\varphi(\mathbf{x})), \quad (3.24)$$

mit der Jakobi-Matrix der Transformation $D\varphi: \mathbb{R}^2 \rightarrow \mathbb{R}^2$.

Eine wichtige Eigenschaft der Transformation ist ihre Invertierbarkeit. Ein Satz aus der Analysis besagt, dass wenn für $\mathbf{x} \in \Omega$ gilt $\det(D\varphi(\mathbf{x})) \neq 0$, dann ist die Deformation φ in einer Umgebung von \mathbf{x} invertierbar [30]. Existiert ein Punkt $\mathbf{x} \in \Omega$, für den $\det(D\varphi(\mathbf{x})) = 0$ gilt, sprechen wir davon, dass die Transformation eine FALTUNG aufweist. Dieser Ausdruck begründet sich darauf, dass bei einer Visualisierung der Deformation über die Deformation eines diskreten Gitters eine Faltung auftritt, siehe hier z. B. Abbildung 3.10 oder Abbildung 3.12. Wir sprechen bei dieser Visualisierung vom DEFORMATIONSFELD. Kommt es bei der Transformation zu einer lokalen Umorientierung von Punkten, d. h. die Eckpunkte einer Zellen tauschen ihre Positionen, weisen die durch das Gitter aufgespannten Zellen eine Faltung auf. Gilt $\det(D\varphi(\mathbf{x})) > 0$ für alle $\mathbf{x} \in \Omega$, sprechen wir von einer FALTUNGSFREIEN TRANSFORMATION.

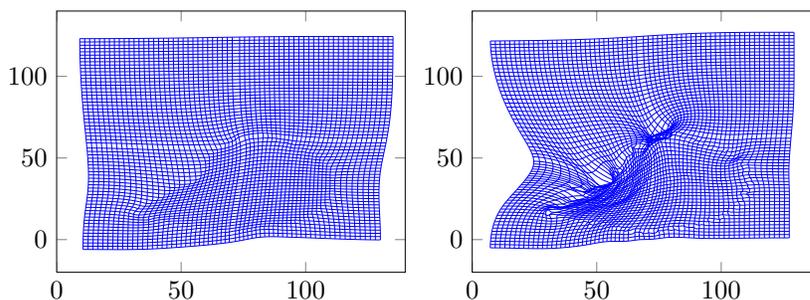


Abbildung 3.10.: Wir sehen die Darstellung einer faltungsfreien (links) Transformation und einer Transformation mit Faltung (rechts). Faltungen resultieren aus einer Umorientierung einzelner Gitterpunkte. Die Transformation ist nicht invertierbar.

Eine weitere wichtige Eigenschaft, die in der Funktionaldeterminante steckt, ist die **VOLUMENÄNDERUNG**. Wir bezeichnen dabei

$$\text{vol}(\Omega) := \int_{\Omega} d\mathbf{x} \quad (3.25)$$

als das **VOLUMEN** des Gebiets Ω , unabhängig von der Dimension d . Nach dem Transformationssatz für Integrale [31] gilt für eine kompakte Menge Ω

$$\text{vol}(\Omega) = \int_{\Omega} |\det(D\varphi(\mathbf{x}))| d\mathbf{x} \quad (3.26)$$

für das deformierte Gebiet $\varphi(\Omega)$, wenn φ bijektiv und sowohl φ als auch φ^{-1} stetig differenzierbar sind. Wir erkennen sofort dass

$$|\det(D\varphi(\mathbf{x}))| = 1 \quad \forall \mathbf{x} \in \Omega \Rightarrow \text{vol}(\Omega) = \text{vol}(\varphi(\Omega)),$$

d. h. das Volumen bleibt nach der Transformation erhalten. Ebenso ergibt sich, dass für $|\det(D\varphi(\mathbf{x}))| > 1$ in einer Umgebung von \mathbf{x} eine Vergrößerung in der Umgebung von \mathbf{x} auftritt. Analog beobachten wir eine Verkleinerung in der Umgebung von \mathbf{x} , wenn $|\det(D\varphi(\mathbf{x}))| < 1$ in einer Umgebung von \mathbf{x} ist.

3.4.2. Diskretisierung

Da wir in einem diskretisieren-dann-optimieren Rahmen arbeiten, benötigen wir auch die Diskretisierungen der Transformation und Funktionaldeterminante. Starten wir mit der Transformation.

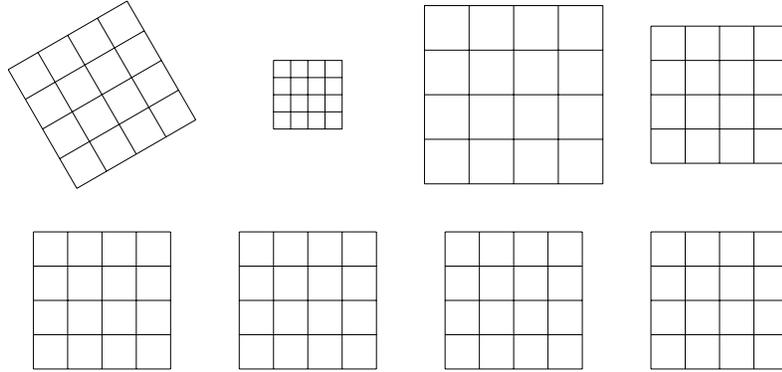


Abbildung 3.11.: Wir sehen 4 verschiedene lineare Transformationen. Von links nach rechts wird eine Rotation, eine Skalierung um den Faktor 0.5 beider Komponenten, eine Skalierung um den Faktor 1.3 beider Komponenten und eine Spiegelung an der x-Achse gezeigt. Die Funktionaldeterminante hat die Werte 1, 0.5, 1.3 bzw. -1 .

Seien Ω, m gegeben. Zur Diskretisierung der Transformation wählen wir ein nodales Gitter $X^n(\Omega, m)$, da uns dieses bei der Betrachtung der Funktionaldeterminante entgegenkommt. Die Diskretisierung der Transformation auf einem zellzentrierten Gitter $X^{cc}(\Omega, m)$ verläuft analog. Wir zeigen beispielhaft den zweidimensionalen Fall. Sei dazu $\varphi(\mathbf{x}) = (\varphi^1(\mathbf{x}), \varphi^2(\mathbf{x}))^T$. Die diskrete Transformation φ ist dann gegeben durch

$$\varphi = \begin{pmatrix} \varphi^1 \\ \varphi^2 \end{pmatrix} \in \mathbb{R}^{2(m_1+1)(m_2+1)} \quad (3.27)$$

mit

$$\varphi_i^1 = \varphi^1(\bar{\mathbf{x}}_i^n, \bar{\mathbf{x}}_{i+s}^n) \quad \varphi_i^2 = \varphi^2(\bar{\mathbf{x}}_i^n, \bar{\mathbf{x}}_{i+s}^n), \quad (3.28)$$

wobei $s = (m_1 + 1)(m_2 + 1)$. Die diskrete Transformation steht also wie ein Gitter (Beispiel, Gleichung (3.7)) in einem Vektor.

Bei der Diskretisierung der Funktionaldeterminante wählen wir anstatt einer Approximation der Jakobi-Matrix und deren partiellen Ableitungen einen anderen Zugang. Wie wir gesehen haben, erhalten wir durch die Funktionaldeterminante die Volumenänderung eines Gebietes V_i . Für dieses Gebiet V_i wählen wir das Gebiet, welches durch einen Voxel beschrieben wird, d. h.

$$\text{vol}(\varphi(V_i)) = \int_{V_i} \det(D\varphi(\mathbf{x})) \, d\mathbf{x}. \quad (3.29)$$

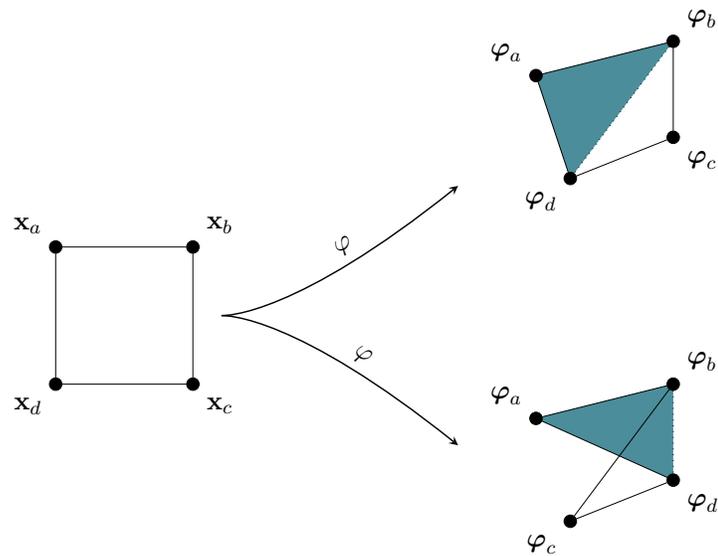


Abbildung 3.12.: Diskretisierung der Funktionaldeterminante über Volumenbestimmung eines Voxels. Das Kreuzprodukt der Diagonalen berechnet das Volumen des Voxels. Dieses ist eine gute Methode für glatte Transformationen (oben). Treten allerdings Faltungen in der Transformation auf, kann bei dieser Berechnungsweise trotzdem das Volumen überall größer 0 sein. Abhilfe schafft hier die Unterteilung des Voxels in Dreiecke. Entsteht eine Faltung, hat eines der Dreiecke ein negatives Volumen. Unten rechts sehen wir gut eine Veranschaulichung des Begriffs Faltung

Dies ist für unsere Berechnungen die kleinste mögliche Einheit und ist dadurch eine plausible Wahl. Wir betrachten der Übersicht halber wieder den zweidimensionalen Fall. Der deformierte Voxel $\varphi(V_i)$ sei gegeben über die Eckpunkte $\varphi_a = \varphi(\mathbf{x}_a)$, $\varphi_b = \varphi(\mathbf{x}_b)$, $\varphi_c = \varphi(\mathbf{x}_c)$ und $\varphi_d = \varphi(\mathbf{x}_d)$ wie dargestellt in Abbildung 3.12. Über das Kreuzprodukt der beiden Diagonalen des Voxels, d.h. $(\varphi_c - \varphi_a) \times (\varphi_b - \varphi_d)$ bestimmen wir das Volumen der Zelle bei einer Approximationsgüte von h^2 [40] und wir erhalten die Diskretisierung

$$v_i := \text{vol}(\varphi(V_i)) \approx \det((\varphi_c - \varphi_a), (\varphi_b - \varphi_d)). \quad (3.30)$$

Bei der Berechnung des Volumens des Viereckes über das Kreuzprodukt besteht die Möglichkeit, dass Faltungen nicht erkannt werden. Dies können wir umgehen, indem wir statt des kompletten Viereckes, Dreiecke betrachten. Genauer gesagt berechnen wir die Volumen der Dreiecke, aufgespannt durch die Punkte mit den Subskripten $\{a, b, d\}$ und $\{b, c, d\}$. Kommt es zu einer Faltung, weist eines der Dreiecke ein negatives Volumen auf. Mit

$$V_{abd} := \det((\varphi_d - \varphi_a), (\varphi_b - \varphi_a)). \quad (3.31)$$

wird die Fläche des durch $\varphi_a, \varphi_b, \varphi_d$ aufgespannten Parallelogramms berechnet. Es ergibt sich das Volumen der Zelle und damit die Diskretisierung der Funktionaldeterminante mittels Zerlegung in Dreiecke durch

$$\text{vol}(\varphi(V_i)) \approx \frac{1}{2} (V_{abc} + V_{bcd}). \quad (3.32)$$

Damit haben wir mittels der Diskretisierung sowohl Werkzeuge zur Verfolgung der Volumenänderung als auch der Überprüfung auf Faltungsfreiheit der Transformation. Eine detaillierte Ausführung ist in [24, 56] zu finden.

Für den dreidimensionalen Fall werden statt der Dreiecke Tetraeder betrachtet. Dabei gibt es unterschiedliche Zerlegungen des Einheitswürfels in fünf, sechs oder z. B. 24 Tetraeder. Bei letzterem kann eine Richtungsunabhängigkeit bei Verfahren gezeigt werden, bei denen das Volumen in die Optimierung mit einfließt. Richtungsunabhängigkeit in diesem Zusammenhang bedeutet, dass Änderungen des Volumens sich in alle Richtungen gleich auswirken. Für eine nähere Betrachtung dieser Fälle sei auf [39, 41, 56, 33] verwiesen.

Die Wahl der Diskretisierung hängt von der jeweiligen Anwendung ab. Bei der volumenerhaltenden Registrierung (z. B. [41]) hängt diese unter anderem davon ab, ob die Volumina als Gleichheits- oder Ungleichheitsnebenbedingungen in das Modell einfließen. Sind beide Diskretisierungen anwendbar, rücken rechentechnische

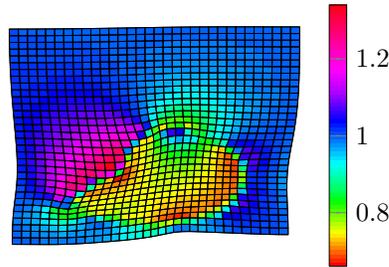


Abbildung 3.13.: Visualisierung der Volumenänderung über eine farbliche Kodierung der diskreten Zellvolumina. Wir sehen ein Beispiel für eine nichtlineare, stark volumenändernde Transformation. Wir können starke Volumenänderungen durch die Farbänderungen sofort feststellen. Eine Faltung tritt dabei nicht auf. Es sind keine negativen Werte präsent.

Aspekte in den Blickwinkel. Dabei weist die Diskretisierung über das Kreuzprodukt den großen Vorteil auf, dass für ein Voxel nur eine Determinantenberechnung erfolgt.

Eine beispielhafte Visualisierung der Volumenänderung ist Abbildung 3.13 zu entnehmen. Hierbei sind die Änderungen der Zellvolumina farblich kodiert.

3.5. Distanzmaße

Laut Problemformulierung gilt es, eine Transformation zu finden, die korrespondierende Punkte möglichst gut aufeinander abbildet. In diesem Abschnitt wollen wir uns einen Einblick in die Formulierung "möglichst gut" verschaffen.

Eine möglichst gute Abbildung korrespondierender Punkte führt zu einer hohen Ähnlichkeit von Bildern. Das Ziel eines Distanzmaßes ist es, die (Un-)Ähnlichkeit zweier Bilder \mathcal{T} und \mathcal{R} zu messen. In unserem Rahmen, in dem wir ein Zielfunktional minimieren, sollte die Distanz bei hoher Ähnlichkeit klein bzw. bei großer Unähnlichkeit groß sein. Aus Optimierungssicht sind dabei einige Punkte zu beachten. Zum einen sollte das Minimum des gewählten Distanzmaßes die gesuchte Lösung möglichst gut beschreiben, d. h. die erwartete Lösung sollte durch ein Minimum der Distanz charakterisiert sein. Zum anderen sollte das gewählte Distanzmaß differenzierbar sein, damit wir ableitungsbasierte Optimierungsverfahren (siehe Abschnitt 2) anwenden können.

In der Registrierung können wir zwischen zwei Arten von Distanzmaßen unterscheiden, den FEATURE-BASIERTEN und den INTENSITÄTS-BASIERTEN Distanzmaßen.

Ein Beispiel für feature-basierte Distanzmaße ist der Abstand zwischen Landmarken. Hierbei hängt die Ähnlichkeit zweier Bilder allein von einer Menge besonderer, hervorstechender Punkte ab. Auf ein Distanzmaß dieser Klasse werden wir im Abschnitt 4.5.3 treffen und es dort genauer beleuchten. Die Anzahl der Korrespondenzen ist dabei eher gering und die Ähnlichkeit wird über den euklidischen Abstand der korrespondierenden Punkte gemessen. Bei intensitäts-basierten Distanzmaßen hängt die Ähnlichkeit dagegen von den Grauwerten der Bilder im gesamten Bereich ab. Die Verwendung des Begriffes Intensität in diesem Zusammenhang geht auf die physikalische Entstehung des Bildes zurück (vgl. Abschnitt 3.1). Im Folgenden betrachten wir diese Klasse von Distanzmaßen.

Viele verschiedene Distanzmaße können in der Literatur gefunden werden. Da die Literatur größtenteils in Englisch gehalten ist, verwenden wir ebenfalls die englischen Begriffe daraus. Das Sum-of-Squared Differences (SSD), Normalized Cross-Correlation (NCC), Mutual Information (MI) und Normalized Gradient Fields (NGF) sind einige der prominentesten Beispiele. Im Hinblick auf die späteren Anwendungen werden wir nachfolgend SSD, NGF und das von Heldmann vorgestellte Least-Squares-Differences (LSD) [47] näher behandeln. Für weitere Informationen verweisen wir auf [20, 93] für MI sowie auf z. B. [68] für NCC.

Nach einer kontinuierlichen Formulierung stellen wir jeweils kurz die Diskretisierung vor und nennen vertiefende Quellen.

Sum-of-Squared Differences - SSD Wir starten mit dem intuitivsten Distanzmaß. Prinzipiell betrachtet dieses die Grauwertunterschiede an jedem Punkt im Bildbereich Ω . In seiner linearen Variante ist es durch Horn und Schunck [52] in der Bildverarbeitung eingeführt worden.

Das kontinuierliche Funktional des SUM-OF-SQUARED-DIFFERENCES ist gegeben durch

$$\mathcal{D}^{\text{SSD}}(\mathcal{R}, \mathcal{T}; \varphi) := \frac{1}{2} \int_{\Omega} (\mathcal{T}(\varphi(\mathbf{x})) - \mathcal{R}(\mathbf{x}))^2 \, d\mathbf{x}. \quad (3.33)$$

Die Stärke des SSD liegt in der Einfachheit. Es ist sowohl leicht zu interpretieren, einfach zu implementieren als auch schnell zu berechnen. Die Vorteile bringen allerdings auch Nachteile mit sich. Sind die Bilder mit verschiedenen Modalitäten aufgenommen, können sie nicht mehr zwingend mit Hilfe ihrer Grauwerte verglichen werden. Durch die Quadrierung der Differenzen ist das Distanzmaß zudem sensitiv bezüglich starker Intensitätsunterschiede.

Wir betrachten die Diskretisierung unter der Annahme, dass die diskretisierte Transformation φ auf einem zellzentrierten Gitter gegeben ist. Die Rechteckregel (3.15)

mit $f = (\mathcal{T}_\varphi - \mathcal{R})^2$ führt zu

$$\mathbf{D}(\varphi) = \frac{\bar{h}}{2} \|\mathbf{T}_\varphi - \mathbf{R}\|_2^2.$$

Wir erhalten eine Least-Squares-Form, die wir bei der Anwendung des Gauß-Newton-Verfahrens (Abschnitt 2.1.5) voraussetzen. Mit $\mathbf{r} = \mathbf{T}_\varphi - \mathbf{R}$ und der Kettenregel erhalten wir

$$d\mathbf{D}(\varphi) = d\mathbf{r} \mathbf{r} = d\mathbf{T}_\varphi (\mathbf{T}_\varphi - \mathbf{R}) \quad (3.34)$$

für die Approximation der Ableitung. Mittels Interpolation, d. h. $[\mathbf{T}_\varphi, d\mathbf{T}_\varphi] = \text{inter}(\mathbf{T}, \varphi, 3)$, berechnen wir die dafür benötigten Werte.

Least-Squares Differences - LSD Wir betrachten in dieser Arbeit das LSD, da es zwei Vorteile bietet. Zum einen bietet es uns die Möglichkeit eine Ähnlichkeit von Bildern verschiedener Modalitäten einfach zu messen. Zum anderen erhalten wir als Beiprodukt eine Abbildung der Grauwerte der beiden Bilder. Dies ist gerade für die Anwendung aus Kapitel 3.1 interessant, bei der die Differenzen zweier MRT-Bilder betrachtet werden sollen. Im Gegensatz zum CT ist bei diesen Bildern z. B. nicht immer der gleiche Grauwert für das gleiche Gewebe gegeben. Durch die Grauwertabbildung können Differenzen, die nur durch diese unterschiedlichen Intensitäten und nicht durch Strukturen auftreten, besser betrachtet werden. Das Distanzmaß wollen wir daher im Folgenden vorstellen.

Beim SSD gehen wir davon aus, dass in den betrachteten Bildern an korrespondierenden Punkten gleiche bzw. ähnliche Grauwerte vorliegen. Abbildung 3.14 zeigt T_1 - und T_2 - gewichtete MR-Aufnahmen desselben Gehirns. Wir sehen, dass sich die Grauwerte hier deutlich unterscheiden. Das von Heldmann eingeführte LSD [47] bezieht Grauwertunterschiede zwischen korrespondierenden Punkten mit in die Berechnung ein und nutzt dazu eine Abbildung zwischen den Grauwerten der Bildern.

Sei $g : \mathbb{G} \rightarrow \mathbb{G}$ eine Abbildung des Grauwertbereiches auf sich selbst. Dann ist das LEAST-SQUARES-DIFFERENCES Distanzmaß gegeben durch

$$\begin{aligned} \mathcal{D}^{\text{LSD}}(\mathcal{R}, \mathcal{T}; \varphi) &:= \inf_{g: \mathbb{G} \rightarrow \mathbb{G}} \frac{1}{2} \int_{\Omega} (\mathcal{T}(\varphi(\mathbf{x})) - g(\mathcal{R}(\mathbf{x})))^2 dx \\ &= \inf_{g: \mathbb{G} \rightarrow \mathbb{G}} \mathcal{D}^{\text{SSD}}(g(\mathcal{R}), \mathcal{T}; \varphi) = \mathcal{D}^{\text{SSD}}(g^*(\mathcal{R}), \mathcal{T}; \varphi), \end{aligned} \quad (3.35)$$

mit der optimalen Grauwerttransformation $g^* : \mathbb{G} \rightarrow \mathbb{G}$. Das LSD ist also eine Verallgemeinerung des SSD. Durch das Einfügen der Abbildung g werden beliebige Abbildungen des Grauwertbereiches auf sich selbst nicht bestraft. Im Gegensatz

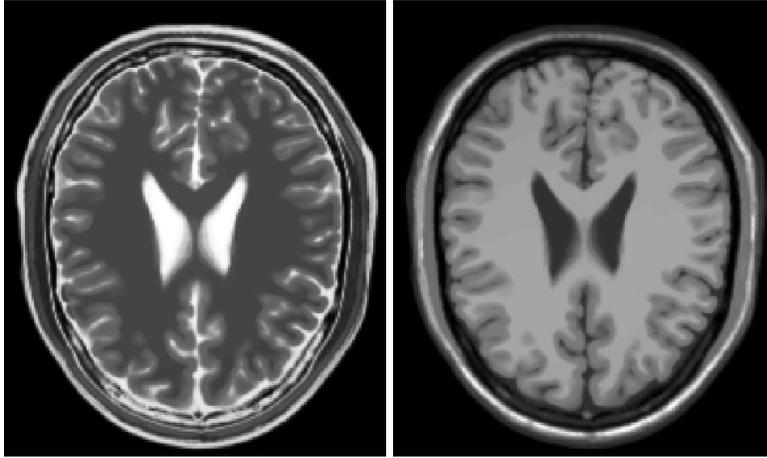


Abbildung 3.14.: Bilddaten desselben Objektes aufgenommen mit unterschiedlichen Modalitäten. Die Bilder sind Simulationen von T_2 - (links) bzw. T_1 -gewichteten (rechts) MRTs und kommen von der BrainWeb-Homepage [1]. Informationen und Referenzen zu dieser erhalten wir in [19, 57, 58, 21].

zum NCC, bei dem nur eine affin-lineare Abbildung der Grauwerte durch das Distanzmaß nicht bestraft wird [68], werden beim LSD beliebige globale Grauwertabbildungen nicht bestraft.

Auf den ersten Blick mag ein Problem des LSD darin liegen, die optimale Abbildung g^* zu berechnen. Es stellt sich jedoch heraus, dass dies explizit getan werden kann und somit eine einfache und effiziente Implementierung des LSD möglich ist.

Bei der Diskretisierung wenden wir uns zunächst der Grauwertabbildung g zu. Betrachten wir dazu das diskrete Bild $\mathbf{R} \in \mathbb{C}^n$, $n = m_1 m_2$, im Zweidimensionalen. Dieses Bild nimmt endlich viele Grauwerte $\mathbf{v} = (v_1, \dots, v_{N_g})^T$, $1 \leq N_g \leq n$, an. Mit der Matrix $\mathbf{Q} \in \{0, 1\}^{n \times N_g}$,

$$\mathbf{Q}_{ij} = \begin{cases} 1, & R_i = v_j \\ 0, & \text{sonst,} \end{cases}$$

können wir dann schreiben $\mathbf{R} = \mathbf{Q}\mathbf{v}$. Offensichtlich hat die Matrix \mathbf{Q} vollen Spaltenrang. Wenden wir nun die Grauwertabbildung g auf das Bild \mathbf{R} an, bilden wir jeden Grauwert v_i , $i = 1, \dots, N_g$ auf den Grauwert $g(v_i)$ ab. Wir erhalten dann nach der Grauwertabbildung das Bild

$$g(\mathbf{R}) = \mathbf{Q}\mathbf{g},$$

mit der diskreten Repräsentante $\mathbf{g} = (g(v_1), \dots, g(v_{N_g}))^T$ der Grauwertabbildung. Analog zum SSD erhalten wir

$$\int_{\Omega} \mathcal{T}(\varphi(\mathbf{x})) - g(\mathcal{R}(\mathbf{x})) \, d\mathbf{x} \approx \|\mathbf{T}_\varphi - \mathbf{Q}\mathbf{g}\|_2^2.$$

Zur vollständigen Diskretisierung des Terms (3.35) benötigen wir noch die optimale Grauwertabbildung g^* . Aus Gründen der Übersichtlichkeit berechnen wir hier direkt die optimale diskrete Grauwertabbildung \mathbf{g}^* als Minimierer von (3.5), d. h.

$$\min_{\mathbf{g}} \frac{1}{2} \|\mathbf{T}_\varphi - \mathbf{Q}\mathbf{g}\|_2^2. \quad (3.36)$$

Da \mathbf{Q} vollen Spaltenrang besitzt, hat das lineare Least-Squares-Problem (3.36) eine eindeutige Lösung. Mit Hilfe der Normalengleichung erhalten wir

$$\mathbf{g}^* = (\mathbf{Q}^T \mathbf{Q})^{-1} \mathbf{Q}^T \mathbf{T}_\varphi.$$

Zusätzlich hat die Matrix $\mathbf{Q}^T \mathbf{Q}$ Diagonalstruktur mit den Histogrammwerten als Einträgen und ist damit schnell zu berechnen und invertieren. Damit ergibt sich für die Diskretisierung des Distanzmaßes

$$\begin{aligned} \mathbf{D}(\varphi) &= \frac{1}{2} \|\mathbf{T}_\varphi - \mathbf{Q}\mathbf{g}^*\|_2^2 \\ &= \frac{1}{2} \left\| \mathbf{T}_\varphi - \mathbf{Q} (\mathbf{Q}^T \mathbf{Q})^{-1} \mathbf{Q}^T \mathbf{T}_\varphi \right\|_2^2 \\ &= \frac{1}{2} \left\| \left(\mathbf{I} - \mathbf{Q} (\mathbf{Q}^T \mathbf{Q})^{-1} \mathbf{Q}^T \right) \mathbf{T}_\varphi \right\|_2^2 \\ &= \frac{1}{2} \|\mathbf{P}\mathbf{T}_\varphi\|_2^2, \end{aligned}$$

mit $\mathbf{P} = \mathbf{I} - \mathbf{Q} (\mathbf{Q}^T \mathbf{Q})^{-1} \mathbf{Q}^T$. Man kann zeigen, dass $\mathbf{P}^T \mathbf{P} = \mathbf{P}$ und damit

$$\mathbf{D}(\varphi) = \frac{\bar{h}}{2} \mathbf{T}_\varphi^T \mathbf{P} \mathbf{T}_\varphi.$$

Wir lernen, dass \mathbf{g}^* nicht explizit berechnet werden muss. Bei der Implementierung macht es aus Speicherplatzgründen Sinn, die Matrix \mathbf{P} nicht explizit aufzubauen, sondern die dünnbesetzten Matrizen \mathbf{Q} und $(\mathbf{Q}^T \mathbf{Q})^{-1}$ mitzuführen und mehrere kostengünstige Matrix-Vektor-Multiplikationen durchzuführen.

Analog zum SSD erhalten wir die Ableitung mit

$$d\mathbf{D}(\varphi) = d\mathbf{T}_\varphi^T \mathbf{P} \mathbf{T}_\varphi.$$

Für eine detailliertere Ausführung verweisen wir auf [47].

Normalized Gradient Fields - NGF Wie wir bereits im Abschnitt über das LSD und in Abbildung 3.14 gesehen haben, sind Bilder nicht immer direkt über ihre Grauwerte vergleichbar. In der Abbildung sehen wir jedoch, dass Intensitätsänderungen typischerweise an den gleichen Stellen auftreten. An diesen Stellen entstehen im Bild Kanten. Das hier resümierte Distanzmaß macht sich diese Beobachtung zum Vorteil. Zwei Bilder werden demnach als ähnlich betrachtet, wenn deren Kanten sich überlappen.

Das NORMALIZED GRADIENT FIELD (NGF) ist definiert durch

$$\mathcal{D}^{\text{NGF}}(\mathcal{R}, \mathcal{T}; \varphi) = \frac{1}{2} \int_{\Omega} 1 - \left(\frac{\langle \nabla \mathcal{T}(\varphi(\mathbf{x})), \nabla \mathcal{R}(\mathbf{x}) \rangle}{\|\nabla \mathcal{T}(\varphi(\mathbf{x}))\|_2 \|\nabla \mathcal{R}(\mathbf{x})\|_2} \right)^2 dx \quad (3.37)$$

$$= \frac{\text{vol}(\Omega)}{2} - \frac{1}{2} \int_{\Omega} \left(\frac{\langle \nabla \mathcal{T}(\varphi(\mathbf{x})), \nabla \mathcal{R}(\mathbf{x}) \rangle}{\|\nabla \mathcal{T}(\varphi(\mathbf{x}))\|_2 \|\nabla \mathcal{R}(\mathbf{x})\|_2} \right)^2 dx. \quad (3.38)$$

Das Distanzmaß wurde eingeführt von Haber und Modersitzki [43]. Dabei wird unter dem Integral (3.37) der Kosinus zwischen den Kanten der Bilder, $\nabla \mathcal{T}(\varphi(\mathbf{x}))$ und $\nabla \mathcal{R}(\mathbf{x})$, betrachtet. Sind diese parallel zueinander, ist der Kosinus ± 1 und die Distanz Null. Der größte Wert wird bei Orthogonalität der Kanten erreicht. Die betrachteten Bilder sind allerdings typischerweise verrauscht. Durch dieses Rauschen und die Normalisierung werden unnatürliche Kanten eingeführt. Um dagegen anzuwirken, kann zusätzlich ein Parameter ϵ eingeführt werden. Anstatt der euklidischen Norm $\|\nabla \mathcal{T}(\varphi(\mathbf{x}))\|_2$ und $\|\nabla \mathcal{R}(\mathbf{x})\|_2$ betrachten wir dann $\|\nabla \mathcal{T}(\varphi(\mathbf{x}))\|_{\epsilon} = \sqrt{\|\nabla \mathcal{T}(\varphi(\mathbf{x}))\|_2^2 + \epsilon^2}$ bzw. $\|\nabla \mathcal{R}(\mathbf{x})\|_{\epsilon} = \sqrt{\|\nabla \mathcal{R}(\mathbf{x})\|_2^2 + \epsilon^2}$. Für eine natürliche Kante gilt dann $\|\nabla \mathcal{T}(\varphi(\mathbf{x}))\|_2 > \epsilon$ bzw. $\|\nabla \mathcal{R}(\mathbf{x})\|_2 > \epsilon$. Die Wahl des Parameters ist für die Effektivität des Distanzmaßes sehr wichtig, jedoch auch sehr schwierig. Eine automatische Wahl ist in [43] angegeben.

Wir betrachten die Diskretisierung des zweiten Terms aus (3.38) auf einem zellzentrierten Gitter. Zur Bestimmung der Kantenbilder nutzen wir den Differenzenquotienten (3.16) mit Neumann-Randbedingungen, d. h. $\nabla \mathcal{T}_{\varphi}(\mathbf{x}) = \nabla \mathcal{R}(\mathbf{x}) = 0$, $\mathbf{x} \in \partial\Omega$. Dadurch sollen die Kanten am Rand gleich ihrer Nachbarn im Inneren fortgesetzt werden. Exemplarisch betrachten wir den zweidimensionalen Fall. Mit

$$D_k = \begin{pmatrix} -1 & 1 & & & \\ -1 & 0 & 1 & & \\ & \ddots & \ddots & \ddots & \\ & & -1 & 0 & 1 \\ & & & 1 & -1 \end{pmatrix} \in \mathbb{R}^{m_k \times m_k}, \quad k \in \{1, 2\}, \quad (3.39)$$

und

$$\begin{aligned}\mathbf{D}^1 &= I_{m_2} \otimes D_1 \\ \mathbf{D}^2 &= D_2 \otimes I_{m_1}\end{aligned}$$

können wir die finiten Differenzen in x_1 - bzw. x_2 -Richtung auf Bildvektoren anwenden. Wir erhalten die Diskretisierung des Skalarprodukts der Kantenbilder mit

$$\langle \nabla \mathcal{T}(\varphi), \nabla \mathcal{R} \rangle \approx \text{diag}(\mathbf{D}^1 \mathbf{T}_\varphi) \mathbf{D}^1 \mathbf{R} + \text{diag}(\mathbf{D}^2 \mathbf{T}_\varphi) \mathbf{D}^2 \mathbf{R} =: \mathbf{s}$$

und

$$\mathbf{r}_1 := \text{diag}(\mathbf{s}) \mathbf{s}, \quad (3.40)$$

mit $\text{diag} : \mathbb{R}^n \rightarrow \mathbb{R}^{n \times n}$,

$$\text{diag}(\mathbf{s}) := \begin{pmatrix} s_1 & & \\ & \ddots & \\ & & s_n \end{pmatrix}.$$

Die Diskretisierung der um ϵ erweiterten Norm ist gegeben durch

$$\begin{aligned}\|\nabla \mathcal{R}\|_\epsilon &\approx \sqrt{\text{diag}(\mathbf{D}^1 \mathbf{R}) \mathbf{D}^1 \mathbf{R} + \text{diag}(\mathbf{D}^2 \mathbf{R}) \mathbf{D}^2 \mathbf{R} + \epsilon^2} =: \mathbf{nR} \\ \|\nabla \mathcal{T}(\varphi)\|_\epsilon &\approx \sqrt{\text{diag}(\mathbf{D}^1 \mathbf{T}_\varphi) \mathbf{D}^1 \mathbf{T}_\varphi + \text{diag}(\mathbf{D}^2 \mathbf{T}_\varphi) \mathbf{D}^2 \mathbf{T}_\varphi + \epsilon^2} =: \mathbf{nT}_\varphi,\end{aligned}$$

bei der die Wurzeloperation punktweise ist. Damit ergibt sich für den Nenner des Residuums

$$\|\nabla \mathcal{T}_\varphi\|_\epsilon \|\nabla \mathcal{R}\|_\epsilon \approx \text{diag}(\mathbf{nT}_\varphi) \mathbf{nR} =: 1 \otimes \mathbf{r}_2. \quad (3.41)$$

Mit \mathbf{r}_1 (3.40), \mathbf{r}_2 (3.41) und $\bar{\omega} := \text{vol}(\Omega) = \prod_{i=1}^2 (\omega_{2i} - \omega_{2i-1})$ erhalten wir

$$\mathbf{D}(\varphi) = \frac{\bar{\omega}}{2} - \frac{1}{2} \|\text{diag}(\mathbf{r}_1) \mathbf{r}_2\|_2^2.$$

Es zeigt sich wiederum, dass das Distanzmaß in Least-Squares-Form vorliegt und damit mit dem Gauss-Newton-Verfahren angegangen werden kann. Für die Ableitung ergeben sich mit der Kettenregel und punktweiser Potenzierung

$$\begin{aligned}d\mathbf{r}_1 &= \text{diag}(\mathbf{D}^1 \mathbf{R}) \mathbf{D}^1 + \text{diag}(\mathbf{D}^2 \mathbf{R}) \mathbf{D}^2 \\ d\mathbf{r}_2 &= -\text{diag}(1 \otimes \text{diag}(\mathbf{nR}) \mathbf{nT}^3) (\text{diag}(\mathbf{D}^1 \mathbf{T}_\varphi) \mathbf{D}^1 + \text{diag}(\mathbf{D}^2 \mathbf{T}_\varphi) \mathbf{D}^2).\end{aligned}$$

Folglich gilt

$$d\mathbf{r} = (\text{diag}(\mathbf{r}_2) d\mathbf{r}_1 + \text{diag}(\mathbf{r}_1) d\mathbf{r}_2) d\mathbf{T}_\varphi.$$

Eine genauere Betrachtung ist z. B. in [68] zu finden.

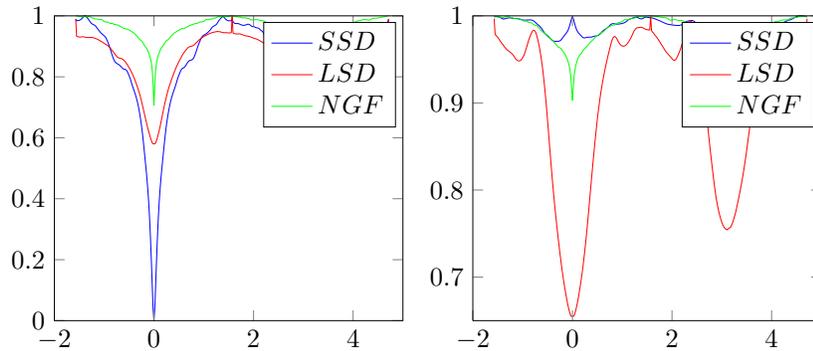


Abbildung 3.15.: Plot der Distanzmaße ausgewertet für Rotationen um den Winkel $\gamma \in \{-\frac{\pi}{2}, \dots, \frac{3\pi}{2}\}$. Links ist der monomodale (T_2 vs T_2), rechts der multimodale (T_2 vs T_1) Fall aus Abbildung 3.14 zu betrachten. Es ist deutlich das Problem des SSD im multimodalen Fall zu sehen

Vergleich der Distanzmaße Zum Abschluss des Abschnitts über Distanzmaße wollen wir ein einfaches Beispiel anführen, um die obigen Aussagen visuell zu unterstützen. Dazu betrachten wir einen mono- und einen multimodalen Fall basierend auf den Daten aus Abbildung 3.14. Der monomodale Fall betrachtet eine Simulation eines T_2 -gewichteten MR-Bildes, das wir um sich selbst rotierten lassen. Bei einem Rotationswinkel $\gamma = 0$ sollten beide Bilder gleich sein und damit ein Minimum vorliegen. Für das zweite Beispiel dient die Simulation des T_2 -gewichteten MR-Bildes wiederum als Referenz. Als Template wählen wir jedoch eine Simulation eines T_1 -gewichteten MRTs. Die Daten kommen wiederum vom MRT-Simulator der BrainWeb-Homepage [1].

Die Distanzen wurden für beide Fälle für Rotationswinkel $\gamma \in \{-\frac{\pi}{2}, \dots, \frac{3\pi}{2}\}$ ausgewertet. Die resultierenden Plots sind in Abbildung 3.15 zu finden. Während das SSD nur im monomodalen Fall ein Minimum für $\gamma = 0$ annimmt, haben wir im multimodalen Fall sogar ein Maximum. Das LSD und das NGF Distanzmaß zeigen dagegen in beiden Fällen Minima an. Ein deutlicher Unterschied zwischen LSD und NGF ist das Verhalten in der Umgebung des Minimums. Hier ist die Steigung des NGF deutlich höher. Auch aus Optimierungssicht ist dieses Verhalten besser, allerdings ist das LSD einfacher zu berechnen und benötigt keinen zusätzlichen Parameter.

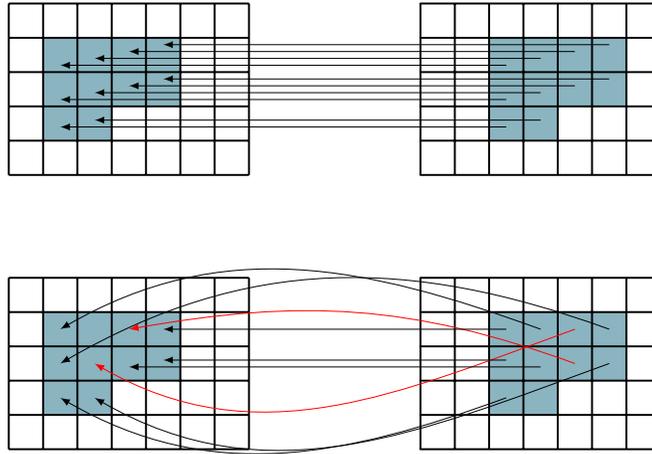


Abbildung 3.16.: Zwei mögliche Transformationen von dem Bild links auf das Bild rechts. Oben sehen wir eine einfache Translation. Nachbarschaftsverhältnisse zwischen den einzelnen Pixel bleiben erhalten. Die Transformation wirkt plausibel. Unten sehen wir hingegen eine Transformation, die die farbigen Voxel quasi beliebig austauscht. Diese Transformation wirkt, besonders im Zusammenhang mit medizinischer Bildverarbeitung, nicht plausibel. Die Nachbarschaftsverhältnisse bleiben nicht erhalten

3.6. Plausibilität von Transformationen

Im vorangegangenen Abschnitt haben wir Möglichkeiten kennengelernt, Bilder miteinander zu vergleichen. Wir könnten jetzt starten Bilder zu registrieren, indem wir eine sinnvolle Distanz bezüglich der Transformation minimieren, d. h.

$$\min_{\varphi} \mathcal{D}(\mathcal{R}, \mathcal{T}; \varphi). \quad (3.42)$$

Dieses Problem ist schlecht-gestellt [66] und führt typischerweise zu ungewollten Lösungen. Im Beispiel der nichtparametrischen Registrierung zweier Binärbilder mittels SSD ist erst einmal jede Transformation optimal, die die beiden weißen Flächen aufeinander bringt. Dadurch können unter anderem vorher benachbarte Punkte voneinander getrennt werden, das Nachbarschaftsverhältnis bleibt nicht erhalten. Betrachten wir statt Binärbildern, medizinische Bilddaten könnte dies z. B. einem Vertauschen von Knochen entsprechen. Eine solche Transformation ist nicht plausibel. Diese Problematik ist in Abbildung 3.16 dargestellt. Eine PLAUSIBLE

TRANSFORMATION sollte dagegen Nachbarschaftsverhältnisse und somit anatomische Strukturen erhalten. Des Weiteren sollten korrespondierende Punkte aufeinander abgebildet werden. Um unplausible Transformationen zu verhindern, werden wir bestimmte Forderungen an die Transformation stellen. Eine Möglichkeit ist die Transformation zu parameterisieren, z. B. rigide Transformationen zu wählen. Offensichtlich bekommen wir plausible und vertrauenswürdige Lösungen. Die Kehrseite der Medaille ist, dass durch diese Modellannahme wichtige Transformationen ausgeschlossen werden. So können nichtlineare Deformationen, z. B. verursacht durch Herzschlag oder Atmung, nicht eingefangen werden. Um plausible nichtlineare Transformationen zu erlangen, werden wir das Modell (3.42) um einen Regularisierer erweitern. Der Regularisierungsterm soll gewisse Eigenschaften der Transformation im Registrierungsschritt begünstigen bzw. ungewollte Eigenschaften bestrafen und damit den Raum der möglichen Lösungen einschränken und das schlecht-gestellte Problem verbessern.

Nichtsdestotrotz, Regularisierung alleine ist nicht immer ausreichend, eine plausible Transformation zu finden. Erinnern wir uns an das Problem der Registrierung prä- und postoperativer Leber-Bilder aus der Einführung. Modellieren wir unser Problem mittels eines (Standard-)Distanzmaßes und eines der vorgestellten Regularisierer, wird die Minimierung des Zielfunktional eine Lösung liefern, die ein Nachwachsen der postoperativen Leber erzeugt. Für das Ziel der Registrierung, einer Evaluation der Umsetzung der Operation bezüglich der Operationsplanung, ist diese Transformation allerdings nicht plausibel, denn es werden nicht immer korrespondierende Punkte aufeinander abgebildet. Abbildung 3.17 veranschaulicht dieses anhand eines synthetischen Beispiels. Es stellt sich die Frage, wie ein solches Verhalten verhindert werden kann. Hierfür werden wir im Anschluss das Modell aus Distanzmaß und Regularisierer um geeignetes Zusatzwissen erweitern.

3.6.1. Regularisierer

Wie gerade erwähnt ist die Einführung eines Regularisierers eine Möglichkeit, Plausibilität bei nichtlinearen Transformationen zu gewinnen. Dieser Regularisierer wird zum Distanzmaß in Problem (3.42) addiert und soll bestimmte Arten von Transformationen, meist glatte, bevorzugen. Durch die damit verbundene Einschränkung des Suchraums dient der Regularisierer auch dazu, das schlecht-gestellte Problem zu verbessern. Ein REGULARISIERUNGSPARAMETER $\alpha \in \mathbb{R}$ balanciert das Verhältnis zwischen Datentreue, gegeben durch das Distanzmaß und der durch den Regularisierer gestellten Eigenschaft der Transformation.

Wir erhalten das ZIELFUNKTIONAL für die nichtparametrische Bildregistrierung

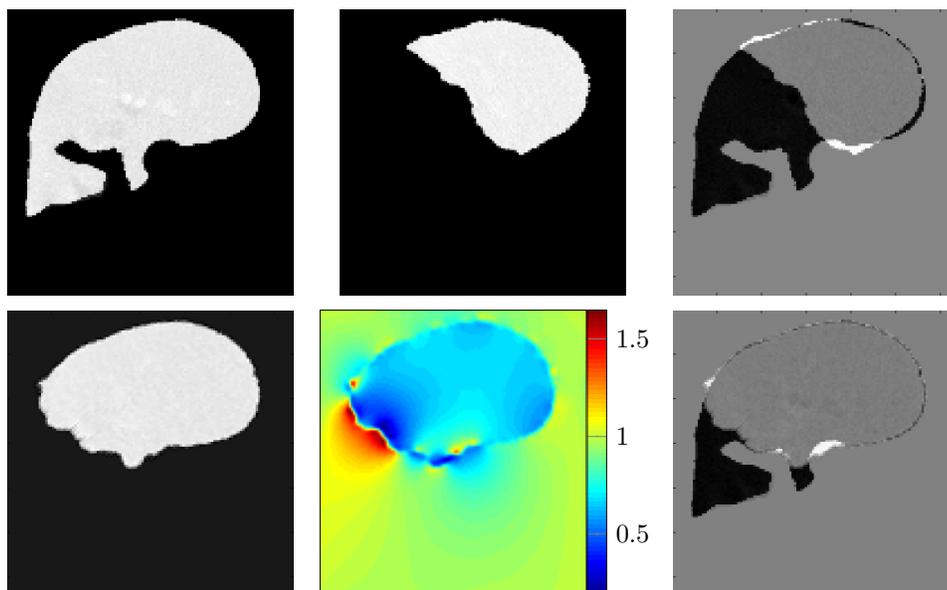


Abbildung 3.17.: Eine elastische Registrierung mit SSD einer präoperativen (oben links) mit einer postoperativen (oben Mitte) Leberschicht. Das Startdifferenzbild (oben rechts) zeigt die Differenz der Grauwerte zu Beginn der Registrierung. Durch das SSD Distanzmaß getrieben, erreichen beide Bilder ihre größt-mögliche Ähnlichkeit bei vollständiger Überlappung der weißen Flächen. Das resultierende deformierte Templatebild (unten links) weist daher deutlich mehr Volumen auf als in seiner ursprünglichen Form. Dies ist auch im Differenzbild des deformierten Templatebildes mit dem Referenzbild (unten rechts) deutlich zu erkennen. Die Volumenbetrachtung (unten, Mitte) besitzt ein Minimum bei 0.1869 und ist damit immer größer 0. Der Regularisierer verhindert damit zwar das Auftreten von Faltungen, kann aber das ungewollte Nachwachsen nicht verhindern.

mit

$$\mathcal{J}(\mathcal{R}, \mathcal{T}; \varphi) = \mathcal{D}(\mathcal{R}, \mathcal{T}; \varphi) + \alpha \mathcal{S}(\varphi - \varphi^{(0)}), \quad (3.43)$$

mit REGULARISIERER \mathcal{S} . Die Deformation $\varphi^{(0)}$ ist eine Referenztransformation, deren Anteile nicht bestraft werden sollen. Diese kann z. B. aus einer Vorregistrierung stammen. Hier werden wir die drei prominentesten Regularisierer in der Registrierung vorstellen, den diffusiven, den curvature und den elastischen Regularisierer.

Durch die Regularisierer werden typischerweise erste oder zweite Ableitungen bestraft, wir sprechen dann von einer Regularisierung ERSTER bzw. ZWEITER ORDNUNG.

Diffusiver Regularisierer Ein Regularisierer erster Ordnung ist der DIFFUSIVE REGULARISIERER, gegeben durch

$$\mathcal{S}(\varphi - \varphi^{(0)}) = \frac{1}{2} \int_{\Omega} \sum_{i=1}^2 \|\nabla (\varphi_i(\mathbf{x}) - \varphi_i^{(0)}(\mathbf{x}))\|^2 dx. \quad (3.44)$$

Die diffusive Regularisierung wurde 2001 von Fischer und Modersitzki [29, 27] in die Bildregistrierung eingeführt. Der Regularisierer bestraft die Gradienten der Transformation, $\nabla \varphi_i$, jeder Richtung $i \in \{1, 2\}$. Dadurch werden Oszillationen der Transformation unterdrückt.

Für die Betrachtung der Diskretisierung setzen wir $\varphi - \varphi^{(0)} = u$. Wir starten mit dem eindimensionalen Fall auf einem zellzentrierten Gitter $X^{cc}(\Omega, m)$ und nutzen Neumann-Null-Randbedingungen, d. h. $u'(\omega_1) = u'(\omega_2) = 0$ um später Randterme wegfallen lassen zu können. Im Hinblick auf den später vorgestellten curvature Regularisierer diskretisieren wir hier nicht direkt die erste Ableitung, sondern wählen einen anderen Zugang. Eine Anwendung der partiellen Integration und Ausnutzung der Randbedingungen führt zu

$$\begin{aligned} \mathcal{S}(u) &= \frac{1}{2} \int_{\omega_1}^{\omega_2} (u'(x))^2 dx \\ &= \frac{1}{2} \left(\int_{\omega_1}^{\omega_2} -u''(x)u(x) dx + [u'(x)u(x)]_{\omega_1}^{\omega_2} \right) \\ &= \frac{1}{2} \int_{\omega_1}^{\omega_2} -u''(x)u(x) dx. \end{aligned}$$

Das Integral wird mittels Rechteckregel diskretisiert. Es folgt

$$\mathcal{S}(u) \approx \frac{h}{2} \sum_{i=1}^m -u''(x_i)u(x_i).$$

Nun können wir die finiten Differenzen (3.17) für die zweite Ableitung mit einer Approximationsgüte von $O(h^2)$ nutzen. Wir erlangen die Darstellung

$$\mathcal{S}(u) \approx \frac{h}{2} \sum_{i=1}^m \frac{1}{h^2} (-u(x_{i-1}) + 2u(x_i) - u(x_{i+1})) u(x_i).$$

Die Randbedingungen liefern $u(x_0) = u(x_1)$ und $u(x_{m+1}) = u(x_m)$. Durch Einführung der Ableitungsmatrix

$$\mathbf{A}^m = \frac{1}{h^2} \begin{pmatrix} 1 & -1 & & & \\ -1 & 2 & -1 & & \\ & \ddots & \ddots & \ddots & \\ & & -1 & 2 & -1 \\ & & & -1 & 1 \end{pmatrix} \in \mathbb{R}^{m \times m}$$

erhalten wir die Diskretisierung des diffusiven Regularisierers mit

$$\mathbf{S}(\mathbf{u}) = \frac{h}{2} \mathbf{u}^T \mathbf{A}^m \mathbf{u},$$

wobei $\mathbf{u} \in \mathbb{R}^m$ die Diskretisierung von u ist. Die Erweiterung auf den mehrdimensionalen Fall erhalten wir mittels des Kroneckerproduktes. Wir formulieren dies exemplarisch für den zweidimensionalen Fall. Es ergibt sich

$$\mathbf{A}^{\text{diff}} = \begin{pmatrix} \mathbf{A} & \\ & \mathbf{A} \end{pmatrix} \text{ mit } (\mathbf{A} = (\mathbf{I}_{m_2} \otimes \mathbf{A}^{m_1}) + (\mathbf{A}^{m_2} \otimes \mathbf{I}_{m_1}))$$

und damit

$$\mathbf{S}(\mathbf{u}) = \frac{\bar{h}}{2} \mathbf{u}^T \mathbf{A}^{\text{diff}} \mathbf{u}.$$

Wir sehen leicht, dass sich für die diskreten Ableitungen

$$d\mathbf{S} = \mathbf{A}^{\text{diff}} d\mathbf{u}$$

und

$$d^2\mathbf{S} = \mathbf{A}^{\text{diff}} d^2\mathbf{u}$$

ergeben. Im Gegensatz zur Berechnung der zweiten Ableitung des Distanzmaßes nutzen wir beim Regularisierer die exakte zweite Ableitung.

Curvature Regularisierer Der curvature Regularisierungsterm, eingeführt 2002 von Fischer und Modersitzki [28], gehört zu den Regularisierern zweiter Ordnung und ist definiert durch

$$\mathcal{S}(\varphi - \varphi^{(0)}) = \frac{1}{2} \int_{\Omega} \sum_{i=1}^2 \left(\Delta \left(\varphi_i(\mathbf{x}) - \varphi_i^{(0)}(\mathbf{x}) \right) \right)^2 dx. \quad (3.45)$$

Die Haupteigenschaft besteht in der Begünstigung von glatteren Transformationen als durch Regularisierer erster Ordnung bevorzugte. Durch die Bestrafung der zweiten Ableitung mit dem Laplace-Operator liegen zudem lineare Transformationen im Kern des Regularisierers und werden nicht bestraft. Neben diesem Regularisierer gibt es weitere Vorschläge zur Regularisierung zweiter Ordnung [12].

Die Diskretisierung für den curvature Regularisierer sowie dessen Ableitungen verläuft analog, ohne Anwendung der partiellen Integration, zum diffusiven Regularisierer. Es ergibt sich

$$\mathbf{A}^{\text{curv}} = \left(\mathbf{A}^{\text{diff}} \right)^T \mathbf{A}^{\text{diff}}, \quad (3.46)$$

unabhängig von der Dimension d .

Elastischer Regularisierer Bereits 1981 wurde von Broit ein Regularisierungsterm vorgestellt [10]. Während der diffusive und der curvature Regularisierer ihre Ursprünge in Glattheitsannahmen haben, ist dieser Regularisierer physikalisch motiviert. Dem Regularisierer liegt das Modell der linearen Elastizität zu Grunde.

Der ELASTISCHE REGULARISIERER ist definiert durch

$$\begin{aligned} \mathcal{S}(\varphi - \varphi^{(0)}) = \frac{1}{2} \int_{\Omega} \sum_{i=1}^2 \mu \|\nabla \left(\varphi_i(\mathbf{x}) - \varphi_i^{(0)}(\mathbf{x}) \right)\| \\ + (\mu + \lambda) \left(\nabla \cdot \left(\varphi_i(\mathbf{x}) - \varphi_i^{(0)}(\mathbf{x}) \right) \right)^2 dx, \end{aligned} \quad (3.47)$$

wobei μ, λ die sogenannten Lamé-Konstanten sind, die das Verhalten verschiedener Materialien modellieren. Für $\mu = 1, \lambda = -1$ kann die diffusive Regularisierung als ein Spezialfall angesehen werden.

Da im Folgenden der elastische Regularisierer nicht weiter verwendet wird, überspringen wir die Diskretisierung hier. Für Informationen zu diesem Thema sei auf [68] verwiesen.

Die Wahl des Regularisierers hängt von der jeweiligen Anwendung ab. Abbildung 3.18 zeigt das Verhalten der drei Regularisierer. Wir können deutlich sehen, dass die

curvature Regularisierung die glattesten Ergebnisse erzielt. Aber auch das Nachwachsen der Leber kann in allen Fällen beobachtet werden. Eine Möglichkeit, dieses zu verhindern, besteht darin, Zusatzwissen in das Modell einfließen zu lassen.

3.6.2. Zusatzwissen

Wie wir im letzten Beispiel gesehen haben, ist es manchmal unvermeidbar mehr zu tun, als ein Distanzmaß und einen Regularisierer aus dem Baukastensystem zu wählen, um das Problem zu modellieren. Wir müssen mehr Wissen in das Modell einfließen lassen. In unserem Problem aus Abbildung 3.17 wird die Registrierung sonst versuchen, die weißen Bereiche der beiden Bilder komplett überlappen zu lassen, anstatt korrespondierende Punkte aufeinander abzubilden. Dieses zusätzliche Wissen, oder auch ZUSATZWISSEN, zu identifizieren, ist jedoch eine schwierige Aufgabe. Wir können versuchen, jedes Detail in das Modell zu integrieren. Dieser Ansatz führt allerdings auf ein sehr komplexes Problem, welches schwer zu lösen aber vor Allem schwierig aufzustellen und zu verifizieren ist. Ein zu einfaches Modell, bestehend z. B. alleine aus einem Distanzmaß und einem Regularisierer mag jedoch unspezifisch sein und das eigentliche Problem nicht einfangen. Ziel muss es sein, eine Balance im Modell zwischen Akkuratess, Detailtiefe und klarem Arrangement bzw. Einfachheit zu finden. Zu diesem Zwecke kann Expertenwissen aus dem Bereich der Anwendung äußerst hilfreich sein.

Einige Beispiele dazu sind in der Literatur zu finden. Eines hiervon ist das Einbringen von Landmarkeninformationen in die intensitätsbasierte Registrierung [62, 73, 61]. Dabei wird darauf geachtet, dass besonders markante, korrespondierende Punkte, z. B. Gefäßverzweigungen, aufeinander abgebildet werden. Da das Setzen solcher Landmarken im Allgemeinen mit einem Fehler behaftet ist, können auch Toleranzbereiche angegeben werden. Eine anderes Beispiel ist die Verhinderung oder Einschränkung von Volumenänderung (siehe Abschnitt 3.4) [79, 42, 44]. In Applikationen, bei denen z.B. die Größe von Tumoren erhalten bleiben soll, kann dies von großem Wert sein. Auch Unterschiede in den möglichen Deformationen verschiedener Regionen wurden integriert [67, 85]. So verhält sich Knochen rigide, während sich das angrenzende Gewebe nichtlinear verformen kann. Durch das Einfügen von Zusatzwissen können auch lokale Minima ausgeschlossen werden. Besonders bei der Verwendung der Landmarkeninformationen bei Gefäßverzweigungspunkten ist dies offensichtlich. Mit der Korrespondenz der Gefäßverzweigungen geht auch die Korrespondenz der Gefäße einher, die sonst teilweise schwierig herzustellen ist. Die Robustheit des Registrierungsalgorithmus wird dadurch zusätzlich erhöht.

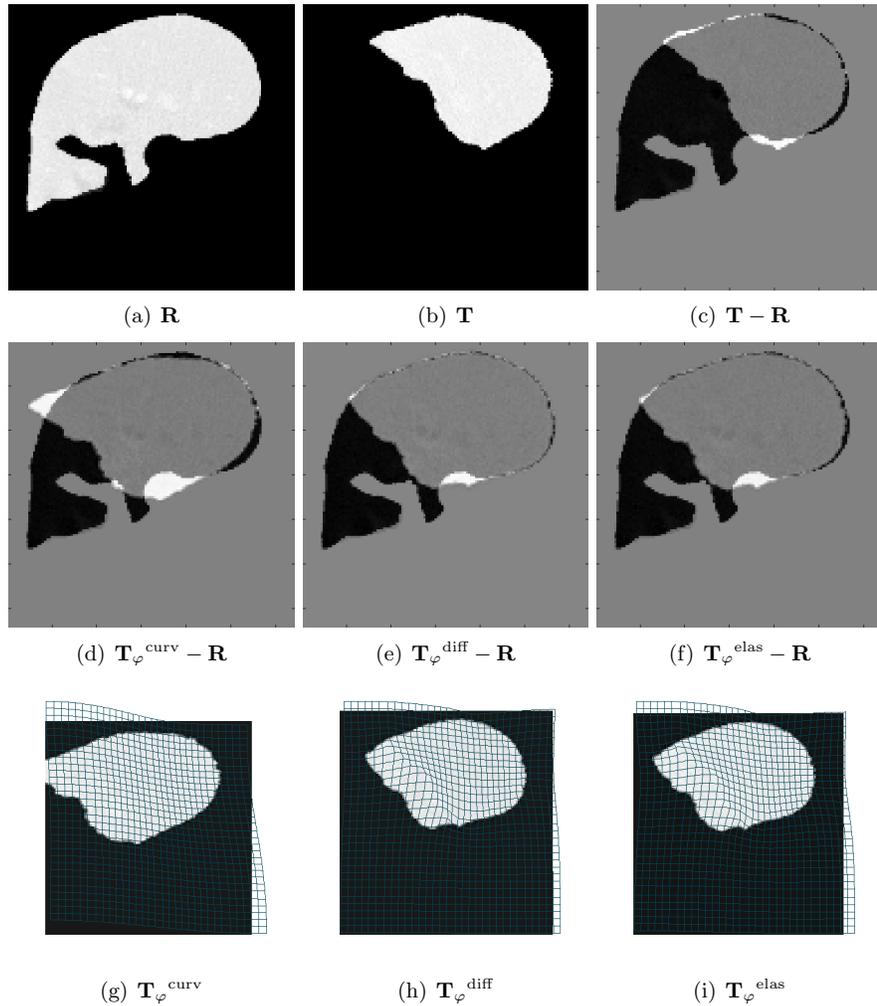


Abbildung 3.18.: Es ist ein synthetisches Beispiel der Registrierung einer post-operativen auf eine präoperative Leber zu sehen. Dazu wurde ein nichtlineares Transformationsmodell mit drei unterschiedlichen Regularisierern gewählt. Curvature Regularisierung (links), diffusive Regularisierung (Mitte) und elastische Regularisierung (rechts) wurden eingesetzt mit einem Regularisierungsparameter, so dass faltungsfreie Transformationen erreicht werden. Oben sehen wir die Ausgangslage. In der Mitte die erzeugten Differenzbilder und in der untersten Zeile die deformierten Templatebilder mit dem Deformationsfeld. Wir sehen deutlich den Unterschied zwischen Regularisierern erster und zweiter Ordnung. Der curvature Regularisierer erzeugt eine deutlich glattere Lösung.

Das Registrierungsproblem ist dann mittels

$$\min_{\varphi} \mathcal{J}(\mathcal{R}, \mathcal{T}; \varphi) \quad \text{u. d. NB} \quad \begin{cases} c_i(\varphi) = 0, & i = \{1, \dots, m\} \\ c_i(\varphi) \geq 0, & i = \{m+1, \dots, \tilde{m}\} \end{cases}$$

gegeben. Im Rahmen dieser Arbeit wollen wir den Fall mit Gleichheitsnebenbedingungen betrachten und erhalten damit das Registrierungsproblem mit integriertem Zusatzwissen in der Form

$$\min_{\varphi} \mathcal{J}(\mathcal{R}, \mathcal{T}; \varphi) \quad \text{u. d. NB} \quad c_i(\varphi) = 0, \quad i = 1, \dots, m, \quad (3.48)$$

mit

$$\mathcal{J}(\mathcal{R}, \mathcal{T}; \varphi) = \mathcal{D}(\mathcal{R}, \mathcal{T}; \varphi) + \alpha \mathcal{S}(\varphi - \varphi^{(0)}).$$

Oft werden zur Lösung des Problems die Nebenbedingungen über einen quadratischen Strafterm in die Zielfunktion eingebracht und das resultierende unrestringierte Problem gelöst, d. h.

$$\min_{\varphi} \mathcal{J}(\mathcal{R}, \mathcal{T}; \varphi) + \mu \mathcal{P}(\varphi),$$

wobei $\mathcal{P}(\varphi) = \|c(\varphi)\|^2$ und $c(\varphi)^T = (c_1(\varphi), \dots, c_m(\varphi))$ [85, 79]. Im Gegensatz zum Verfahren des quadratischen Strafterms für restringierte Optimierung (vgl. Abschnitt 2.2), wird häufig der Parameter μ über die Iterationen fixiert. Wir sprechen dann auch von der Einbindung WEICHER NEBENBEDINGUNGEN. Die Wahl des Parameters μ , zusätzlich zum Regularisierungsparameter α , stellt eine weitere Herausforderung dar. Wird μ zu groß gewählt, wird der Wert auf Ähnlichkeit der Bilder und Glattheit der Lösung kaum noch beachtet. Bei einer zu kleinen Wahl hingegen, hat das Zusatzwissen zu wenig Einfluss auf die Lösung. Wie wir gesehen haben, liefert dieser Ansatz auch keine Lösung des Problems (3.6.2). Zudem ist die Hessematrix bei Anwendung des quadratischen Strafterms für sehr große μ , und damit hoher Genauigkeit der Nebenbedingungen, schlecht konditioniert.

Im Optimierungskapitel hatten wir uns mit dem Augmented Lagrangian Ansatz beschäftigt, bei dem die Zielfunktion die Form

$$\mathcal{J}(\mathcal{R}, \mathcal{T}; \varphi) = \mathcal{D}(\mathcal{R}, \mathcal{T}; \varphi) + \alpha \mathcal{S}(\varphi - \varphi_0) - \lambda^T c(\varphi) + \mu \mathcal{P}(\varphi)$$

besitzt. Eine Lösung des Augmented Lagrangian Ansatzes liefert auch eine Lösung des Problems (3.48). Diese Art der Optimierung wurde in [83, 73] umgesetzt. Bei unseren Problemen werden wir diesen Ansatz ebenfalls wählen. Für Ungleichheitsnebenbedingungen sind Lösungen in [61, 39] gegeben.

In unseren Anwendungen werden wir Probleme beleuchten, die ohne den Einsatz von Zusatzwissen nicht sinnvoll zu lösen sind, wie das Problem der Registrierung

prä- und postoperativer Leberdaten. Wir werden Ideen für Zusatzinformationen vorstellen, diese miteinander vergleichen und innerhalb des Augmented Lagrangian Ansatzes die Wahl unrestringierter Optimierungsverfahrenen testen.

Bevor wir mit diesem Teil anfangen, stellen wir einen allgemeinen Registrierungsrahmen und zwei spezielle Registrierungstechniken vor, die Robustheit und Geschwindigkeit der Verfahren erhöhen.

3.7. Allgemeiner Registrierungsrahmen

Wir haben im vorangegangenen Teil des Kapitels die Hauptbausteine einer Registrierung kennengelernt. Hier geben wir einen allgemeinen Rahmen zur Verbindung dieser innerhalb dieser Arbeit an. Eine erste Übersicht ist Abbildung 3.19 zu entnehmen. Diese geht davon aus, dass als Eingabe ein Referenzbild \mathcal{R} , Templatebild \mathcal{T} und eine initiale Transformation $\varphi^{(0)}$ existieren. Die Bilder \mathcal{T}, \mathcal{R} durchlaufen einen Vorverarbeitungsschritt und werden dann in der Vor-Registrierung grob aneinander ausgerichtet. Als Zwischenergebnis erhalten wir die Transformation φ^{grob} , die als Startwert für den Haupt-Registrierungsschritt dient. Dieser berechnet typischerweise eine nichtlineare Lösung φ^{opt} . Im Folgenden gehen wir kurz auf die Schritte der Vorverarbeitung, Vorregistrierung und Haupt-Registrierung ein.

Vorverarbeitung Nicht immer liegen die Daten in einer geeigneten Form vor. So sind wir vielleicht an einer Registrierung von Leberdaten interessiert, die Daten beinhalten jedoch den kompletten abdominalen Bereich. Damit das nachfolgende Registrierungsverfahren sich auf den wichtigen Bereich konzentrieren kann, blendet man z. B. den Bereich außerhalb der Leber aus. Dazu ist eine Segmentierung notwendig [7, 60, 13]. Bei Bildern aus der Magnetresonanztomographie kann beispielsweise aufgrund von Aufnahmeartefakten eine sogenannte Bias-Korrektur notwendig sein [92, 91].

Da wir uns in dieser Arbeit im Registrierungsteil vertiefen, wollen wir auf die Vorverarbeitung nicht weiter eingehen. Aus Optimierungssicht ist jedoch eine Glättung der Bilder häufig von Vorteil. Mit diesem Aspekt beschäftigen wir uns deshalb im nächsten Abschnitt über Multiskalen-Ansätze näher. Als initiale Transformation $\varphi^{(0)}$ dient typischerweise die Identität $\varphi^{(0)} = id$. Liegen aber die Urbildbereiche der Bilder weit auseinander, so dass keine initiale Überlappung existiert, kann z. B. der Schwerpunkt der beiden Bilder mittels einer Translation aufeinander gebracht werden.

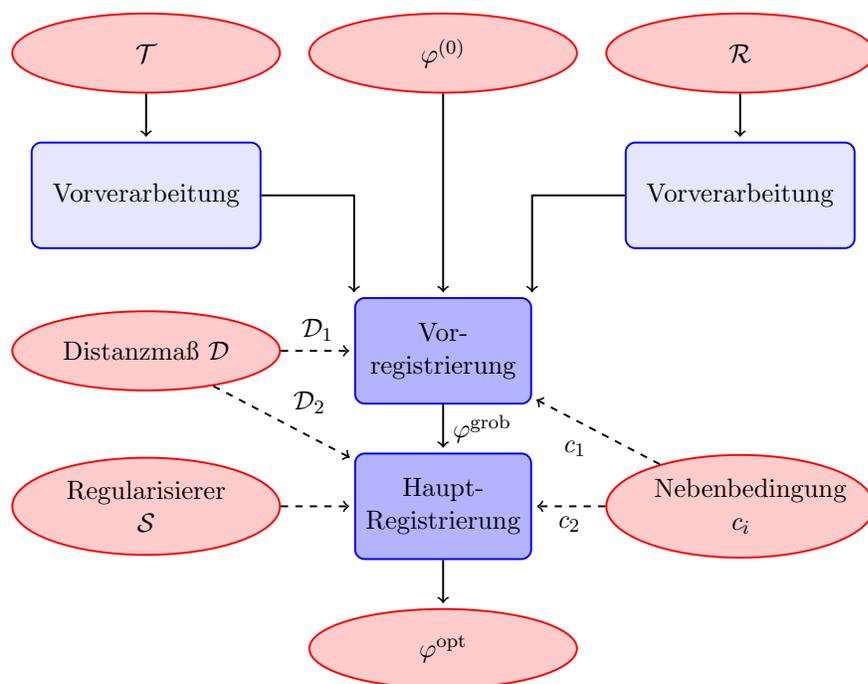


Abbildung 3.19.: Ein genereller Registrierungsrahmen, der im Zuge dieser Arbeit verwendet wird. Eingabe hierbei ist ein Templatebild \mathcal{T} , Referenzbild \mathcal{R} und eine initiale Deformation $\varphi^{(0)}$. Als Ausgabe erhalten wir eine Lösung φ^{opt} des Problems 3.48.

Vorregistrierung Dem Vorverarbeitungsschritt folgt die Vorregistrierung. Diese spielt eine essentielle Rolle bei der Ausrichtung von \mathcal{R} und \mathcal{T} . Die Idee ist, eine robuste und vertrauenswürdige Grundausrichtung der beiden Bilder zu finden. Danach sollen nur noch leichte, nichtlineare Korrekturen nötig sein. Ist diese Vorregistrierung nicht akkurat, wird der Registrierungsalgorithmus im Allgemeinen ein ungewolltes lokales Minimum finden.

In diesem Schritt ist es schon wichtig, das richtige Modell zu wählen. Für die Transformation wird häufig ein affin-lineares oder rigides Modell gewählt. Sind Nebenbedingungen c_i , $i \in \{1, \dots, m\}$ im Problem enthalten, sollte dies bei der Wahl des Transformationsmodells berücksichtigt werden. Bei der Forderung nach Volumenerhaltung ist so z. B. das rigide Transformationsmodell das Mittel der Wahl. Das affin-lineare Transformationsmodell hingegen könnte aufgrund der zugelassenen Skalierung eine nicht zulässige Lösung als Startwert für die Hauptregistrierung liefern. Die Wahl eines adäquaten Distanzmaßes \mathcal{D}_1 ist problemabhängig und hängt von den Daten ab. Aspekte, die diese Wahl beeinflussen, sind unter anderem die gegebenen Modalitäten und Vorverarbeitungsschritte wie z. B. Segmentierungen. Die resultierende Zielfunktion wird diskretisiert und dann optimiert. Wir erhalten eine erste grobe Ausrichtung φ^{grob} , die sehr vertrauenswürdig ist.

Hauptregistrierung Die Hauptregistrierung berechnet, basierend auf der Grobausrichtung φ^{grob} , die nichtlinearen Anteile der gesuchten Transformation φ^{opt} . Diese optimale Transformation ist die Lösung des Modells, bestehend aus einem Modell der Transformation φ^{opt} , einem Distanzmaß \mathcal{D}_2 und einem Regularisierer \mathcal{S} . Zusätzlich kann Zusatzwissen über Nebenbedingungen c_i , $i \in \{1, \dots, m\}$ eingebracht werden, so dass das zugrundeliegende Problem adäquat beschrieben wird. Für das Distanzmaß \mathcal{D}_2 muss dabei nicht zwingend gelten $\mathcal{D}_2 = \mathcal{D}_1$, da für die Grob- und Feinausrichtung eventuell auf unterschiedliche Merkmale geachtet werden muss. So kann in einem multimodalen Fall die Grobausrichtung anhand von Segmentierungen (0-1 Bildern) der interessanten Bereiche erfolgen, während die Hauptregistrierung auf die gesamten Informationen in diesen Bereichen zurückgreift. Im ersten Fall ist eine geeignete Wahl das SSD, allerdings könnte auch für die Hauptregistrierung ein Distanzmaß wie z. B. NGF oder LSD gewählt werden. Die Wahl des Regularisierers \mathcal{S} hängt von den gestellten Eigenschaften an die Transformation ab. Häufig hat die Wahl der Nebenbedingungen c_i einen Einfluss darauf. Das resultierende Modell wird wiederum diskretisiert und optimiert. Wir erhalten dann die gesuchte Transformation φ^{opt} .

Zum Abschluss des Registrierungskapitels beschäftigen wir uns mit einem Optimierungsaspekt zur Reduzierung lokaler Minima und damit zur Erhöhung der Robust-

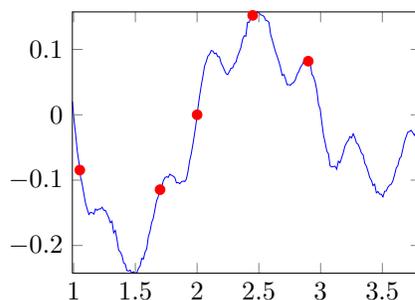


Abbildung 3.20.: Eine Funktion mit lokalen Minima ist zu sehen. Durch zusätzliches Rauschen wird die Anzahl dieser noch verstärkt. Mittels roter Punkte sind mögliche Startwerte für eine Optimierung markiert, die typischerweise alle in einem anderen lokalen Minimum landen.

heit, dem Multiskalen/Multilevel-Ansatz. Diesen verwenden wir sowohl in der Vorals auch in der Hauptregistrierung.

3.8. Multiskalen Ansätze in der Bildregistrierung

Ein Registrierungsalgorithmus sollte robust sein, d. h. die berechnete Lösung sollte möglichst immer dieselbe sein, unabhängig von der initialen Situation. In der Registrierung betrachten wir allerdings stark nicht-konvexe Funktionen. Nehmen wir Abbildung 3.20 als ein Beispiel hierfür. Die Funktion besitzt viele lokale Minima, unter anderem verursacht durch Rauschen. Offensichtlich wird ein Optimierungsverfahren aus Kapitel 2 für verschiedene Startwerte verschiedene Lösungen liefern. Um eine gewisse Robustheit gewährleisten zu können, müssen zusätzliche Techniken verwendet werden. Wir stellen dafür zwei weitbekannte Techniken vor, den Multiskalen- und den Multilevel Ansatz. Ersterer dient hauptsächlich der Robustheit, während der Zweite zusätzlich eine Reduktion der Datenmenge und damit eine Beschleunigung des Verfahrens mit sich bringt.

3.8.1. Grundlagen

Eine Möglichkeit, die Nicht-Konvexität der Zielfunktionen zu reduzieren, besteht darin, einen Multiskalen Ansatz zu nutzen. Grundlage dieses Ansatzes ist der SKALENRAUM. Er ist zu betrachten als eine Folge von Bildern, bei denen wir über einen SKALENRAUMPARAMETER Informationen ausblenden. Das Ausblenden der

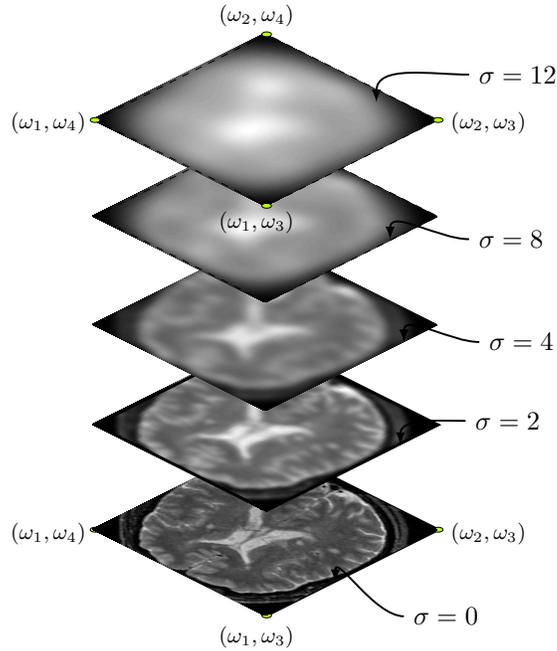


Abbildung 3.21.: Eine Gauß'sche Pyramide für eine Schicht eines MR-Bildes. Wir betrachten den Bildbereich $\Omega = (\omega_1, \omega_2) \times (\omega_3, \omega_4)$ auf fünf verschiedenen Skalen des Skalenraums. Der Skalenraumparameter σ steigt von unten nach oben mit den Werten $\sigma = 0, 2, 4, 8, 12$. Alle Bilder besitzen die Bildgröße $m = (176, 140)$

Informationen in den Bildern spiegelt sich im Distanzmaß wieder. Je weniger Information in den Bildern ist, desto glatter ist im Allgemeinen das Distanzmaß. Ein wichtiger Repräsentant der Skalenräume ist der GAUSS'SCHE SKALENRAUM. Die Folge der Bilder wird hierbei durch eine Faltung mit der Gauß Funktion

$$f(\mathbf{x}; \sigma) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{\|\mathbf{x}\|_2^2}{2\sigma^2}} \quad (3.49)$$

erzeugt. Die Standardabweichung $\sigma > 0$ nimmt dabei die Rolle des Skalenraumparameters ein. Je höher der Skalenraumparameter in diesem Fall ist, desto glatter werden die Bilder. Die Folge der Gauß-geglätteten Bilder wird auch Gauß'sche Pyramide genannt [2, 45]. Abbildung 3.21 veranschaulicht diese beispielhaft anhand von fünf Wahlen des Skalenraumparameters σ . Ein weiterer interessanter Skalen-

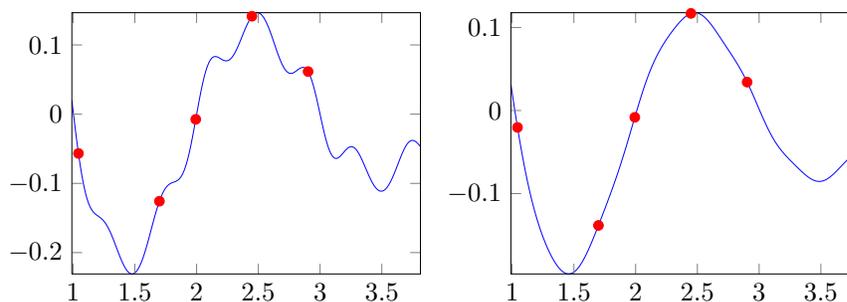


Abbildung 3.22.: Zwei Folglieder aus dem Gauß'schen Skalenraum für die verrauschte Funktion aus Abbildung 3.20. Dabei ist links eine kleinere Standardabweichung gewählt als rechts. Wir sehen, dass nach und nach lokale Minima ausgeblendet werden. Wir können allerdings nicht sicher gehen, selbst bei hohem Parameter, dass wir nach Durchführung des Multiskalen Ansatz immer im globalen Optimum landen. Der rechte rote Punkt als Startwert ist dafür ein Beispiel. Die Robustheit wird allerdings deutlich erhöht

raum hat seinen Ursprung in der Bildregistrierung von Gefäßbäumen [49]. Hierbei bestimmt der Skalenraumparameter die Dicke der noch sichtbaren Gefäße.

Die Idee der hier vorgestellten MULTISKALEN ANSÄTZE besteht darin, zuerst die Registrierung auf glatten Repräsentanten von \mathcal{T} , \mathcal{R} des Skalenraums durchzuführen und in weiteren Iterationen nach und nach mehr Informationen einzublenden. Dabei dient das Ergebnis des vorherigen als Startwert des aktuellen Schritts. Diese Strategie weist einige Vorteile auf. Durch die Glattheit in den ersten Iterationen sollte die Optimierung einfacher und das Risiko, in lokalen Minima gefangen zu werden, reduziert sein. Dadurch wird die Robustheit des Verfahrens deutlich erhöht. Optimieren wir danach auf den Bildern mit mehr Informationen/Details, erwarten wir durch einen guten Startwert zudem wieder weniger Iterationsschritte. Allerdings laufen die Berechnungen auf den verschiedenen Skalen, auf denen die Bilder alle dieselbe Auflösung besitzen. Ein Beispiel für den eindimensionalen Multiskalen Ansatz sehen wir in Abbildung 3.22.

3.8.2. Ein Multilevel Ansatz

Der Multiskalen Ansatz weist bereits viele Vorteile auf. Der MULTILEVEL ANSATZ setzt auf dem Multiskalen Ansatz auf und fügt diesem einen weiteren Vorteil hinzu. Betrachten wir dazu den Gauß'schen Skalenraum. Durch die Faltung mit einem

Gaußfilter blenden wir Informationen aus, wir wenden einen Tiefpassfilter an. Das Theorem von Shannon besagt nun, dass die verlustfreie Diskretisierung mit der Bandbreite des Signals einhergeht [75]. Daraus folgt, dass wir nach Anwendung des Gaußfilters eine gröbere Diskretisierung durchführen können, ohne Informationen dabei zu verlieren. Kleinere Datenmengen ergeben wiederum weniger Unbekannte in unserer Optimierung und schnellere Auswertungen der Zielfunktion. Auf den Leveln mit grober Diskretisierung, auf denen wir noch weit von einer Lösung entfernt sind, werden im Vergleich zu späteren Leveln viele Iterationsschritte notwendig. Durch die geringe Auflösung der Bilder und Funktion sind wir dennoch schnell. Auf den Leveln mit feiner Diskretisierung erwarten wir einen guten Startwert und damit wenig Iterationschritte zu höheren Kosten. Ein weiterer großer Vorteil des Ansatzes besteht darin, dass durch die grobe Auflösung auf den ersten Leveln zudem größere Deformationen möglich sind. Aufgrund der unterschiedlichen Diskretisierungen sind im Gegensatz zum Multiskalen Ansatz jedoch zusätzliche Schritte notwendig, die Restriktion und die Prolongation.

Restriktion und Prolongation Wir werden jetzt auf die Berechnung der Bilder von einem feinen auf ein grobes Level und das Überführen der Transformation von einem groben auf ein feines Level näher eingehen. Erstes wird RESTRIKTION, zweites PROLONGATION genannt.

Wir betrachten die Restriktion bezüglich eines Gauß'schen Skalenraums. Es gibt viele Möglichkeiten hierfür, wir haben folgenden Ansatz verfolgt. Betrachten wir beispielhaft das Templatebild \mathcal{T} im Bildbereich Ω . Es sei die diskrete Repräsentation \mathbf{T}^{fein} mit Bildgröße $m^{\text{fein}} = (m_1^{\text{fein}}, \dots, m_d^{\text{fein}})$ und zugehöriger Voxelgröße h^{fein} auf dem feinen Level gegeben. Wir wollen eine Restriktion um den Faktor zwei in alle Richtungen betrachten. Das Bild auf dem gröberen Level sei bezeichnet mit \mathbf{T}^{grob} , der Bildgröße $m^{\text{grob}} = (m_1^{\text{grob}}, \dots, m_d^{\text{grob}})$ und korrespondierender Voxelgröße h^{grob} . Eine Interpolation des Bildes auf dem zellzentrierten Gitter $X^{CC}(\Omega, m^{\text{grob}})$ würde nach dem Theorem von Shannon zu einem Informationsverlust führen. Da wir die Restriktion bezüglich des Gauß'schen Skalenraums durchführen, glätten wir jedoch vorher \mathbf{T}^{fein} mit der diskretisierten Gaußfunktion. Da die Gaußfunktion separabel ist, überführen wir die d -dimensionale Faltung in d eindimensionale Faltungen mit der Filtermaske

$$\mathbf{g} = \frac{1}{4} \begin{bmatrix} 1 & 2 & 1 \end{bmatrix}.$$

Eine nachfolgende Interpolation auf dem zellzentrierten Gitter $X^{CC}(\Omega, m^{\text{grob}})$ liefert das Bild

$$\mathbf{T}^{\text{grob}} = \text{inter}(\mathbf{T}, \bar{\mathbf{x}}^{cc}, 1)$$

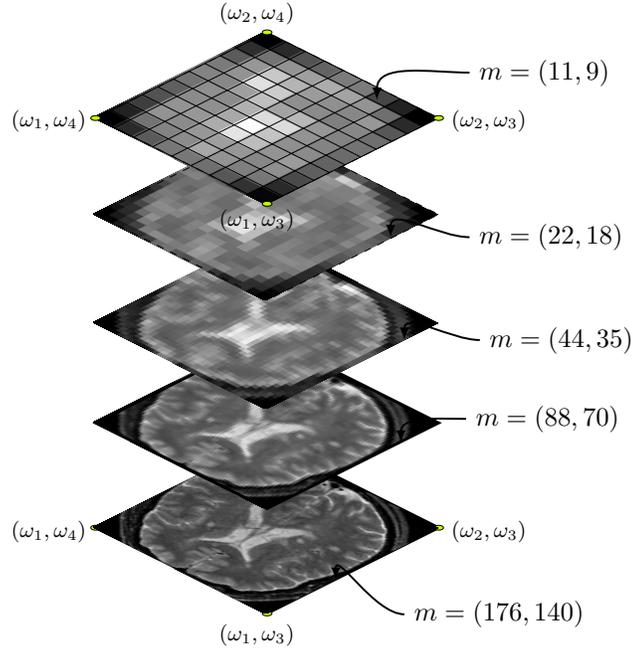


Abbildung 3.23.: Eine mittels Restriktion erzeugte Bildpyramide für ein Bild mit der Ausgangs-Bildgröße $m = (176, 140)$. Wir betrachten den Bildbereich $\Omega = (\omega_1, \omega_2) \times (\omega_3, \omega_4)$ in fünf verschiedenen Resolutionen. Die Auflösungen sind von unten nach oben vergrößert

auf dem groben Level mit der Bildgröße

$$m^{\text{grob}} = (m_1^{\text{grob}}, \dots, m_d^{\text{grob}}) = \left(\lfloor \frac{m_1^{\text{fein}}}{2} \rfloor, \dots, \lfloor \frac{m_d^{\text{fein}}}{2} \rfloor \right).$$

Wir nutzen hier die lineare Interpolation, da keine Ableitungen benötigt werden und das min – max-Prinzip eingehalten wird. Eine Veranschaulichung einer Multilevel-Pyramide ist in Abbildung 3.23 zu finden. Im Gegensatz zur Gauß'schen Pyramide (Abbildung 3.21) können wir eine zusätzliche Reduzierung der Bildauflösung erkennen.

Wir betrachten in dieser Arbeit die Prolongation im Zusammenhang mit der Bildregistrierung für Transformationen. Für die Restriktion der Bilder haben wir noch ein zellzentriertes Gitter verwendet. Bei den kommenden Registrierungsverfahren

verwenden wir allerdings Transformationen, die auf einem nodalen Gitter ausgewertet werden. Daher betrachten wir die Prolongation der Transformation auf einem nodalen Gitter. Sei diese Transformation auf grobem Level auf einem nodalen Gitter $X^n(\Omega, m^{\text{grob}})$ gegeben durch $\varphi^{\text{grob}} = (\varphi_1^{\text{grob}}, \dots, \varphi_d^{\text{grob}})$. Die Transformation φ^{fein} auf dem feineren Level kann nun durch komponentenweise lineare Interpolation der Verrückung $\mathbf{u}_i^{\text{grob}}$, $i \in \{1, \dots, d\}$ auf einem nodalen Gitter $X^n(\Omega, m^{\text{fein}})$ berechnet werden. Die prolongierten Verrückungen werden anschließend zu den Repräsentanten $\vec{\mathbf{x}}$ des nodalen Gitters $X^n(\Omega, m^{\text{fein}})$ addiert. Bei der zellzentrierten Prolongation muss zusätzlich auf den Rand geachtet werden. Der Schritt der Prolongation ist für uns nur bei der nichtparametrischen Registrierung notwendig, da rigide und affin-lineare Transformationen global mit den Parametern γ gegeben sind.

Die Prolongation kann auch beim Einsatz von Nebenbedingungen für diese zusätzlich notwendig sein. Die Prolongation muss hierbei dann sowohl für die Nebenbedingungen als auch für die Lagrange-Multiplikatoren durchgeführt werden.

3.9. Zusammenfassung

Übersicht: Grundlagen der Bildregistrierung		
Begriffe		
Begriff	Kontinuierlich	Diskret
Templatebild	\mathcal{T}	\mathbf{T}
deformiertes Templatebild	\mathcal{T}_φ	\mathbf{T}_φ
Referenzbild	\mathcal{R}	\mathbf{R}
Bildbereich	Ω	
Bildgröße		m
Voxelgröße		h
Zellzentriertes Gitter		$X^{cc}(\Omega, m)$
Nodales Gitter		$X^n(\Omega, m)$
Transformation	φ	$\boldsymbol{\varphi}$
Verrückungsfunktion	u	\mathbf{u}
Transformationsparameter	γ	
Distanzmaß	\mathcal{D}	\mathbf{D}
Regularisierer	\mathcal{S}	\mathbf{S}
Nebenbedingung	c	\mathbf{c}

Operatoren und Konstrukte	
Interpolationsoperator	$[\mathbf{T}_Y, d\mathbf{T}_Y] = \text{inter}(\mathbf{T}, Y, i), i \in \{1, 3\}$
Volumen	vol
Funktionaldeterminante	$\det(D\varphi)$

In diesem Kapitel haben wir die Grundlagen der Registrierung in einem diskretisieren-dann-optimieren Rahmen betrachtet. Sie stellen die Grundlagen für die Entwicklung problemspezifischer Registrierungsalgorithmen. Der Übersicht halber fassen wir die wichtigen Notationen noch einmal zusammen mit einer Gegenüberstellung der kontinuierlichen Version. Im Folgenden betrachten wir spezielle Anwendungen, die ohne das Einbringen von Zusatzwissen nicht zufriedenstellend bearbeitet werden können.

Teil II.
Anwendung

4. Registrierung Prä- und Postoperativer Leberdaten

Die erste Anwendung, die wir in dieser Arbeit betrachten wollen, beschäftigt sich mit der Registrierung prä- und postoperativer Leberdaten. Der interessante Aspekt an diesem Problem ist, dass ein Teil der postoperativen Leber nicht mehr vorhanden ist. Dieser wurde bei einer Resektion entfernt. Die Anwendung hat ihren Ursprung im BMBF-Projekt FUSION (**F**uture Environment for Gentle Liver **S**urgery Using **I**mage-Guided Planning and Intra-**O**perative **N**avigation). Das Projekt ist anzusetzen in der navigierten Leberchirurgie. Hierbei geht es unter anderem darum, von Tumor infiziertes Gewebe durch eine Resektion zu entfernen. Ziel von FUSION ist es aus präoperativ gewonnenen Volumendaten (CT, MRT) eine Operationsplanung zu erstellen und diese während der Operation nutzbar zu machen. Dabei entstand auch die Fragestellung, wie diese Operationsplanung umgesetzt wurde. Dazu sollen die präoperativen mit den postoperativen CT-Daten verglichen werden. Wir haben bereits in der Einführung gesehen, dass durch die positionellen und anatomischen Unterschiede eine spezielle Registrierung von Nöten ist. Im Folgenden werden wir einen Einblick in die Anatomie der Leber und die Leberresektion vornehmen und darauf aufbauend uns genauer mit der Motivation des Problems beschäftigen. Folgend wird die Datenlage und Vorverarbeitung vorgestellt. Im Anschluss werden wir verschiedene Verfahren zur Vorregistrierung der Daten betrachten, um eine gute Ausgangsposition für eine abschließende nichtlineare Registrierung zu gewährleisten. Für die nichtlineare Registrierung betrachten wir zwei unterschiedliche Ansätze. Zum einen werden wir eine punktweise Volumenerhaltung der Leber und zum anderen eine Längenerhaltung der Gefäße fordern.

4.1. Grundlagen der Leberresektion

Metastasenbildung in der Leber ist ein häufig auftretendes Problem. Neben dem hepatozellulären Karzinom bilden sich, verursacht durch die Filterfunktion der Leber, zusätzlich viele Lebermetastasen. Ein chirurgischer Eingriff, bei dem der Tumor entfernt wird, ist häufig die einzig kurative Therapie. Hierbei sprechen wir von der



Abbildung 4.1.: Verschiedene Gefäßbäume der Leber. Wir sehen von links nach rechts die Gefäßbäume PV, HV und HA, gewonnen aus einer CT-Aufnahme.

Leberresektion. Für ein allgemeines Verständnis der Begrifflichkeit und Argumentationen, beschreiben wir im Folgenden kurz die Anatomie der Leber sowie einige Grundlagen der Leberresektion. Sofern nicht anders verwiesen, halten wir uns bei den Beschreibungen hauptsächlich an [14, 81, 82].

4.1.1. Leberanatomie

Die Leber liegt hinter den Rippen im Bereich des rechten Oberbauchs. Sie ist durch das Ligamentum coronarium hepatis mit der Unterfläche des Zwerchfells verwachsen, wodurch ihre Lage stark von der Atmung beeinflusst wird. Sie weist eine glatte, homogene, rotbraune Oberfläche auf. Von der Vorderseite (*facies diaphragmatica*) wird sie in einen linken und rechten Leberlappen durch das Ligamentum falciforme getrennt. Neben den beiden Hauptlappen sind von der Unterseite (*facies visceralis*) zwei weitere sichtbar, der Lobus Quadratus und der Lobus caudatus. Auf dieser Seite treten auch LEBERATERIEN (*Ateria hepatica propria*, HA) und die PFORTADERN (*Vena portae hepatis*, PV) durch die LEBERPFORTE (*porta hepatis*) ein bzw. aus. Auch die mit der Gallenblase verbundenen Gallengänge (*ductus hepaticus*) treten dort aus. Volumen, Form und Gewicht weisen eine hohe Variabilität über die Population hinweg auf. Einen wichtigen Bestandteil der Befestigung der Leber stellt die VENA CAVA INFERIOR (VCI). Zum einen geschieht dies durch die Lebervenen, die dort austreten und zum anderen durch das Bindegewebe. Für die Resektion sind wir besonders an der sogenannten funktionellen Anatomie interessiert, die auf der Struktur der Gefäßsysteme und damit der Versorgung beruht. Die verschiedenen Gefäßbäume sind in Abbildung 4.1 dargestellt.

Funktionelle Anatomie Die funktionelle Anatomie hat ihren Ursprung in den Arbeiten von Couinaud [22]. Startpunkt der Unterteilung sind dabei die durch die Leberpforte eintretenden Gefäße (PV, HA, Gallengänge), die innerhalb der Leber von einer bindegewebigen Abspaltung umgeben sind. Diese verzweigen sich bei ihrem Verlauf durch die Leber und sorgen somit für voneinander unabhängige Lebersegmente. Dabei verzweigt sich der eintretende Ast zunächst in zwei größere Äste. Diese teilen die Leber in die funktionellen Teile Pars hepatis sinistra und Pars hepatis dextra. Diese Äste teilen sich in den beiden Teilen weiter auf und werden zusätzlich von den Lebervenen voneinander getrennt. Dabei teilt sich die Lebervene von der VCI in drei Äste, die sich dann wiederum verzweigen. Abbildung 4.2 veranschaulicht die daraus resultierende Einteilung in acht Lebersegmente. Durch die relative Unabhängigkeit der Segmente bildet diese Einteilung eine sehr gute Voraussetzung zur Entnahme von einzelnen Leberanteilen. Zudem besitzt die Leber ein hohes Regenerationsvermögen, das eine Wiederherstellung der Funktionalität erlaubt.

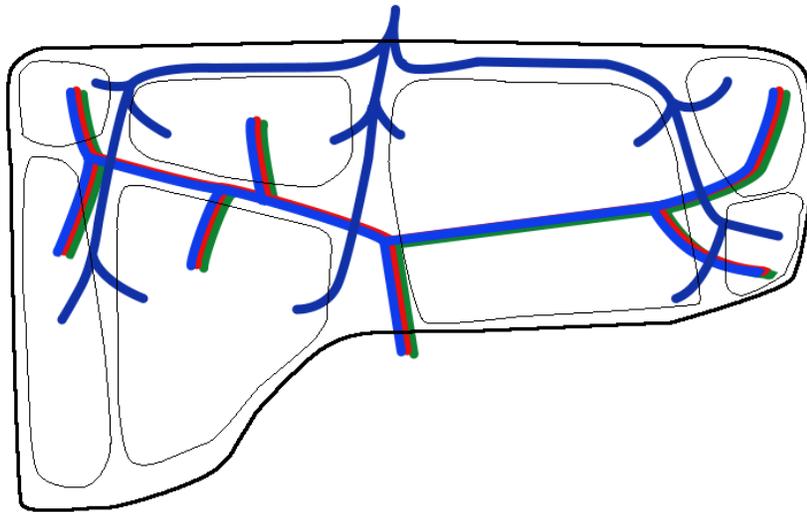


Abbildung 4.2.: Schematische Skizze der Einteilung der Lebersegmente nach Couinaud, frei nach [82].

4.1.2. Durchführung der Leberresektion

Die Durchführung der Leberresektion wird durch zwei konkurrierende Faktoren beeinflusst. An erster Stelle soll der Tumor samt eines notwendigen Sicherheitsbereichs

entfernt werden. Dieser Sicherheitsbereich darf allerdings nicht zu groß gewählt werden, da möglichst viel Lebervolumen erhalten bleiben soll. Bei einer Leber, die nicht vorgeschädigt ist, kann das Restvolumen bei ca. 20-30% liegen, um später wieder annähernd volle Funktionalität zu gewährleisten. Zudem soll das Patiententrauma, z. B. der Blutverlust, minimiert werden. Um dies möglichst gut zu bewerkstelligen wird präoperativ, basierend auf CT-Volumendaten, eine mehr oder weniger umfangreiche Operationsplanung durchgeführt. Grundlage der Leberresektionen ist die Aufteilung der Leber nach Couinaud (Abschnitt 4.1.1). Allerdings hat sich gezeigt, dass diese nicht regelhaft ist, sondern eine hohe Variabilität von Patient zu Patient aufweist. Um die Leberresektionen möglichst gut planen zu können, werden durch Kenntnis der genauen Gefäßstrukturen (PV, HV, HA) die unabhängigen Segmente patientenindividuell bestimmt. Basierend auf diesen Informationen wird geprüft, ob die Resektionen durchgeführt werden kann, und nach den oben genannten Faktoren geplant. Um während der Operation auch an schwer zugängliche Orte zu kommen, muss in den meisten Fällen eine Mobilisierung der Leber stattfinden. Dies wird durch die Durchtrennung der betreffenden Ligamente bewerkstelligt. Dies und die Homogenität der Leber erschwert die Orientierung des Chirurgen. Um sich dennoch orientieren zu können wird Ultraschall (US) intraoperativ genutzt. Durch eine Registrierung der intraoperativen US- und präoperativen CT-Daten (siehe z. B. [48]), wird der Chirurg während des Eingriffes unterstützt. Nach der Operationen treten Schwellungen auf, die in der Regel nach wenigen Tagen wieder deutlich abklingen.

4.2. Motivation

Im vorangegangenen Abschnitt haben wir einen kurzen Einblick in die Durchführung einer Leberresektion samt Planung erhalten. Eine Fragestellung steht allerdings noch im Raum:

”Wie gut hat das Zusammenspiel zwischen präoperativer Operationsplanung und deren Durchführung funktioniert?”

Zum einen impliziert diese Fragestellung, wie genau die Planung des Eingriffs umgesetzt wurde. Also, ob der gesamte Tumor entfernt wurde und aber auch, ob die betreffenden Segmente korrekt entfernt wurden und somit die geplante Funktionalität erhalten ist. Andererseits ist mit dieser Fragestellung die Erörterung der Effektivität des Systems zur Unterstützung des Chirurgen verbunden. Um einen punktuellen Vergleich der prä- und postoperativen Lebervolumen und deren Segmente und damit eine Validierung des Eingriffs zu erlauben, ist eine Registrierung

der beiden Volumina notwendig. Dabei ist vor Allem eine genaue Ausrichtung der Gefäßbäume wichtig, da die Einteilung der Segmente auf ihnen beruht. Anstatt Wert auf die komplette Leber zu legen, werden wir in dieser Arbeit deshalb uns vorrangig mit der Registrierung von Gefäßbäumen beschäftigen.

Eine spezielle Problematik bei dieser Problemstellung ist das Vorhandensein von fehlenden Korrespondenzen, da in der postoperativen Aufnahme der resektierte Teil der Leber fehlt. Diese Problematik spiegelt sich sowohl in der Vor- als auch in der Hauptregistrierung wider und macht sich bei Standardverfahren in einem ungewollten "Nachwachsen" der Leber bemerkbar. Durch die Mobilisierung der Leber und Bewegung im Abdomen durch Atmung und Herzschlag, sind zudem starke nichtlineare Deformationen präsent. Eine rigide Registrierung wird daher keine zufriedenstellenden Ergebnisse liefern. Wir werden aus diesem Grund im Folgenden verschiedene Modelle vorschlagen, die diese Probleme einbeziehen.

4.3. Stand der Technik

Zur Registrierung von prä- und postoperativen Datensätzen findet man nur wenige Ansätze in der Literatur. Zu unserer besten Kenntnis sind bisher nur zwei Ansätze im Bereich der Validierung der Leberresektion vorgeschlagen. Dumpuri et al. [26] nutzen ein rigides Deformationsmodell und registrieren die Leberoberflächen mittels der Iterative Closest Point- Methode (ICP, [6]). Die Validierung basiert auf dem Vergleich des geplanten und des tatsächlichen Restvolumens. Durch die rigide Registrierung erlangen sie einen visuellen Eindruck ihrer Berechnungen. Wie wir bereits oben gesehen haben, ist die Annahme eines rigiden Deformationsmodells schwierig. Die nicht-linearen Deformationen und damit verbundenen Oberflächenänderungen können nicht eingefangen werden. Dies wird auch in den Visualisierungen der Ergebnisse deutlich. Ein weiteres Problem ist, dass der Vergleich der Volumina, durch den globalen Standpunkt, eine sehr grobe Validierung der Resektion ist. Lange et al. [62] schlagen daher vor, die Gefäßbäume als Anhaltspunkt zu nehmen. Wie bereits in Abschnitt 4.1 beschrieben spielen diese eine wichtige Rolle bei der funktionalen Einteilung der Leber in Segmente. Interaktiv werden korrespondierende Verzweigungspunkte und Gefäßsegmente (mit anisotroper Toleranz) als Landmarkenpaare gewählt. In ihrem Paper vergleichen sie interpolierende und approximierende landmarken-basierte Verfahren. Als Basisfunktionen dienen hier die Thin-Plate-Splines (TPS, [8]) bzw. Gaussian-Elastic-Body-Splines (GEBS, [95]). Zu starkes Nachwachsen der Leber wird durch Glattheitsannahmen unterdrückt, die durch die Verwendung der Basisfunktionen vorhanden sind. Um aussagekräftige Ergebnisse zu erzielen, wird das Setzen von zwölf Landmarken vorgeschlagen. Unter Verwendung eines automatischen Verfahrens zum Detektieren der Landmarken entfällt der

Nachteil des interaktiven Setzens. Nicht zu vernachlässigen ist jedoch der Fakt, dass die Landmarken die einzige treibende Kraft der Registrierung sind. Ein Ansatz aus dem Bereich der Resektion im Gehirn ist vorgestellt von Chitphakdithai und Duncan [17]. Sie kombinieren eine nicht-lineare Registrierung mit einer Segmentierung. Die Segmentierung liefert ihnen z. B. Informationen, ob es sich um einen resezierten Bereich handelt. Dazu wird ein Maximum-A-Posteriori (MAP) Rahmen mit einem Expectation Maximization (EM) Algorithmus genutzt, um das Deformationsfeld und die Segmentierungen zu berechnen. Der Ansatz liefert gute Resultate, jedoch müssen für das Distanzmaß Standardabweichungen angegeben werden. Ein genereller Ansatz für Registrierung mit fehlenden Daten, ebenfalls basierend auf einer Kombination von Registrierung und Segmentierung, wurde von Periaswamy und Farid [77] eingeführt. Risholm et al. [78] versuchen ebenfalls die Resektionsregionen während des Registrierungsschrittes zu identifizieren. Die Regionen beeinflussen wiederum das Registrierungsverhalten.

Ziel soll im Folgenden ein robuster Algorithmus sein, der sowohl nichtlineare Deformationen zulässt, als auch die fehlenden Korrespondenzen einbezieht. Dabei soll am Ende eine zufriedenstellende Ausrichtung der Gefäßbäume stehen. Da das Ergebnis der Registrierung nicht zeitkritisch ist, verzichten wir bei der Umsetzung der Ideen, im Rahmen dieser Arbeit, auf Implementierungsdetails, die einen zeitlichen Gewinn mit sich bringen, ohne dabei das Verfahren an sich zu verändern. Wir starten mit einer robusten Methode zur Vorregistrierung und werden zwei nichtlineare Ansätze zur Lösung der beschriebenen Problematik als Hauptregistrierungsschritt vorstellen.

4.4. Datenlage

Sowohl prä- als auch postoperative CT-Aufnahmen werden vom Patienten erstellt. Diese Daten durchlaufen mehrere Vorverarbeitungsschritte. In diesem Abschnitt wollen wir die daraus erlangten Datensätze vorstellen, die im Folgenden genutzt werden, um unsere Problemstellung in Angriff zu nehmen.

Grundlage aller verfügbaren Datensätze sind prä- und postoperative CT-Aufnahmen des Abdomens auf dem eine Region-Of-Interest (ROI) definiert wird, in der die Leber liegt (siehe Abbildung 4.3). Zu diesen wurden durch MEVIS Segmentierungen der Leber (Abbildung 4.4) sowie der unterschiedlichen Gefäßbäume berechnet. Auch eine Planung der Resektion liegt vor. Zu den einzelnen Gefäßbäumen existieren zudem Daten in Baumstruktur. Diese beinhalten unter anderem Informationen zu Gefäßlängen, -radii und -volumina. Im Folgenden wollen wir drei Datensätze ge-



Abbildung 4.3.: Eine Schicht aus der axialen Schnittebene des prä- (links) und postoperativen (rechts) Datensatz ist dargestellt

nauer vorstellen. Diese wurden bereitgestellt von Dr. med. Thilo Schumacher und Dr. med. Christoph Logge vom Universitätsklinikum Hamburg-Eppendorf.

Testdatensätze Die hier vorgestellten Datensätze wollen wir mit Datensatz 1-3 benennen. Im späteren Verlauf der Arbeit werden wir nur die Bezeichnungen der Datensätze nutzen, ohne weitere Informationen aufzuführen. Wir werden die Bilder in der ROI vorstellen. Dazu betrachten wir die segmentierte Leber, die Gefäßbäume PV und HV sowie das Skelett des Gefäßbaums PV, auf das wir in Abschnitt 4.7.2 genauer zu sprechen kommen. Zudem haben wir zu den betrachteten Gefäßbäumen Landmarkeninformationen gegeben, die ebenfalls angeführt werden.

Datensatz 1 Die kompletten präoperativen CT-Daten des ersten Datensatzes haben eine Bildgröße von $m^{\text{orig}} = (512, 512, 216)$ bei einer dazugehörigen Voxelgröße von $h = (0.68, 0.68, 1)$ mm. Die ROI weist eine Bildgröße von $m = (346, 296, 196)$ auf. Für die postoperativen Daten ist $m^{\text{orig}} = (512, 512, 241)$, ebenfalls bei einer Voxelgröße von $h = (0.68, 0.68, 1)$ mm. Die Bildgröße der ROI ist $m = (261, 209, 201)$. Die Ausgangsdaten werden in Abbildung 4.5 vorgestellt. Des Weiteren sind Landmarkenpaare gegeben. Für den Gefäßbaum PV sind 14, für HV 5 korrespondierende Punkte angegeben. Zur Visualisierung der Lage dieser Punkte sei auf Abbildung 4.6 verwiesen. Zudem ist die Skelettierung der Gefäßbäume der postoperativen Leber gegeben. Für den in Abschnitt 4.7 beschriebenen Ansatz ist keine Skelettierung



Abbildung 4.4.: Lebersegmentierungen der prä- (links) und postoperativen (rechts) CT-Daten sowie die geplante Resektions (Mitte). Dabei soll der dunkle Teil der Leber entfernt werden



Abbildung 4.5.: Datensatz 1: Prä- (links) und postoperative (Mitte) maskierte Leberdaten im ROI. Rechts ist die initiale Situation vor der Registrierung zu betrachten

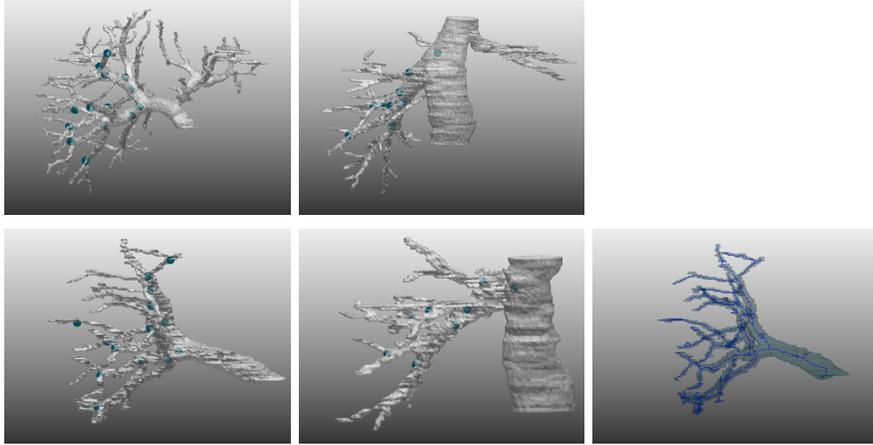


Abbildung 4.6.: Datensatz 1: Visualisierung der Gefäßbäume mit den gesetzten Landmarken in den präoperativen (oben) und postoperativen (unten) Daten. Links ist der Gefäßbaum PV, in der Mitte HV und rechts das Skelett der PV zu sehen

des präoperativen Gefäßbaumes notwendig. Für den Gefäßbaum PV besteht dieser beispielhaft aus 974 Punkten.

Datensatz 2 Die präoperativen CT-Daten des zweiten Datensatzes liegen in einer Bildgröße von $m^{\text{orig}} = (512, 512, 241)$ bei einer Voxelgröße von $h = (0.71, 0.71, 1)$ mm vor. Die ROI hat eine Bildgröße von $m = (398, 330, 253)$. Die Bildgröße der postoperativen Daten ist $m^{\text{orig}} = (512, 512, 301)$ mit $h = (0.68, 0.68, 1)$ mm. Die Bildgröße des ROI ist hier $m = (361, 299, 210)$. Die segmentierten Lebern und deren Orientierung zueinander sind in Abbildung 4.7 zu betrachten. In Abbildung 4.8 sind die Segmentierungen der Gefäßbäume mit den gesetzten Landmarken und das Skelett der postoperativen PV visualisiert. Im Gefäßbaum PV sind fünf, in HV neun Landmarken gesetzt. Das Skelett ist über 673 Punkte gegeben.

Datensatz 3 Dieser Datensatz hat präoperativ die Bildgröße $m^{\text{orig}} = (512, 512, 206)$ mit Voxelgröße $h = (0.74, 0.74, 1)$ mm und einer ROI der Größe $m = (297, 236, 188)$. Die postoperative Aufnahme weist eine Bildgröße von $m^{\text{orig}} = (512, 512, 171)$ mit Voxelgröße $h = (0.68, 0.68, 1)$ mm auf. Der ROI hat dabei die Bildgröße $m = (272, 270, 213)$. Abbildung 4.9 zeigt die initiale Situation der Daten. Auch für diesen Datensatz sind Landmarken für die Gefäßbäume PV und HV gegeben, die aus fünf

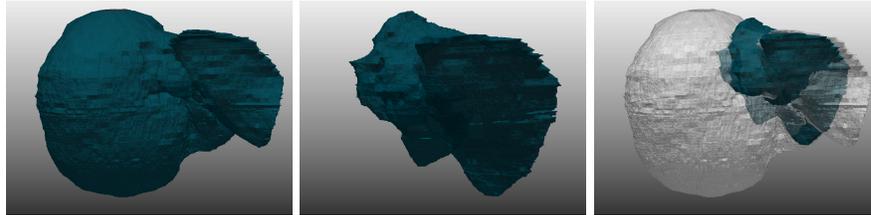


Abbildung 4.7.: Datensatz 2: Prä- (links) und postoperative (Mitte) maskierte Leberdaten im ROI. Rechts ist die initiale Situation vor der Registrierung zu betrachten

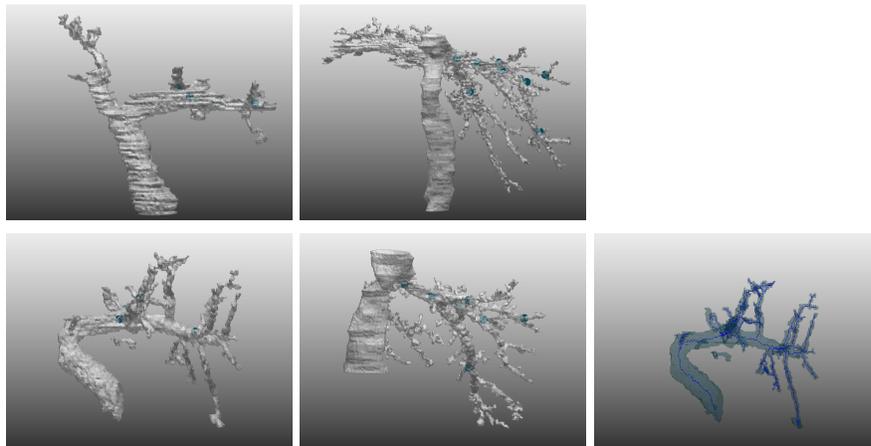


Abbildung 4.8.: Datensatz 2: Visualisierung der Gefäßbäume mit den gesetzten Landmarken in den präoperativen (oben) und postoperativen (unten) Daten. Links ist der Gefäßbaum PV, in der Mitte HV und rechts das Skelett der PV zu sehen



Abbildung 4.9.: Datensatz 3: Prä- (links) und postoperative (Mitte) maskierte Leberdaten im ROI. Rechts ist die initiale Situation vor der Registrierung zu betrachten

bzw. vier Paaren bestehen. Die Verteilung der Landmarken kann in Abbildung 4.10 inspiziert werden, zusammen mit dem Skelett des postoperativen Gefäßbaums PV. Das diskrete Skelett ist über 612 Punkte gegeben.

Kommen wir nun zu den essentiellen Punkte der Registrierung. Nach dem Registrierungsrahmen aus Abschnitt 3.7 beinhaltet dies eine Vorverarbeitung, eine Vorregistrierung und eine (nichtlineare) Haupt-Registrierung. Die Vorverarbeitung nehmen wir als bereits gegeben hin und beschäftigen uns im Folgenden mit den letzten beiden Punkten.

4.5. Vorregistrierung

Wie bereits in Abschnitt 3 erwähnt, ist eine gute Vorregistrierung notwendig, um einen geeigneten Startpunkt für die nachfolgende nichtlineare Registrierung zu gewährleisten. Häufig ist eine rigide intensitäts-basierte Registrierung das Mittel der Wahl. Im Falle der Registrierung einer prä- und postoperativen Leber ist die Wahrscheinlichkeit in ein ungewolltes lokales Minimum dabei zu kommen höher als bei Daten, die sich anatomisch kaum unterscheiden. Abbildung 4.11 zeigt ein Beispiel, in dem die postoperative Leber in ihrer initialen Position ungünstig liegt. Offensichtlich wird diese Problematik umso deutlicher, je größer das resezierte Volumen ist. Im Folgenden werden wir eine robuste Methode vorstellen, um einen geeigneten Startwert für die nachfolgende nichtlineare Registrierung zu gewährleisten.

4.5.1. Schwerpunktregistrierung

Ein Standardansatz um eine initiale Positionierung zu berechnen, ist die Schwerpunktsregistrierung. Hierbei wird der Translationsvektor $\mathbf{t} = (t_1, \dots, t_d)^T$ gesucht,

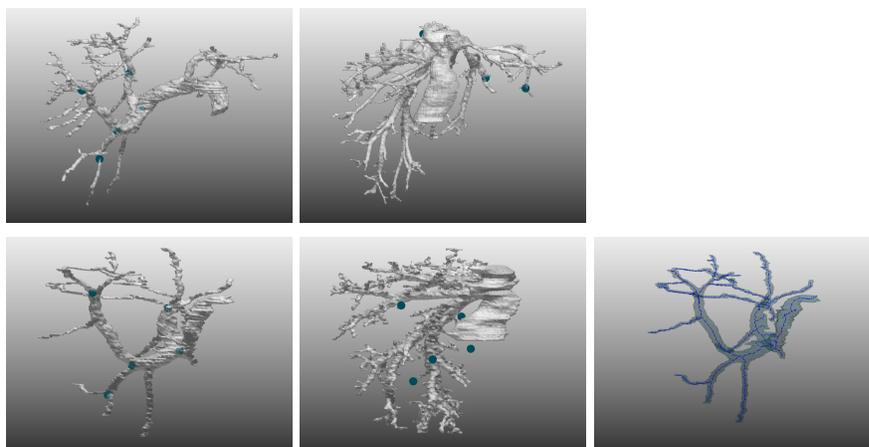


Abbildung 4.10.: Datensatz 3: Visualisierung der Gefäßbäume mit den gesetzten Landmarken in den präoperativen (oben) und postoperativen (unten) Daten. Links ist der Gefäßbaum PV, in der Mitte HV und rechts das Skelett der PV zu sehen

so dass die Schwerpunkte der Bilder übereinander liegen. Der Schwerpunkt eines Bildes \mathcal{T} auf Ω ist dabei über

$$\mathbf{x}_s = \frac{1}{M_0} (M_1, \dots, M_d)^T \quad (4.1)$$

mit den Momenten ersten und nullten Grades

$$M_i = \int_{\Omega} x_i \mathcal{T}(\mathbf{x}) \, d\mathbf{x}, \quad i = 1, \dots, d, \quad \text{bzw.} \quad M_0 = \int_{\Omega} \mathcal{T}(\mathbf{x}) \, d\mathbf{x} \quad (4.2)$$

definiert [53]. Bei dem betrachteten Fall ist allerdings nicht anzunehmen, dass der Schwerpunkt der präoperativen mit dem der postoperativen Leber übereinstimmt. Die Datensätze sind aus anatomischer Sicht zu unterschiedlich und wir können nicht davon ausgehen, dass an allen Seiten die gleichen Anteile der Leber entnommen wurden, so dass der Schwerpunkt der prä-operativen Leber bei der post-operativen erhalten bleibt. Dieser Vorregistrierungsschritt kann somit im Allgemeinen keinen vernünftigen Startwert für die nachfolgende intensitätsbasierte Registrierung ermöglichen.

Ergebnisse der Schwerpunktregistrierung Abbildung 4.12 zeigt die Anwendung des Verfahrens, inklusive einer nachfolgenden intensitäts-basierten rigidem Registrierung auf den Volumendaten, und demonstriert das beschriebene Problem. Besonders bei Datensatz 2 treten starke Probleme auf. Die postoperative Leber wird

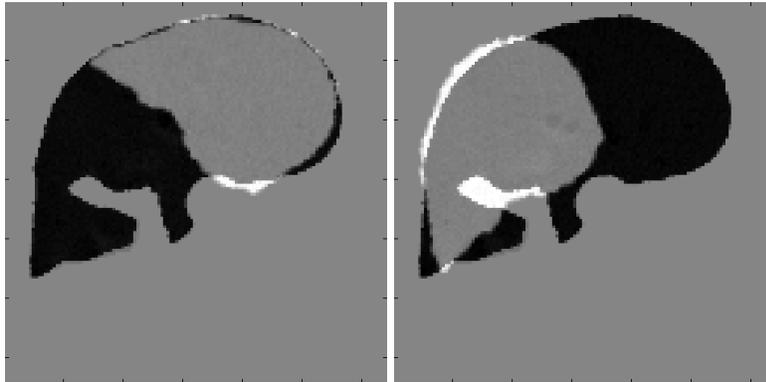


Abbildung 4.11.: Die Abbildung zeigt zwei rigide intensitäts-basierte Vorregistrierungen mit verschiedenen Ausgangspunkten für einen synthetischen Fall der Prä-Post-Registrierung. Es sind die Differenzbilder dargestellt. Links ist der gutmütige Fall zu sehen. Hier ist die initiale Lage bereits ähnlich der angestrebten Lage. Die Registrierung liefert somit einen guten Ausgangswert für die nachfolgende nicht-lineare Registrierung. Im anderen Fall befindet sich die post-operative Leber an einer falschen Position (rechts). Die rigide Registrierung liefert eine Transformation in ein ungewolltes lokales Minimum. Eine nichtlineare Registrierung im Anschluss wird fehlschlagen und den Ausgangsfehler nicht ausgleichen können.

nicht an den korrespondierenden Ort der präoperativen Leber transformiert. Die nachfolgende nichtlineare Registrierung kann nicht erfolgreich werden. Auch für die anderen beiden Fälle sind die Ergebnisse nicht überzeugend. Der Ansatz eignet sich nicht für die vorliegende Applikation. Indem man sich allerdings auf Bereiche konzentriert, die in beiden Datensätzen vorhanden sein müssen, kann man diese Problematik aufweichen. Es sei angemerkt, dass dieser Schritt auch als Einbringen von Zusatzwissen interpretiert werden kann.

4.5.2. VCI-Vorregistrierung

Einer dieser Bereiche ist die VCI (siehe Abschnitt 4.1). Aufgrund ihrer essentiellen Funktion, ist diese immer in den post-operativen CTs zu finden. Auf den Segmentierungen der VCI kann eine Schwerpunktsregistrierung durchgeführt werden. Im Anschluss daran sollte eine rigide Registrierung auf den segmentierten Leber-Daten geschehen, da dort mehr Informationen vorliegen. Durch vorherige

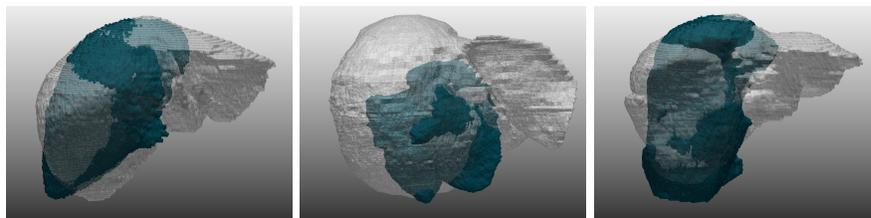


Abbildung 4.12.: Ergebnisse der Schwerpunktregistrierung bei Anwendung auf die drei Testdatensätze. Bei Testdatensatz 1 (links) und 3 (rechts) wird ein Ergebnis berechnet, das eine grobe, plausible Ausrichtung der Daten ermöglicht. Allerdings passt auch hier die Orientierung nur bedingt. In Datensatz 2 (Mitte) kommt es zu einem völlig ungewollten Ergebnis. Unterschiedliche Leberbereiche werden überlagert. Es wird ein lokales Minimum des Distanzmaßes angenommen, welches auf falschen Korrespondenzen der Leberstrukturen beruht

Schwerpunktausrichtung der VCI werden unplausible lokale Minima vermieden. Die Übereinanderlagerung der Schwerpunkte bedeutet noch nicht, dass die Gefäße auch die richtige Orientierung besitzen. Problematisch ist der Fall, wenn die Gefäße der beiden Aufnahmen in unterschiedliche Richtungen zeigen. Wir gehen allerdings davon aus, dass der Patient mit der gleichen Orientierung im CT liegt, d. h. die Kopfposition ist fest. Damit sollte dieses Problem ausgeräumt sein.

Ergebnisse der VCI-Vorregistrierung Abbildung 4.13 zeigt das Ergebnis dieses Zweischritt-Verfahrens. Die Robustheit des Verfahrens wird im Gegensatz zur einfachen Schwerpunktausrichtung deutlich erhöht. Auch Datensatz 2 wird nun grob ausgerichtet. Ein Problem bei diesem Ansatz ist allerdings noch, dass bei der Konzentration auf die VCI nur Informationen am Rande der Leber genutzt werden. Für unser Ziel einer möglichst guten Ausrichtung der Gefäßbäume sind wir jedoch an einer präzisen Ausrichtung der Gefäßbäume PV, HV und HA interessiert, die sich durch die gesamte Leber ziehen.

4.5.3. Rigide Landmarkenregistrierung

Wie wir gesehen haben, ist die Leber von verschiedenen Gefäßbäumen durchzogen. Bei einer Deformation der Leber werden die Gefäße mit deformiert, gleich bleibt jedoch deren Struktur. Auch die Lage innerhalb der Leber ist relativ stabil. Dies wollen wir uns im Folgenden für die Registrierung zu Nutze machen.

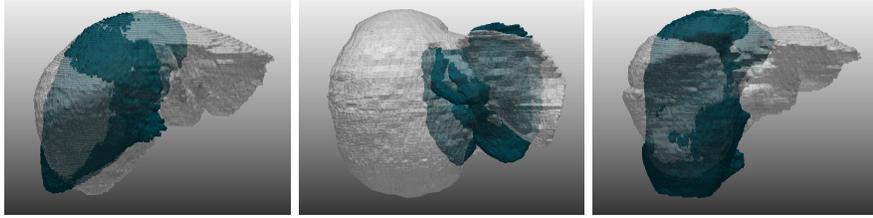


Abbildung 4.13.: Ergebnisse der VCI-Vorregistrierung für die drei Testdatensätze. Alle Fälle führen zu einer plausiblen Vorregistrierung. Im Gegensatz zu der einfachen Schwerpunktregistrierung wird auch der Testdatensatz 2 (Mitte) vernünftig ausgerichtet. Testdatensätze 1 (links) und 2 (rechts) verhalten sich gleich (vgl. Abbildung 4.12)

Die Gefäßverzweigungen sollen uns als Anhaltspunkte für die komplette Deformation dienen. Bei der bisherigen Vorregistrierung wurde bemängelt, dass wir hauptsächlich Wissen außerhalb bzw. am Rand der Leber einbeziehen. Haben wir jedoch Korrespondenzen für eine gewisse Anzahl von Verzweigungspunkten, so können wir diese für eine initiale rigide Registrierung nutzen. Durch die genauere Ausrichtung der Gefäße, wird die Anzahl der lokalen Minima reduziert, die sonst durch Übereinanderlagerung unterschiedlicher Gefäße hervorgerufen werden. Zudem können diese auch der im Anschluss stattfindenden nichtlinearen Registrierung als Zusatzwissen dienen (vgl. Abschnitt 3.6.2). Im diesem Abschnitt wollen wir uns jedoch nicht mit der Korrespondenzsuche der Verzweigungspunkte beschäftigen, sondern nehmen an, dass diese bereits gegeben sind. Im Ausblick dieses Kapitels (Abschnitt 4.8) werden wir noch einmal Bezug auf Möglichkeiten einer automatisierten Korrespondenzsuche nehmen.

Seien nun n Korrespondenzen zwischen Verzweigungspunkten gefunden. Diese korrespondierenden Punkte werden als LANDMARKEN bzw. LANDMARKENPAARE bezeichnet. Ziel ist es, eine Transformation zu finden, die die korrespondierenden Landmarken so gut wie möglich aufeinander abbildet. Wir sprechen hierbei von einer Landmarkenregistrierung. Im Folgenden werden wir dafür notwendige Notation für diesen Abschnitt einführen und den Lösungsweg kurz beschreiben.

Notation Seien mit $\mathbf{r}^i = (r_1^i, \dots, r_d^i)^T \in \mathbb{R}^d$, $i = 1, \dots, n$ die Landmarkenkoordinaten im Referenzbild \mathcal{R} und analog mit $\mathbf{t}^i = (t_1^i, \dots, t_d^i)^T \in \mathbb{R}^d$, $i = 1, \dots, n$ die korrespondierenden Landmarkenkoordinaten im Templatebild \mathcal{T} gegeben. In der Matrix $\mathbf{R} \in \mathbb{R}^{n \times d}$ stehen zeilenweise die Koordinaten \mathbf{r}^i . Zudem sei $\mathbf{t} \in \mathbb{R}^{dn}$ gegeben durch $\mathbf{t} = (t_1^1, t_1^2, \dots, t_d^n)^T$, also einer komponentenweisen Aneinanderreihung

der Landmarkenkoordinaten des Templatebildes.

Problem Es soll eine rigide Transformation φ berechnet werden, die die korrespondierenden Punkte räumlich so gut wie möglich übereinander lagert, d. h.

$$\min_{\varphi} \sum_{i=1}^n \|\varphi(\mathbf{r}^i) - \mathbf{t}^i\|_2^2. \quad (4.3)$$

Die Deformation wird dabei auf die Landmarken des Referenzbildes angewendet, da der Urbildbereich Ω_R auf Ω_T abgebildet wird. Die Formulierung ist als Approximationsproblem aufgefasst, da wir mehr Bedingungen haben als Freiheitsgrade existieren. Dies ist den nichtlinearen Deformationen der Leber zwischen den Aufnahmen geschuldet, durch die keine exakte rigide Transformation im Allgemeinen gefunden werden kann (vgl. [66]). Im Falle der rigiden Registrierung ist die Deformation φ parameterisierbar (siehe Abschnitt 3.4). Gleichung (4.3) können wir dann schreiben als

$$\min_{\mathbf{q}} \|\mathbf{Q}\mathbf{q}(\gamma) - \mathbf{t}\|^2 \quad (4.4)$$

mit

$$\mathbf{Q} = \mathbf{I}_d \otimes (\mathbf{R} \ \mathbf{1}), \quad (4.5)$$

wobei $\mathbf{1} = (1, \dots, 1)^T \in \mathbb{R}^n$. Der Vektor $\mathbf{q}(\gamma)$ ist gewählt wie bei den rigiden Transformationen beschrieben (Abschnitt 3.4). Dadurch stehen in $\mathbf{Q}\mathbf{q}(\gamma)$ die zu \mathbf{t} korrespondierenden Elemente der transformierten Koordinaten der Referenzlandmarken. Wir können keine direkte Lösung des Problems angeben. Die Least-Squares-Form des Problems ermöglicht die Anwendung des Gauß-Newton-Verfahrens.

Ergebnisse der landmarkenbasierten Vorregistrierung Das Verfahren testen wir an den drei Testdatensätzen. Dazu werden Landmarken auf dem Gefäßsystem PV genutzt. Die Ergebnisse sind in Abbildung 4.14 visualisiert, um einen Eindruck der Güte zu ermöglichen. Dabei ist zu beachten, dass die Ausgangslage der beiden Datensätze zueinander keinen Unterschied ausmacht. Die Ergebnisse suggerieren, dass selbst fünf Landmarkenpaare für eine sehr gute Vorregistrierung ausreichen.

4.5.4. Ergebnisse der Vorregistrierung

Zum Abschluss des Abschnitts der Vorregistrierung prä- und postoperativer Leberdaten stellen wir noch einmal die Ergebnisse aller vorgestellten Ansätze an einem Beispiel in Abbildung 4.15 visuell vor. Dieses greift noch einmal die beschriebene Problematik auf und zeigt die Vorteile des landmarkenbasierten Ansatzes. Der

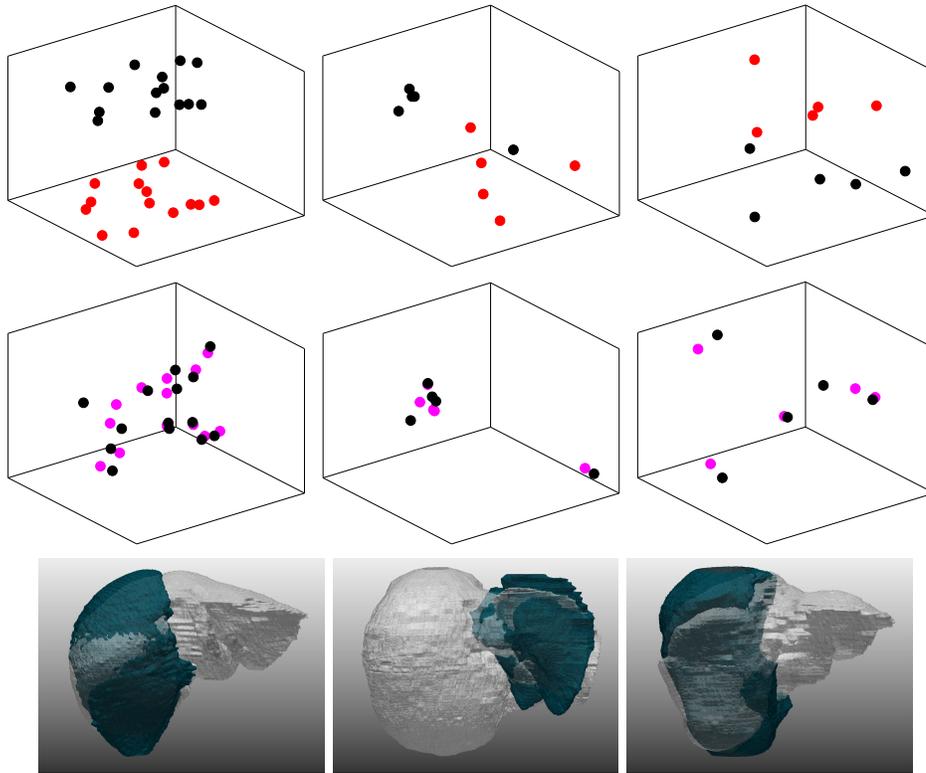


Abbildung 4.14.: Resultate der landmarkenbasierten rigiden Registrierung der drei Testfälle (von oben nach unten). In der obersten Reihe sehen wir die Lage der postoperativen (schwarze Punkte) zu den präoperativen Landmarken (rote Punkte). Die mittlere Reihe zeigt die Lage der Landmarken nach der landmarkenbasierten Vorregistrierung zueinander. Die Punkte liegen sehr eng beieinander. Im Gegensatz zu den beiden vorherigen Verfahren erhalten wir auch visuell bessere Ergebnisse bei Betrachtung der rigide ausgerichteten Lebersegmentierungen (vgl. Abbildungen 4.12 und 4.13)

Ansatz eine Schwerpunktausrichtung vor der intensitäts-basierten rigiden Registrierung auf den Segmentierungen der Leber durchzuführen, findet aufgrund der initialen Lage ein unzureichendes lokales Minimum. Der anschließende nichtlineare Ansatz hat keine Chance mehr eine sinnvolle Lösung des Problems zu erzielen.

Der erste visuelle Eindruck ist ein äußerer, der die Oberfläche der Leber betrach-

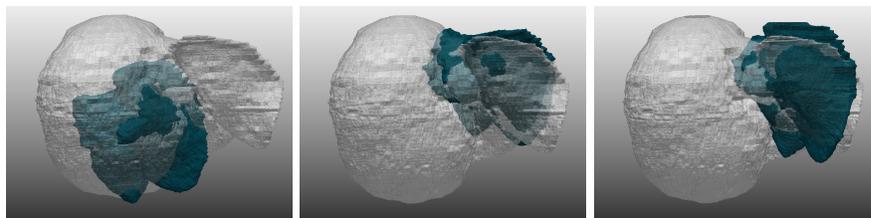


Abbildung 4.15.: Drei verschiedene Arten der Vorpositionierung sind zu betrachten. Die Schwerpunktregistrierung (links), die Vorregistrierung über die IVC (Mitte) und die rigide Registrierung der Landmarken (rechts). Die Registrierung der Landmarken liefert die plausibelste Vorpositionierung und benötigt zudem nur einen Schritt vor dem nichtlinearen Ansatz. Besonders gut zu erkennen ist, dass die Schwerpunktausrichtung eine völlig unzureichende Vorregistrierung einleiten kann

tet. Um einen Eindruck der Korrespondenzen im Inneren der Leber zu gewinnen, betrachten wir die Abstände in der 2-Norm auf den Landmarkenpaaren der Gefäßsysteme HV und PV. Bei der landmarkenbasierten Vorregistrierung sind diese Korrespondenzen für das Gefäßsystem PV natürlich optimiert. Zum Vergleich der Verfahren sind diese Informationen trotzdem sehr interessant. Die maximalen und durchschnittlichen Abstände der korrespondierenden Landmarken sind in Tabelle 4.1 dargestellt. Trotz der wenigen verwendeten Landmarken weist die landmarkenbasierte Registrierung, selbst bei Datensatz 2, sehr gute Ergebnisse für die Distanzen auf. Die großen maximalen Abstände sind auf die starken nichtlinearen Deformationen zwischen den Aufnahmen zurückzuführen.

4.6. Volumenerhaltende Registrierung

Nach einer robusten Vorregistrierung der prä- und postoperativen Daten kommen wir nun zur nichtlinearen Haupt-Registrierung. Wie wir in der Motivation (Abschnitt 4.2) gesehen haben, ist die Anwendung von Standardverfahren nicht zielführend. Sowohl die rigide, als auch die nichtlineare Registrierung, basierend auf standardmäßigem Distanzmaß und Regularisierer scheitern. In diesem Abschnitt wollen wir die Anwendung einer volumenerhaltenden Registrierung motivieren, erläutern und durchführen. Die Grundlagen für diesen Ansatz gehen auf [79] zurück. Hier wurde die Volumenerhaltung in einem Penalty Ansatz umgesetzt, um das Nachwachsen von Tumoren in der Mammographie zu verhindern. In [40, 41] wurde die Volumenerhaltung als Nebenbedingung gefordert und über den diskretisieren-

Tabelle 4.1.: Evaluation der Vorregistrierung über Landmarkendifferenzen $d_L(\mathbf{t}^i, \mathbf{r}^i) = \|\mathbf{t}^i - \mathbf{r}^i\|_2$. Wir sehen deutlich die Stärke des landmarkenbasierten Ansatzes, der in allen Testfällen die besten Ergebnisse erzielt (auch für den Gefäßbaum HV). Die Durchschnittswerte der Landmarkendifferenzen ($\varnothing d_L$) zeigen, wie zu erwarten, eine deutliche Differenz zwischen der landmarkenbasierten und der Schwerpunktsregistrierung. Die großen maximalen Differenzen ($\max d_L$) weisen auf starke Nichtlinearitäten hin

Methode	PV		HV		Datensatz
	$\max d_L$	$\varnothing d_L$	$\max d_L$	$\varnothing d_L$	
Schwerpunkt	46.65	23.65	37.33	27.37	1
VCI	46.16	23.44	36.78	27.07	
Landmarken	15.40	6.21	23.09	7.07	
Schwerpunkt	124.02	111.69	122.12	103.70	2
VCI	47.23	35.65	43.78	28.94	
Landmarken	3.57	2.78	12.38	9.20	
Schwerpunkt	27.96	20.75	129.62	86.51	3
VCI	26.10	20.78	129.35	86.56	
Landmarken	12.21	7.54	120.60	79.99	

dann-optimieren Ansatz umgesetzt. Um die Volumenerhaltung einfach als Nebenbedingung in das Verfahren zu integrieren und die vorgestellten numerischen Optimierungsverfahren (Abschnitt 2.2) nutzen zu können, setzen wir ebenfalls auf diesem Ansatz auf. Dazu starten wir mit der Formulierung des kontinuierlichen Modells, kommen kurz auf die Diskretisierung der Nebenbedingungen zu sprechen und zeigen Ergebnisse der Registrierung. Dabei verwenden wir innerhalb des Augmented Lagrangian Ansatzes zwei verschiedene unrestringierte Optimierungsverfahren, das Gauß-Newton und das L-BFGS Verfahren und untersuchen den Einfluss der unterschiedlichen Methoden auf die Gesamtperformance des Verfahrens.

4.6.1. Kontinuierliche Formulierung

Dem vorgestellten Gerüst unserer Modellierung folgend (siehe Abschnitt 3.6.2), beschreiben wir das Problem durch die Minimierung des Zielfunktional

$$\mathcal{J}(\varphi) = \mathcal{D}(\mathcal{R}, \mathcal{T}; \varphi) + \alpha \mathcal{S}(\varphi - \varphi^{(0)})$$

u. d. NB $c_i(\varphi) = 0 \quad i = 1, \dots, m$.

Nachfolgend werden wir auf die einzelnen Komponenten eingehen. Auf den Nebenbedingungen liegt dabei das Hauptaugenmerk.

Nebenbedingungen Wir haben gesehen, dass ein Modell, das auf einem nichtlinearen Transformationsmodell ohne Nebenbedingungen und einem üblichen Distanzmaß ansetzt, ein ungewolltes Nachwachsen der Leber zulässt. Dagegen kann ein auf rigiden Transformationen aufgesetztes Modell die z. B. durch die Mobilisierung verursachten nichtlinearen Deformationen nicht einfangen. Daher schlagen wir ein Modell vor, das eine nichtlineare Transformationen zulässt und dabei ein Nachwachsen der postoperativen Leber verhindert. Beruhend auf Gleichung (3.4.1) fügen wir Nebenbedingungen ein, die eine punktweise Volumenerhaltung an die Transformation in Bildbereich Ω stellen, d. h.

$$\det(D\varphi(\mathbf{x})) = 1 \quad \forall \mathbf{x} \in \Omega.$$

Die im vorherigen Kapitel genutzten Betragsstriche (Gleichung (3.4.1)) können wir hierbei vernachlässigen, da wir als Startwert die Identität id wählen, für die $\det(D\varphi(\mathbf{x})) = 1$ gilt. Die folgende rigide Vorregistrierung liefert dann ebenfalls das Ergebnis $\det(D\varphi(\mathbf{x})) = 1$.

Die Nebenbedingungen wählen wir in der Form

$$c(\varphi) = f^{(i)}(\det(D\varphi(\mathbf{x}))) \quad \forall \mathbf{x} \in \Omega,$$

wobei der VOLUMENSTRAFTERM $f^{(i)} : \mathbb{R} \rightarrow \mathbb{R}$ die Abweichung der Funktionaldeterminante von der Eins und damit von der Volumenerhaltung bestraft. Für $f^{(i)}$ soll gelten, $f^{(i)}(x) = 0 \Leftrightarrow \det(D\varphi(\mathbf{x})) = 1$. Eine offensichtliche Wahl, wie in [40], ist

$$f^{(1)}(x) = x - 1. \tag{4.6}$$

Jedoch hat hierbei eine Vergrößerung des Volumens weniger Auswirkung als eine Verkleinerung des Volumens. Während eine Halbierung des Volumens $f^{(1)}(\frac{1}{2}) = \frac{1}{2}$ liefert, erhalten wir für eine Verdopplung $f^{(1)}(2) = 1$. Die Forderung $f^{(i)}(x) = f^{(i)}(\frac{1}{x})$ auf eine Symmetrie der Volumenänderung verhindert dieses Verhalten. Es sei angemerkt, dass wir im Folgenden den Begriff Symmetrie nicht bezüglich der Funktion $f^{(i)}$ sondern bezüglich der Volumenänderung verwenden. Eine mögliche symmetrische Wahl,

$$f^{(2)}(x) = \left(\frac{(x-1)^2}{x} \right)^2, \tag{4.7}$$

die diese Forderung zusätzlich erfüllt, wurde in [33] im Kontext einer hyperelastischen Regularisierung vorgestellt. Die ebenfalls symmetrische Wahl

$$f^{(2)}(x) = \frac{(x-1)^2}{x}, \tag{4.8}$$

ist eine weitere Variante. Durch das Quadrat weist die Funktion $f^{(2)}$ wenig Steigung im Bereich der Volumenerhaltung ($f^{(2)} = 1$) auf. Dies kann sich eventuell in der Optimierung als problematisch herausstellen, da die Nebenbedingungen nicht stark genug zur Einhaltung geführt werden. Wir werden im späteren Verlauf die unterschiedlichen Möglichkeiten testen.

Unter vereinfachten Modellannahmen, d. h. unter Vernachlässigung von Schwelungsresten und Stärke des Blutflusses, wollen wir eine exakte Volumenerhaltung fordern. Durch diese Einschränkung legen wir einen Schwerpunkt auf eine einfache Struktur des Modells, da für den gesamten Bereich dieselbe Forderung gestellt wird.

Gesucht ist im Folgenden ein $\varphi \in \mathbb{C}^2(\Omega)$ mit

$$\min_{\varphi} \mathcal{J}(\varphi)$$

u. d. NB $c(\varphi) = f^{(i)}(\det(D\varphi(\mathbf{x}))) = 0 \quad \forall \mathbf{x} \in \Omega.$

4.6.2. Vorverarbeitung

Da wir an der Leber und nicht an den umliegenden Strukturen im CT-Bild interessiert sind, maskieren wir den uns interessierenden Bereich in einem Vorverarbeitungsschritt. Für das Bild \mathcal{T} mit Urbildbereich Ω und dem interessierenden Bereich $G \in \Omega$ ergibt sich dann das vorverarbeitete Bild \mathcal{T}_v mit

$$\mathcal{T}_v(\mathbf{x}) = \begin{cases} \mathcal{T}(\mathbf{x}) & \mathbf{x} \in G, \\ 0 & \text{sonst.} \end{cases} \quad (4.9)$$

In Zuge dieser Arbeit werden wir auf Segmentierungen der Gefäßbäume (siehe Abschnitt 4.1, Abbildung 4.1) arbeiten.

4.6.3. Distanzmaß

Da es sich um eine monomodale CT-Registrierung handelt, verwenden wir SSD (Abschnitt 3.5) als Distanzmaß \mathcal{D} . CT-Daten haben den Vorteil, dass bestimmte Strukturen immer den gleichen Grauwert haben. Diese Werte sind in Houndsfield angegeben [45].

4.6.4. Regularisierer

Um eine hohe, globale Glattheit zu präferieren, wählen wir die curvature Regularisierung (siehe Abschnitt 3.6.1). Die globale Glattheit spielt dabei zudem mit der punktwisen Volumenerhaltung auf ganz Ω zusammen, die auf eine global glatte Deformation führen sollte. Als initiale Transformation $\varphi^{(0)}$ verwenden wir hierbei das Ergebnis der Vorregistrierung.

Im Folgenden wenden wir uns der Diskretisierung zu.

4.6.5. Diskretisierung

Aufgrund der volumenerhaltenden Nebenbedingungen nutzen wir ein nodales Gitter zur Diskretisierung unseres Zielfunktional (siehe Abschnitt 3.4). Die Diskretisierung des Distanzmaßes und des Regularisierers wurde bereits in Kapitel 3 besprochen. Bei der Diskretisierung des Distanzmaßes ist darauf zu achten, dass die Auswertung der Differenzen weiterhin auf einem zellzentrierten Gitter stattfindet. Der Wechsel vom nodalen auf das zellzentrierte Gitter findet sich auf in der Ableitung des Distanzmaßes wieder.

In Abschnitt über Transformationen (Abschnitt 3.4) sind wir bereits auf die Diskretisierung der Funktionaldeterminante eingegangen. Dafür wurden verschiedene Vorgehensweisen vorgeschlagen. Wir wählen die Diskretisierung der Funktionaldeterminante ohne Dekomposition in Dreiecke bzw. Tetraeder trotz ihres Nachteils, dass Faltungen eventuell nicht erkannt werden. Diese Wahl beruht auf zwei Gründen. Zum einen fordern wir eine exakte Volumenerhaltung. Die Diskretisierung gilt als konsistent für diese Anwendung [44], es treten also keine undetektieren Faltungen auf. Zum anderen ist es aber auch ein Kostengrund. Da wir ein dreidimensionales Problem vorliegen haben, impliziert die andere Möglichkeit, jeden Voxel in Tetraeder zu zerlegen und deren Volumina zu überprüfen. Im besten Fall haben wir somit fünf Nebenbedingungen pro Voxel anstatt einer einzigen.

Beim Volumenstrafterm $f^{(i)}$ untersuchen wir die unterschiedlichen Wahlen, die symmetrischen und unsymmetrische. Wir betrachten aus Übersichtsgründen den zweidimensionalen Fall mit der Volumendiskretisierung v_i aus Gleichung (3.30). Durch die Wahl der Diskretisierung der Funktionaldeterminante erhalten wir $\bar{m} = m_1 \cdot m_2$ Nebenbedingungen, die wir in dem Vektor $\mathbf{c} = (\mathbf{c}_1, \dots, \mathbf{c}_{\bar{m}})$ zusammenfassen mit

$$\mathbf{c}_i = f^{(1)}(v_i) = v_i - 1$$

$$\mathbf{c}_i = f^{(2)}(v_i) = \left(\frac{(v_i - 1)^2}{v_i} \right)^2$$

bzw.

$$\mathbf{c}_i = f^{(3)}(v_i) = \frac{(v_i - 1)^2}{v_i}.$$

Für die Optimierung benötigen wir zusätzlich die ersten Ableitungen. Mit der Kettenregel ergibt sich

$$d\mathbf{c}_i = df^{(i)} dv_i, \quad i = 1, 2, 3 \quad (4.10)$$

mit

$$\begin{aligned} df^{(1)}(x) &= 1 \quad \text{und} \\ df^{(2)}(x) &= \frac{2(x+1)(x-1)^3}{x^3} \\ df^{(3)}(x) &= \frac{x^2 - 1}{x^2}. \end{aligned}$$

Die Ableitung dv_i der Volumendiskretisierung betrachten wir etwas detaillierter für die Nebenbedingung \mathbf{c}_i . Exemplarisch für die Ableitung nach den einzelnen Komponenten von φ betrachten wir die Ableitung nach den Zellecken $\varphi_a, \varphi_b, \varphi_c, \varphi_d$. Es ergibt sich mit $\varphi_k = (\varphi_k^1, \varphi_k^2)^T$, $k \in \{a, b, c, d\}$,

$$\begin{aligned} \frac{\partial v_i}{\partial \varphi_a^1} &= -\frac{1}{h_1 h_2} (\varphi_b^2 - \varphi_d^2), & \frac{\partial v_i}{\partial \varphi_a^2} &= \frac{1}{h_1 h_2} (\varphi_b^1 - \varphi_d^1), \\ \frac{\partial v_i}{\partial \varphi_b^1} &= -\frac{1}{h_1 h_2} (\varphi_c^2 - \varphi_a^2), & \frac{\partial v_i}{\partial \varphi_b^2} &= \frac{1}{h_1 h_2} (\varphi_c^1 - \varphi_a^1), \\ \frac{\partial v_i}{\partial \varphi_c^1} &= \frac{1}{h_1 h_2} (\varphi_b^2 - \varphi_d^2), & \frac{\partial v_i}{\partial \varphi_c^2} &= -\frac{1}{h_1 h_2} (\varphi_b^1 - \varphi_d^1), \\ \frac{\partial v_i}{\partial \varphi_d^1} &= \frac{1}{h_1 h_2} (\varphi_c^2 - \varphi_a^2), & \frac{\partial v_i}{\partial \varphi_d^2} &= -\frac{1}{h_1 h_2} (\varphi_c^1 - \varphi_a^1), \end{aligned}$$

und Null sonst für

$$dv_i = \left(\frac{\partial v_i}{\partial \varphi_1^1} \quad \cdots \quad \frac{\partial v_i}{\partial \varphi_{(m_1+1)(m_2+1)}^1} \quad \frac{\partial v_i}{\partial \varphi_1^2} \quad \cdots \quad \frac{\partial v_i}{\partial \varphi_{(m_1+1)(m_2+1)}^2} \right) \in \mathbb{R}^{2(m_1+1)(m_2+1)}$$

Die restlichen Komponenten der Deformation φ haben keinen Einfluss auf den betrachteten Voxel.

4.6.6. Ein Multilevel Augmented Lagrangian

Im Grundlagenkapitel der Bildregistrierung (Abschnitt 3.8) haben wir uns mit dem Multilevel Ansatz zur Stärkung der Robustheit und Geschwindigkeitserhöhung beschäftigt. Dies wollen wir auch bei Verwendung von Nebenbedingungen nutzen. Die Volumen Nebenbedingungen werden durch die Diskretisierung voxelweise gestellt. Dies zieht eine Prolongation der Volumenberechnung und damit der Nebenbedingungen mit sich. Daraus resultiert auch eine Prolongation der Lagrange-Multiplikatoren. Um diesen Modifikationen Rechnung zu tragen, erweitern wir den Augmented Lagrangian (siehe Abschnitt 2.2) um die Multilevel Idee, visualisiert in Quelltext 4.1.

Auf kleineren Auflösungen wird eine Approximation der eigentlichen Lösung φ_L^* mit zugehörigem λ_L^* berechnet. Diese dient als Startwert auf dem nächst feineren Level, auf dem wieder eine neue Approximation berechnet wird. Dieser Ansatz wird bis zum feinsten Level verfolgt.

Durch die lineare Prolongation der Deformation (siehe Abschnitt 3.8) werden die Volumen Nebenbedingungen ebenfalls linear prolongiert, da diese auf der prolongierten Deformation direkt wieder berechnet werden. Unter diesen Voraussetzungen und da der Startwert der Schätzung des Lagrange-Vektors $\lambda^{(0)}$ essentiell wichtig für die Effizienz des Verfahrens (vgl. Abschnitt 2.2) ist, führen wir für die Lagrange-Multiplikatoren ebenfalls eine lineare Interpolation durch. Da wir für den Strafparameter μ_L^* typischerweise einen sehr großen Wert nach Beendigung eines Levels erwarten und bei großen Werten von μ die Hessematrix schlechter konditioniert ist, verwenden wir zum Start auf dem neuen Level $\mu_{L+1}^{(0)} \geq \mu_L^{(0)}$ anstatt des letzten Wertes der vorherigen Iteration. Eine Vergrößerung des Strafparameters führt zu einem sicheren Übergang der Volumen Nebenbedingungen, da die Approximation eine Güte von h^2 besitzt und somit kleine Abweichungen auf dem groben Level sich im feinen Level wieder vergrößern können.

Im Zuge dieser Arbeit verwenden wir zur Minimierung der Augmented Lagrangian das Gauss-Newton- und L-BFGS-Verfahren. Beim L-BFGS-Verfahren wird als initiale Approximation der Hessematrix die diskrete exakte zweite Ableitung des Regularisierers genutzt. Aus Abschnitt 3.6.1 geht hervor, dass diese dünn-besetzt ist und damit das resultierende lineare Gleichungssystem effizient gelöst werden kann. Zudem speichern wir die letzten fünf Vektorpaare, also wählen wir $p = 5$.

Quelltext 4.1: Vereinfachter Modell-Algorithmus für Multilevel in der Bildregistrierung mit Prolongation der Nebenbedingungen

```

1  % Zielfunktion  $f: \mathbb{R}^n \rightarrow \mathbb{R}$ 
2  % Nebenbedinugnen  $c: \mathbb{R}^m \rightarrow \mathbb{R}$ 
3  % Startschätzung  $\varphi^{(0)} \in \mathbb{R}^n$ ,  $\lambda^{(0)} \in \mathbb{R}$ ,  $\mu^{(0)} \in \mathbb{R}$ 
4  function  $\varphi^{\text{ende}} = \text{MLAL}(f, c, \varphi^{(0)}, \lambda^{(0)}, \mu^{(0)})$ 
5  for L = minLevel:maxLevel
6      if L == minLevel
7           $\varphi^{(0)} = \varphi^{(0)}$ 
8           $\lambda^{(0)} = \lambda^{(0)}$ 
9           $\mu_L^{(0)} = \mu^{(0)}$ 
10     else
11          $\varphi^{(0)}$  aus linearer Prolongation von  $\varphi_{L-1}$ 
12          $\lambda^{(0)}$  aus linearer Prolongation von  $\lambda_{L-1}$ 
13     end
14
15     Auswertungen der Nebenbedingungen c etc.
16
17      $k = 0$ 
18     while (Abbruchbedingungen nicht erfüllt)
19         Approximiere Minimierer  $\varphi^{k+1}$  von  $\mathcal{L}_A(\varphi, \lambda^{(k)}, \mu_L^{(k)})$ 
20         mit Startwert  $\varphi^{(k)}$  und  $\|\mathcal{L}_A(\varphi^{(k+1)}, \lambda^{(k)}, \mu^{(k)})\| \leq \tau^{(k)}$ 
21         Aufdatierung:  $\mu_L^{(k+1)} \geq \mu_L^{(k)}$ 
22         Aufdatierung:  $\lambda^{(k+1)}$ 
23         Wähle Toleranz  $\tau^{(k+1)}$ 
24          $k = k + 1$ 
25     end
26      $\varphi_L = \varphi^{(k+1)}$ 
27      $\lambda_L = \lambda^{k+1}$ 
28     Wähle  $\mu_{L+1}^{(0)} \geq \mu_L^{(0)}$ 
29 end
30 end

```

4.6.7. Anwendung und Vergleich von Volumenstraftermen

Zu Beginn der Evaluation des Verfahrens stellen wir beide Volumenstrafterme (4.6) und (4.7) gegenüber. Wir testen diese an den drei vorgestellten Testdatensätzen. Für diesen Vergleich verwenden wir den Multilevel Augmented Lagrangian mit einem Gauß-Newton Optimierer und Armijo Backtracking Schrittweitenbestimmung, $\rho = 10^{-4}$. Dabei wenden wir statt der Toleranz $\tau^{(k)}$ jeweils nur einen Iterationsschritt in der unrestringierten Optimierung der Augmented Lagrange Funktion an und führen dadurch eine Aufdatierung von λ in jedem Schritt durch. Wir erhoffen uns dadurch eine schnelle, gute Approximation des Lagrange-Vektors λ . Als Level verwenden wir die Auflösungen $m^{grob} = (16, 16, 16)$ und $m^{fein} = (32, 32, 32)$. Wir verwenden die Aufdatierungsregel (2.18) für λ innerhalb eines Levels und die lineare Interpolation zur Bestimmung von λ_0 auf dem nächst feineren Level. Auf den Leveln verwenden wir $\mu^{(0)} = 10^2, 10^3$. Für die Abbruchkriterien wurde die maximale Anzahl an Iterationsschritten für die einzelnen Level unterschiedlich mit 250, 100 von grob nach fein gewählt. Der Wahl liegen zwei Aspekte zu Grunde. Zum einen sollte durch den guten Startwert die Anzahl der Iterationen auf den feineren Leveln sinken, zum anderen liegt hier aber auch ein Zeitaspekt vor. Die Zeit für einen Iterationsschritt mit Volumenerhaltung ist sehr kostspielig im Vergleich zu einem Standard-Verfahren. Die Berechnungen erfolgen auf den Segmentierungen des Gefäßbaums PV, da besonderer Wert auf die Registrierung der Gefäßbäume gelegt wird. Die initiale Transformation $\varphi^{(0)}$ erhalten wir aus der rigiden Landmarkenregistrierung. Die Ergebnisse sind in Abbildung 4.16 zu sehen. Obwohl die Berechnungen nur auf einem Gefäßbaum erfolgen, sind sogar die restlichen Bereiche der Leber sehr gut angepasst. Es ist rein optisch eine deutliche Verbesserung bezüglich der Vorregistrierung zu erkennen. Die Verbesserung schlägt sich auch im Distanzmaß nieder. Wir betrachten daher die RELATIVE VERBESSERUNG DER DISTANZ

$$d_{\text{rel}} = \frac{\mathcal{D}(\varphi^*)}{\mathcal{D}(\varphi^{(0)})}, \quad (4.11)$$

wobei φ^* das Ergebnis der Registrierung darstellt. Für die Datensätze 1-3 ergeben sich relative Verbesserungen der Distanz von $d_{\text{rel}} = 0.3761$, $d_{\text{rel}} = 0.1909$ bzw. $d_{\text{rel}} = 0.7092$. Es sei angemerkt, dass diese starke Verbesserung trotz starker Einschränkungen an die Transformation, wie sie die Volumenbedingungen einführen, erreicht werden.

Im nächsten Schritt führen wir die Registrierung mit dem symmetrischen Volumenstrafterm (4.7) bei gleicher Optimierung durch. Es stellt sich heraus, dass die Parameter $\alpha, \mu^{(0)}$ anders gewählt werden müssen, da sonst nach der maximalen Anzahl an Iterationsschritten die Nebenbedingungen keineswegs erfüllt sind. Die

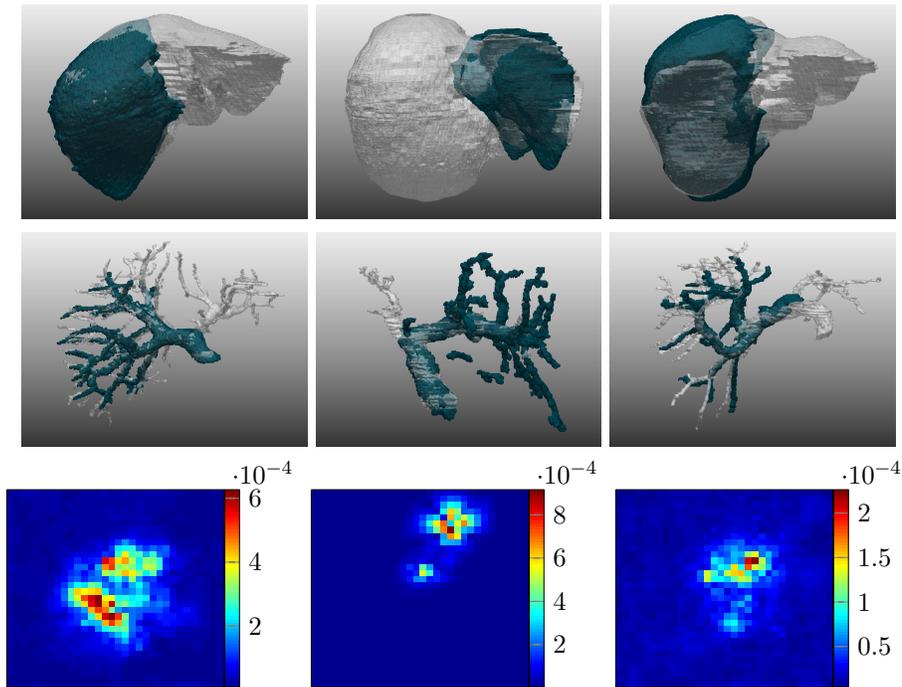


Abbildung 4.16.: Ergebnisse der volumenerhaltenden Registrierung mit dem nichtsymmetrischen Volumenstrafterm (4.6) und $\alpha = 10^8$. Berechnungen erfolgten auf den Segmentierungen der Gefäßbäume PV bei einer höchsten Auflösung von $m^{\text{fein}} = (32, 32, 32)$. Im Vergleich zur rigiden Registrierung konnten deutliche Verbesserungen erzielt werden. Selbst die Oberfläche (obere Reihe) wird deutlich besser ausgerichtet, obwohl diese nicht mit in die Berechnung eingeht. Besonders bei den ersten beiden Datensätzen sind die Ausrichtungen der Gefäßbäume (mittlere Reihe) hervorragend. In der unteren Reihe sind die Volumenänderungen über eine Projektion des maximalen Wertes über die erste Dimension dargestellt. Es ist deutlich sichtbar, dass kaum eine Volumenänderung vorliegt.

neue Wahl lautet $\alpha = 10^8$ und $\mu^{(0)} = 10^8$ für das grobe bzw. 10^{11} für das feine Level. Die Ergebnisse sind in Abbildung 4.17 visualisiert. Wir sehen, dass die Ergebnisse bezüglich der symmetrischen volumenerhaltenden Registrierung deutlich schlechter ausfallen. Der Ansatz ist nicht in der Lage, in einer vernünftigen Anzahl von Iterationsschritten eine äquivalente Genauigkeit im Vergleich zum vorherigen Ansatz zu erzielen. Das Problem liegt dabei am Verhalten der symmetrischen Straffunktion (4.7) nahe der Eins, also der Volumenerhaltung. Betrachten wir Abbildung 4.18, stellen wir fest, dass kaum eine Änderung des Funktionswertes in dieser Region auftritt. Bei der Betrachtung von $f^{(1)}$ ist dies allerdings der Fall. Im Optimierungsalgorithmus spiegelt sich dies in einem geringen Abstieg in den Nebenbedingungen wider. Um diesem Verhalten entgegenzuwirken, haben wir den Start-Strafparameter $\mu^{(0)}$ erhöht. Dies führt schlussendlich auch zu schlechteren Ergebnissen in der relativen Verbesserung der Distanz durch den größeren Einfluss der Nebenbedingungen. Wir erhalten $d_{\text{rel}} = 0.6034$, $d_{\text{rel}} = 0.7127$ und $d_{\text{rel}} = 0.9193$ für die Datensätze 1-3. Gleiches Problem stellt sich bei der zweiten symmetrischen Variante (Gleichung (4.8)) heraus. Es ergeben sich relative Verbesserungen der Distanzen von $d_{\text{rel}} = 0.6036$, $d_{\text{rel}} = 0.7711$ und $d_{\text{rel}} = 0.9193$ bei Regularisierungsparameter $\alpha = 5 \cdot 10^9$ und sonst gleicher Konfiguration der Parameter. Die Ergebnisse sind in Abbildung 4.19 zu betrachten.

Es stellt sich heraus, dass bei exakter Volumenerhaltung der (Volumen-)Straffunktion (4.6) in allen Fällen die besseren Ergebnisse erzielt werden. Wir schlagen daher vor, diesen Strafterm zu verwenden und werden dieses Ergebnis zur Untersuchung der Optimierungsverfahren übernehmen. Zudem stellen wir fest, dass die Rechenzeit in allen Fällen im Vergleich zu einer unrestringierten Registrierung deutlich gestiegen ist. Dies liegt zum einen an der Auswertung des Nebenbedingungen (siehe Abschnitt 3.4) als auch an dem Gegenspiel zwischen Funktionsoptimierung und Erfüllung der Nebenbedingungen im Augmented Lagrangian.

4.6.8. Vergleich der Optimierungsverfahren

Im vorangegangenen Beispiel haben wir die Unterschiede in der Wahl der Nebenbedingungen bei Anwendung des Gauß-Newton-Verfahrens gesehen. Als einen weiteren Schritt wollen wir nun den Einfluss des unrestringierten Optimierers untersuchen. Da wir Probleme mit sehr vielen Unbekannten vorliegen haben, betrachten wir das L-BFGS-Verfahren (siehe Abschnitt 2.1.4) als zweite Option. Für die Approximation der Start-Hesse-Matrix setzen wir $H^{(0)} = d^2\mathbf{S} = \mathbf{A}^{\text{curv}}$ (vgl. Abschnitt 3.6.1), die exakte Hesse-Matrix des Regularisierers. Im Gegensatz zum Augmented Lagrangian mit dem Gauß-Newton-Verfahren, führen wir nicht nur jeweils einen Schritt des L-BFGS-Verfahrens die Augmented Lagrangian Funktion

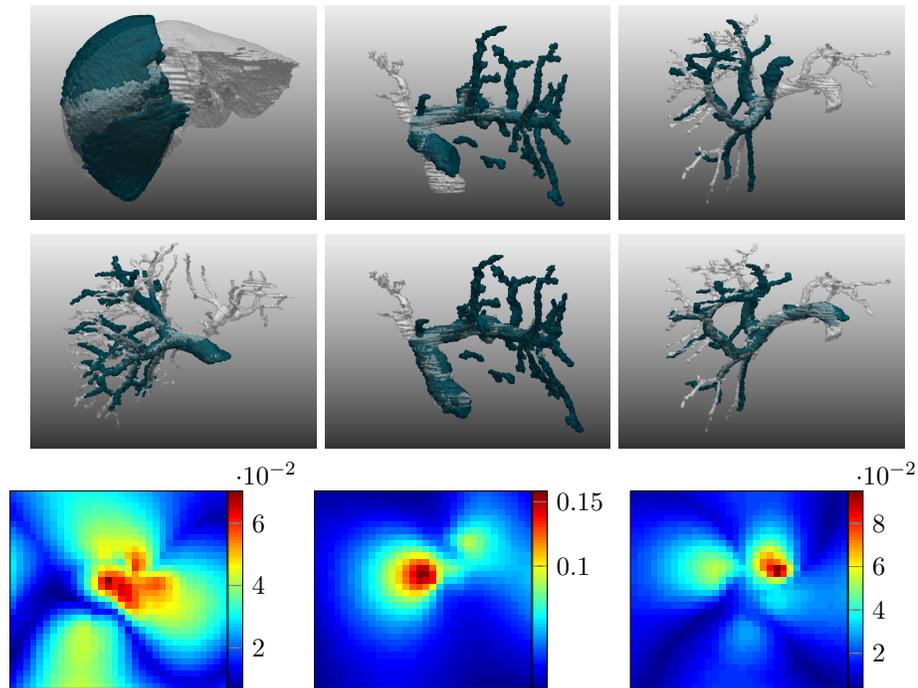


Abbildung 4.17.: Ergebnisse der volumenerhaltenden Registrierung mit dem symmetrischen Volumenstrafterm (4.7) und $\alpha = 10^8$. Berechnungen erfolgten auf den Segmentierungen des Gefäßbaumes PV bei einer höchsten Auflösung der Transformation von $m^{\text{fein}} = (32, 32, 32)$. In der unteren Reihe sind die Volumenänderungen über eine Projektion des maximalen Wertes über die erste Dimension dargestellt. Wir können deutlich eine Volumenänderung sehen. Im Vergleich zur nicht-symmetrischen Funktion werden deutlich schlechtere Ergebnisse bezüglich der Problemstellung erzielt.

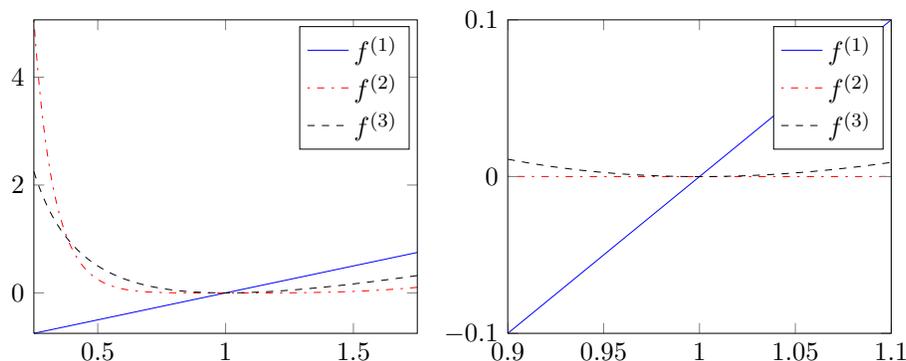


Abbildung 4.18.: Vergleich der Volumenstraffunktionen (4.6), (4.7) und (4.8). Es zeigt sich deutlich, dass gerade nahe der Volumenerhaltung (rechts) die nichtsymmetrische Variante (4.6) eine höhere Strafe bewirkt. Die bezüglich der Volumenänderung symmetrischen Funktionen (4.7) und (4.8) sind in diesem Bereich dagegen beinahe konstant.

$\mathcal{L}_A(\mathbf{x}, \boldsymbol{\lambda}^{(k)}, \mu^{(k)})$ durch, da sonst die Approximation der Hesse-Matrix immer bei $H^{(0)}$ liegt. Wir würden dann nur die Krümmungsinformationen des Regularisierers nutzen. Jedoch setzen wir innerhalb des L-BFGS-Verfahrens eine relativ niedrige Schranke bei der maximalen Iterationszahl. In der äußeren Schleife des Augmented Lagrangian Ansatzes wählen wir zusätzlich eine von groben zu feinen Leveln sinkende maximale Anzahl von Schritten. In unserem Beispiel fällt die Wahl auf 35 äußere Iterationen auf dem groben und 10 Iterationen auf dem feinen Level, bei jeweils maximal 30 Iterationen innerhalb des L-BFGS-Verfahrens mit $p = 5$ gespeicherten Vektorpaaren. Der Regularisierungsparameter wurde auf $\alpha = 10^8$ gesetzt. Die Wahl des Strafparameters fiel auf $\mu^{(0)} = 10^8$ und $\mu^{(0)} = 10^8$ für das grobe bzw. feine Level. Es machten sich bei dieser Wahl keine Probleme mit der Kondition bemerkbar. Für die Abbruchkriterien wurden $\tau_c = 10^{-8}$ und $\tau = 10^{-6}$ gewählt.

Wir wenden das Verfahren wiederum auf die drei bekannten Testdatensätze an. Die Ergebnisse sind in Abbildung 4.20 zu betrachten. Es ergeben sich äquivalente relative Verbesserungen der Distanz von $d_{\text{rel}} = 0.4144$, $d_{\text{rel}} = 0.2248$ bzw. $d_{\text{rel}} = 0.7309$ zum Gauss-Newton-Verfahren. Der große Vorteil bei der Nutzung des L-BFGS-Verfahrens ist aber eindeutig die Rechendauer. Hier ist das L-BFGS-Verfahren deutlich überlegen. Es macht sich eindeutig bemerkbar, dass keine Gleichungssysteme gelöst werden müssen. Dies würde sich noch deutlicher bemerkbar machen, wenn der Multilevel Ansatz am Ende noch auf feineren Leveln als $(32, 32, 32)$ arbeiten würde. Weitere Tests haben allerdings auch einen Nachteil ergeben. Das Verfahren

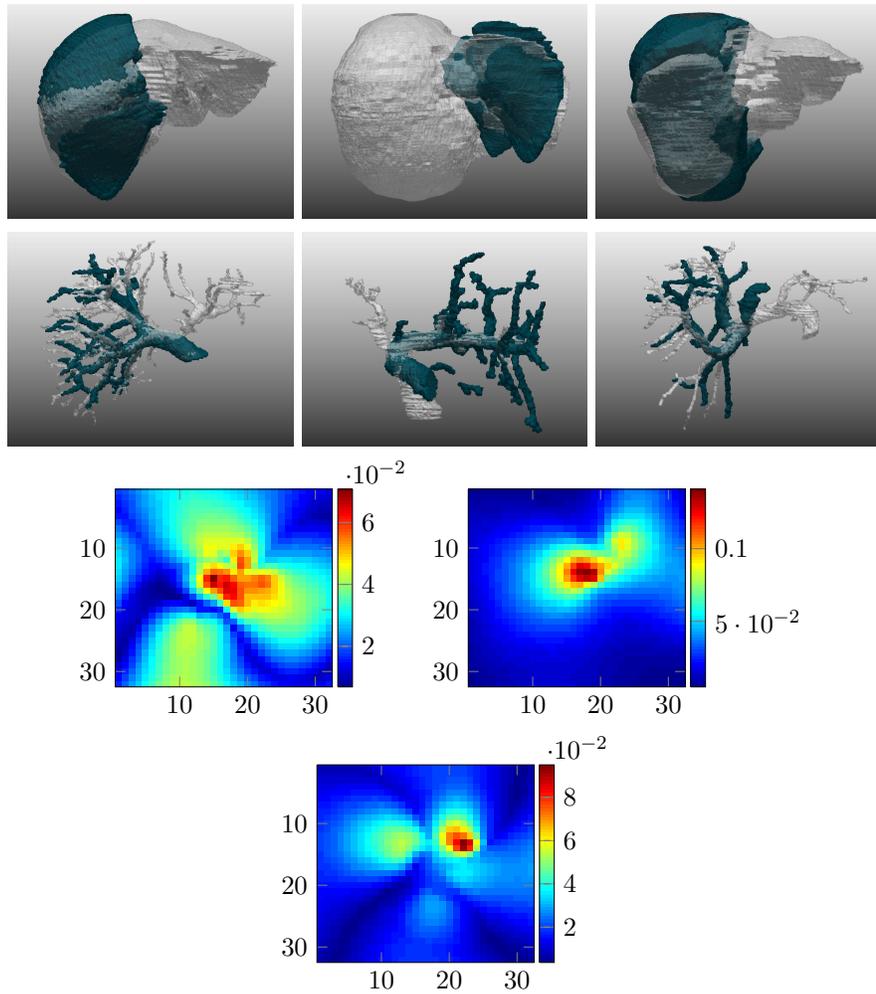


Abbildung 4.19.: Ergebnisse der volumenerhaltenden Registrierung mit dem symmetrischen Volumenstrafterm (4.8) und $\alpha = 5 \cdot 10^9$. Berechnungen erfolgten auf den Segmentierungen des Gefäßbaumes PV bei einer höchsten Auflösung der Transformation von $m^{\text{fein}} = (32, 32, 32)$. In der unteren Reihe ist die Volumenänderung über eine Projektion des maximalen Wertes über die erste Dimension dargestellt. Wir können deutlich eine Volumenänderung sehen. Im Vergleich zur nicht-symmetrischen Funktion werden wiederum deutlich schlechtere Ergebnisse bezüglich der Problemstellung erzielt.

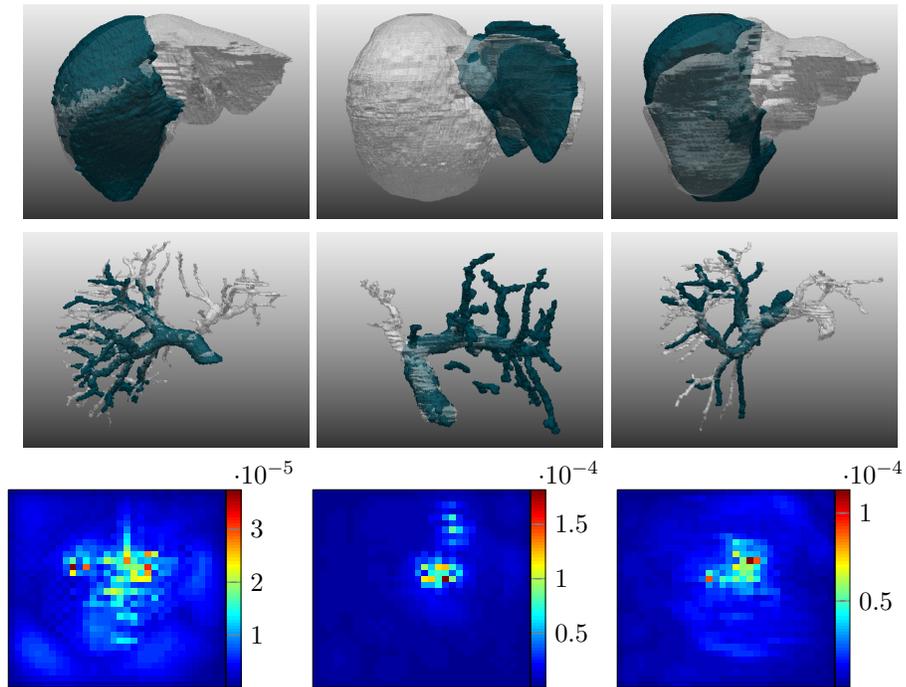


Abbildung 4.20.: Ergebnisse der volumenerhaltenden Registrierung mit dem Augmented Lagrangian unter Nutzung des L-BFGS-Verfahrens als unrestringierten Optimierungsalgorithmus. Berechnungen erfolgten für $\alpha = 10^8$ auf den Segmentierungen des Gefäßbaumes PV bei einer höchsten Auflösung der Transformation von $m^{\text{fein}} = (32, 32, 32)$. In der untersten Reihe ist die Volumenänderung über eine Projektion des maximalen Wertes über die erste Dimension dargestellt. Wir können darin deutlich erkennen, dass die Volumen Nebenbedingungen gut eingehalten wurden. Die optischen Ergebnisse entsprechen denen in Abbildung 4.16 unter Nutzung des Gauß-Newton-Verfahrens.

ist anfälliger bezüglich der Wahl der Startparameter.

4.6.9. Fazit und Ausblick

In diesem Abschnitt haben wir die Volumenerhaltung für den Einsatz im Umfeld der Registrierung prä- und postoperativer Daten untersucht. Dafür haben wir einen Blick auf die Formulierung der Nebenbedingungen geworfen und das Gauß-Newton mit dem L-BFGS-Verfahren im Rahmen des Augmented Lagrangian für die volumenerhaltende Registrierung verglichen. Es stellt sich heraus, dass die Nutzung der einfachen nicht-symmetrischen Nebenbedingungen (4.6) sich im Rahmen der exakten Volumenerhaltung als genauer erweist. Beim Optimierer fällt die Wahl schwieriger aus. Sind vernünftige Parameter bekannt, fällt die Wahl auf das L-BFGS-Verfahren aufgrund des geringeren Rechenaufwands. Ist von vornherein keine geeignete Wahl bekannt, lohnt sich die Verwendung des stabileren Gauß-Newton-Verfahrens. Die Unterschiede in den Ergebnisse sind sonst marginal. Es macht sich jedoch in beiden Fällen bemerkbar, dass die Volumenerhaltung auf dem gesamten Urbildbereich mit großem Rechenaufwand einher geht und die möglichen Transformationen stark einschränkt.

Richten wir unseren Blick einen Schritt weiter. Da wir in dieser Arbeit das Augenmerk auf die Machbarkeit gelegt haben, sind die Algorithmen noch nicht optimiert bezüglich der Rechenzeit. Bisher haben wir mit dem Augmented Lagrangian ein effizientes Verfahren zur Optimierung des diskretisierten Problems genutzt. Innerhalb des Verfahrens treten aber noch rechentechnisch kostenintensive Baustellen auf, die wir hier nicht weiter betrachtet haben. Bisher haben wir die exakte Einhaltung des Volumens gefordert. Durch variierenden Blutfluss und leichte Restschwellungen, kann dies allerdings nicht überall gelten - auch wenn es ein erster einfacher Schritt ist, der gut ins Modell eingeführt werden kann. Es ist jedoch durchaus vorstellbar, das Modell zu verfeinern. Eine Möglichkeit besteht dabei in einer regionalen Anpassung der Volumenbedingungen, z. B. mit Ungleichheitsnebenbedingungen auf den Gefäßen und am Schnitttrand, an dem Schwellungen zu vermuten sind.

Im Folgenden wollen wir uns allerdings einer Alternative zur volumenerhaltenden Registrierung der prä- und postoperativen Leberdaten zuwenden, bei der die oben angesprochenen Probleme hoffentlich nicht oder zumindest weniger stark auftreten.

4.7. Längenerhaltende Registrierung

Mit der volumenerhaltenden Registrierung haben wir im letzten Abschnitt ein sehr gutes Verfahren zur Registrierung prä- und postoperativer Leberdaten kennengelernt. Wie bereits angesprochen ist dieses Verfahren sehr robust. Auf der Kehrseite der Medaille ist die Laufzeit hoch, da für jeden Voxel die Volumenänderung mit berechnet wird. Zudem ist die Volumenerhaltung an manchen Stellen eventuell zu einschränkend an die Deformation. Dies kann z. B. bei den Gefäßen zum Tragen kommen, da durch unterschiedlich starken Blutfluss unterschiedliche Volumina auftreten. Daher möchten wir in diesem Abschnitt eine Alternative vorstellen. Diese soll zum einen die Laufzeit verbessern und zum anderen die Deformation weniger stark einschränken.

Wie wir bereits in Abschnitt 4.6 gesehen haben, gibt es zwischen den beiden Aufnahmen starke, nichtlineare Deformationen. Wir setzen also wieder ein nichtlineares Deformationsmodell an. Da die Operationsplanung sich aufgrund der Segmenteinteilung der Leber nach Couinaud (Abschnitt 4.1.1) an den Gefäßen orientiert, legen wir unser Augenmerk auf diese. Die Volumina der Gefäße können sich zwar ändern, allerdings bleibt deren Länge zwischen den Aufnahmen fast konstant. Diese Überlegung ist Grundlage der neuen Nebenbedingungen. Diese sollen eine Längenerhaltung der Gefäße fordern. Zudem erhoffen wir uns aus dieser Forderung, dass diese Nebenbedingungen nicht auf dem gesamten Bereich, sondern nur auf den Gefäßen gestellt werden, eine Reduzierung der Anzahl der nötigen Nebenbedingungen und damit eine Reduzierung der Laufzeit durch weniger Berechnungen.

Ziel soll es im Folgenden sein, ein Verfahren zu entwickeln, das bei mindestens gleichbleibender Qualität der Ergebnisse die Laufzeit im Vergleich zur volumenerhaltenden Variante verbessert.

4.7.1. Stand der Technik

Unserer Kenntnis nach ist die Längenerhaltung bis auf in den Arbeiten von Groher und Zikic [36, 96, 37] noch nicht in der Registrierungsanwendung aufgetreten. In ihren Arbeiten geht es um eine nichtlineare 2D-3D-Registrierung von gefäßartigen Strukturen. Das Problem hierbei ist die Projektion des 3D-Volumens auf eine 2D-Schicht. Durch die Projektion von 3D auf 2D gehen Informationen verloren, z. B. über die Länge eines Gefäßes im 3D-Volumen. So kann ein unterschiedlich stark gekrümmtes Gefäß in der Projektionsebene kürzer erscheinen als es eigentlich ist. Um letztendlich eine plausible Registrierung zu gewähren, wurde eine Längenerhaltung der Gefäße als weiche Nebenbedingung in das Optimierungsproblem eingebracht (vgl. Abschnitt 3.6.2). Dazu werden korrespondierende Punkte in beiden Bildern auf

den Gefäßen gesucht. Die Deformation wird dann anhand dieser Punkte bestimmt und mittels Thin Plate Spline-Interpolation auf dem gesamten Bild ausgewertet. Es wird also eine parametrische Transformation (vgl. Abschnitt 3.4) genutzt. Die Distanz wird in den Arbeiten über die Abstände der korrespondierenden Punkten berechnet. Eine feature-basierte Distanz (vgl. Abschnitt 3.5) findet hier also ihre Verwendung.

Im Gegensatz zu diesen Arbeiten werden wir eine intensitäts-basierte Distanz nutzen. Wir verwenden darüber hinaus eine nichtparametrische Transformation und benötigen nur im Templatebild Informationen über die Längen der Gefäße und brauchen daher keine Korrespondenzen herzustellen. Zudem binden wir die Längenerhaltung als Gleichheitsnebenbedingungen ein. Diese Modellannahme mag im ersten Moment zu einschränkend wirken, jedoch ist die Festlegung eines Toleranzbereiches auch fehlerbehaftet. In einem späteren Schritt können wir das Modell jedoch leicht auf Ungleichheitsnebenbedingungen ausweiten. Im Folgenden wollen wir uns mit der Einbindung der Nebenbedingungen in das bestehende Registrierungsmodell, bestehend aus Distanzmaß und Regularisierer, beschäftigen. Da die Längenerhaltung auf den Gefäßen angewendet werden soll, starten wir - nach einer kurzen Schilderung der Gegebenheiten - mit einem Modell dieser.

4.7.2. Modellierung der Gefäße

Wie wir bei der Einführung in die Anatomie der Leber und der Leberresektion gesehen haben (siehe 4.1), sind Informationen über die Lage der Gefäße essentiell wichtig für die Planung der Resektion. Daher sollen Informationen über die Gefäße mit in das Registrierungsmodell einfließen - hier in Form einer Längenerhaltung, da die Länge für korrespondierende, verbleibende Teilstücke der Gefäße vor und nach dem Eingriff gleich bleiben sollte. Für die Bestimmung der Länge eines Gefäßes benötigen wir allerdings nicht das gesamte Volumen des Gefäßes, sondern lediglich Informationen über den Verlauf. Dieser kann durch die zentrale Linie des Gefäßbaums, das Skelett, beschrieben werden. Um ein klares Verständnis der folgenden Definitionen zu ermöglichen, wollen wir zunächst die Datenlage näher beleuchten. Wir betrachten dazu zweidimensionale Beispiele. Die Erweiterung auf den dreidimensionalen Fall erfolgt analog.

Datenlage In Abschnitt 4.4 haben wir erfahren, dass zu den Gefäßbäumen Daten in Form einer Baumstruktur gespeichert sind. Die Wurzel dieses Baumes ist z. B. der Eintrittspunkt des Gefäßes in die Leber. Die Knoten des Baumes enthalten Informationen über die Gefäßverzweigungen, während in den Blättern die Endpunkte

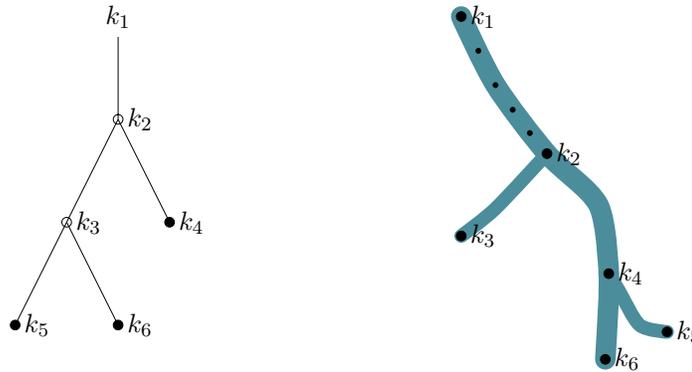


Abbildung 4.21.: Informationen der Gefäßbäume werden in Bäumen gespeichert. Hier sehen wir den Zusammenhang zwischen der Baumstruktur (links) und der Lage des Gefäßbaumes im Raum (rechts). Beispielhaft sind die Positionen der Verzweigungs- bzw. Endpunkte, die mit den Knoten korrespondieren, eingezeichnet und mit k_i bezeichnet. Für die Kante zwischen k_1 und k_2 sind zudem die Punkte eingezeichnet, die den Verlauf des Gefäßes angeben und als Informationen der Kante gespeichert sind.

der meist feinen Gefäße gespeichert sind. Diese Informationen enthalten unter anderem die Lage dieser Punkte im Raum. Zudem liegen Kanteninformationen vor. Diese enthalten zum einen als Information die beiden Knoten, zwischen denen die Kante verläuft. Zum anderen - und gerade für uns wichtig - enthält die Kante des Baumes Informationen über den Verlauf des Gefäßes zwischen den beiden Punkten. Diese sind über eine Ansammlung von aufeinanderfolgenden Punkten im Gefäß mittels Koordinaten gegeben. So können auch Krümmungen zwischen zwei Verzweigungspunkten repräsentiert werden. Abbildung 4.21 veranschaulicht die gegebenen Daten.

Unser Ausgangspunkt befindet sich also im Diskreten. Die Punkte zwischen zwei Endpunkten einer Kante nennen wir auch diskretes Skelett. Diese Punkte sind (nach eventuellem zusätzlichem Vorverarbeitungsschritt) äquidistant verteilt. Dies wird im späteren Verlauf bei der Diskretisierung hilfreich sein. Im Folgenden werden wir für den theoretischen Hintergrund ein kontinuierliches Modell der Gefäße mit deren Längenberechnung vorstellen. Im Hinterkopf sollten wir allerdings berücksichtigen, dass dieses kontinuierliche Modell nicht explizit aufgestellt werden muss, da wir das Optimierungsproblem wiederum im Diskreten lösen werden. Zudem ist es offensichtlich, dass es bei der Längenerhaltung der Gefäße ausreicht, die Längen zwischen den Verzweigungspunkten einzeln zu betrachten. Daher beschäftigen wir uns mit diesem

deutlich anschaulicheren Fall. Wir sprechen im Weiteren trotzdem vom Gefäß, meinen aber genau gesagt einen Gefäßabschnitt. Zur kontinuierlichen Beschreibung der Gefäße und Gefäßlängen benutzen wir Kurven, die an dieser Stelle kurz eingeführt werden sollen.

Kurven Durch die vereinfachte Betrachtung der Gefäße betrachten wir hier eine spezielle Gruppe von Kurven. Wir orientieren uns in diesem Abschnitt an [30].

Wir bezeichnen eine injektive Funktion $\kappa : [0, 1] \rightarrow \mathbb{R}^2$,

$$\kappa(t) := \begin{pmatrix} \kappa_1(t) \\ \kappa_2(t) \end{pmatrix}$$

mit hinreichend glatten $\kappa_1, \kappa_2 : \mathbb{R} \rightarrow \mathbb{R}$, als PARAMETERISIERTE KURVE im \mathbb{R}^2 . Ihre erste und zweite Ableitung ist gegeben durch

$$\kappa'(t) := \begin{pmatrix} \kappa_1'(t) \\ \kappa_2'(t) \end{pmatrix}$$

bzw.

$$\kappa''(t) := \begin{pmatrix} \kappa_1''(t) \\ \kappa_2''(t) \end{pmatrix}.$$

Die (BOGEN-)LÄNGE der Kurve κ ist

$$L(\kappa) := \int_0^1 \|\kappa'(t)\| dt. \quad (4.12)$$

Da der Parameter t der Kurve schlecht greifbar ist und die vorliegenden Punkte äquidistant bezüglich der Länge des Gefäßes verteilt sind, verwenden wir eine Umparameterisierung der Kurve κ nach ihrer Bogenlänge. Wir definieren dazu die Funktion $B : [0, 1] \rightarrow [0, L(\kappa)]$ mit

$$B(t) = \int_0^t \|\kappa'(x)\| dx.$$

B besitzt die Umkehrabbildung $B^{-1} : [0, L(\kappa)] \rightarrow [0, 1]$. Nun können wir die Funktion $\kappa_B : [0, L(\kappa)] \rightarrow \mathbb{R}^2$,

$$\kappa_B(t) := \kappa \circ B^{-1}(t) \quad (4.13)$$

betrachten, die uns die Koordinate eines Punktes auf der Kurve nach der Bogenlänge angibt. Mit der Kettenregel können wir nachrechnen, dass $\|\kappa_B'(t)\| = 1$ gilt. Zudem ergibt sich durch die Umparameterisierung für die Krümmung der Kurve [59]

$$K(\kappa_B(t)) := \|\kappa_B''(t)\|. \quad (4.14)$$

Diese Form werden wir für die Regularisierung nutzen.

Modell des Gefäßbaums Betrachten wir nun die schematische Darstellung des Gefäßes in Abbildung 4.22. Im Folgenden wollen wir das Skelett als Menge S einführen und ein Längenmaß auf dieser definieren.

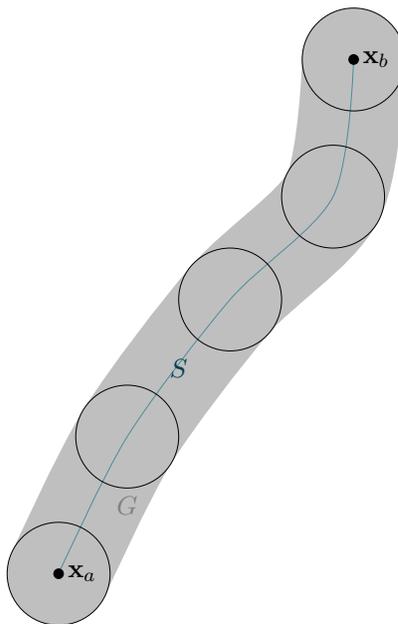


Abbildung 4.22.: Schematische Skizze eines Gefäßes mit dessen Skelett S , dessen Länge durch eine Kurve mit den Endpunkten \mathbf{x}_a und \mathbf{x}_b beschrieben werden kann.

Sei $G \subset \Omega$ eine kurven-zusammenhängende, abgeschlossene Menge und ∂G deren Rand. Kurven-zusammenhängend bedeutet dabei, dass für jedes Paar von Punkten $\mathbf{x}, \mathbf{y} \in G$ eine Kurve $\kappa : [0, 1] \rightarrow G$ existiert mit $\kappa(0) = \mathbf{x}$ und $\kappa(1) = \mathbf{y}$. Zudem sei $\text{dist} : \mathbb{R}^2 \rightarrow \mathbb{R}$ die Funktion, die den minimalen euklidischen Abstand eines Punktes $\mathbf{x} \in G$ zum Rand ∂G misst. Dann wird die zentrale Linie des Gefäßes beschrieben durch

$$S = \{\mathbf{x} \in G : \exists \mathbf{y}, \mathbf{z} \in \partial G, \mathbf{y} \neq \mathbf{z}, \text{dist}(\mathbf{x}) = \|\mathbf{x} - \mathbf{y}\|_2 = \|\mathbf{x} - \mathbf{z}\|_2\} \quad (4.15)$$

und SKELETT genannt [80]. S ist also die Menge der Mittelpunkte aller Kreise mit maximalem Radius, die in G hineingelegt werden können. Wir nehmen im Folgenden an, dass auch S kurven-zusammenhängend ist. In den betrachteten Bildern soll die Menge G ein Gefäß und die Menge S dessen Skelett darstellen.

Die Länge des Gefäßes fassen wir nun auf als Länge $L(\kappa)$ der Kurve $\kappa : [0, 1] \rightarrow S$, die an einem Ende des Gefäßes bei $\kappa(0) = \mathbf{x}_a$ startet und am anderen Ende bei $\kappa(1) = \mathbf{x}_b$ endet, d. h.

$$d_S(\mathbf{x}_a, \mathbf{x}_b) := L(\kappa). \quad (4.16)$$

Durch die geforderte Injektivität und Glattheit ist diese Kurve eindeutig. Dieser Aspekt ist ebenfalls in Abbildung 4.22 veranschaulicht.

4.7.3. Längenerhaltungsnebenbedingungen

Im vorangegangenen Teil haben wir das Skelett S unseres betrachteten Gefäßbaumes kennengelernt und dessen Längenbeschreibung angegeben. Diese Längenbeschreibung soll nun zur Formulierung der Längenerhaltungsnebenbedingungen genutzt werden. Die Forderung besagt, dass die Länge der (Gefäß-)Skelette und damit der Abstand zwischen korrespondierenden Punkten in der prä- und postoperativen Aufnahme erhalten bleibt. Es bleibt jedoch die Frage nach den korrespondierenden Punkten. Durch die Anwendung des Euler Ansatzes für die Deformation φ betrachten wir die Punkte in einem festen Rahmen und nutzen die Information, woher ein Punkt $\mathbf{x} \in S$ kommt (vgl. Abschnitt 3.4). Dadurch gilt für einen Punkt im deformierten Bild $\bar{\mathbf{x}} = \varphi^{-1}(\mathbf{x})$. Dies bedeutet für unsere Längenerhaltung, dass die Längen zwischen zwei festen aber beliebigen Punkten $\mathbf{x}, \mathbf{y} \in S$ und $\varphi^{-1}(\mathbf{x}), \varphi^{-1}(\mathbf{y})$ betrachtet werden müssen. Wir wollen uns dies an einem einfachen Beispiel veranschaulichen.

Betrachten wir ein Bild $\mathcal{T} : \Omega \rightarrow \mathbb{G}$ mit einem Liniensegment, das in x_1 -Richtung verläuft. Auf dieses Bild wenden wir eine affin-lineare Deformation an mit

$$\varphi(\mathbf{x}) = \begin{pmatrix} 1.5 & 0 \\ 0 & 1 \end{pmatrix} \mathbf{x}. \quad (4.17)$$

Wir haben also eine Skalierung in x_1 -Richtung um den Faktor 1.5. Wir können dieses Szenario in Abbildung 4.23 betrachten. Wie bereits beschrieben (vgl. Abschnitt 3.4), verhält sich die Deformation kontraintuitiv und das Liniensegment wird nach der Transformation kürzer. Für die beiden Endpunkte des Liniensegments, $\mathbf{x}_a = (64, 128)^T$ und $\mathbf{x}_b = (192, 128)^T$, haben wir vor der Deformation einen euklidischen Abstand von $\|\mathbf{x}_a - \mathbf{x}_b\|_2 = 128$. Berechnen wir nun den Abstand zwischen den mittels φ transformierten Punkten erhalten wir $\|\varphi(\mathbf{x}_a) - \varphi(\mathbf{x}_b)\|_2 = 192$. Die Länge müsste sich also vergrößert haben. Wie wir jedoch in Abbildung 4.23 beobachten konnten, ist genau das Gegenteil der Fall. Die gesuchte Deformation ist φ^{-1} mit $\|\varphi^{-1}(\mathbf{x}_a) - \varphi^{-1}(\mathbf{x}_b)\|_2 = 85.3333$. Die Längenunterschiede werden in dieser Arbeit daher mit

$$c_L(\mathbf{x}, \mathbf{y}; \varphi^{-1}, S) := d_S(\mathbf{x}, \mathbf{y})^2 - d_S(\varphi^{-1}(\mathbf{x}), \varphi^{-1}(\mathbf{y}))^2, \quad (4.18)$$

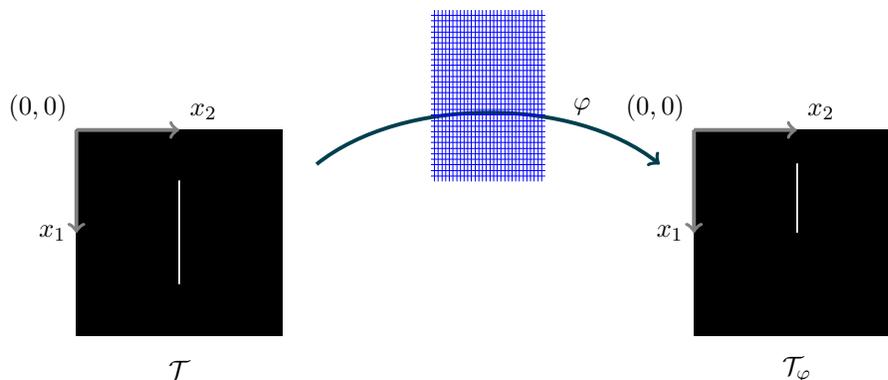


Abbildung 4.23.: Auf das Template wird eine affin-lineare Deformation φ angewendet, hier bestehend aus einer Skalierung in x_1 -Richtung um den Faktor 1.5. Intuitiv nimmt man an, dass die Linie also gestreckt wird. Das deformierte Template \mathcal{T}_φ wird aber offensichtlich gestaucht.

als Nebenbedingung in das Modell eingeführt. Wir betrachten hier die quadratische Differenz der Längen von prä- und postoperativen Daten, um größere Abweichung stärker zu bestrafen.

Die Berechnung der Längenerhaltungsnebenbedingungen benötigt Kenntnis über die inverse Deformation φ^{-1} im Bereich des Skeletts S . Wissen über die inverse Deformation wollen wir im nächsten Abschnitt erlangen und ebenfalls als Nebenbedingung in unser Modell einbinden.

4.7.4. Inversitätsforderung

Zur Berechnung der inversen Deformation gibt es verschiedene Ansätze in der Literatur. Christensen und Johnson [18] berechnen während der Registrierung gemeinsam die Transformation φ und dessen Inverse φ^{-1} , um das Registrierungsergebnis zu verbessern. Durch das gleichzeitige Optimieren der Transformation und ihrer Inversen haben wir allerdings auch die Dimension unseres Problems verdoppelt. Des Weiteren gibt es Verfahren, die für eine gegebene Transformation dessen Inverse berechnen [23, 16]. Bei unserem Problem müsste dies allerdings in jedem Iterationsschritt geschehen. Zusätzlich läuft der Prozess alternierend mit der eigentlichen Optimierung. Sinnvoller wäre es, die beiden Probleme zu kombinieren und als eines aufzufassen. Da wir allerdings nur an der glatten inversen Transformation

$\varphi^{-1} : \mathbb{R}^2 \rightarrow \mathbb{R}^2$ auf dem Skelett des Gefäßbaumes S interessiert sind, fordern wir Kenntnis der inversen Deformation nur an diesen Stellen, d. h.

$$\varphi(\varphi^{-1}(\mathbf{x})) = \mathbf{x} \quad \forall \mathbf{x} \in S. \quad (4.19)$$

Wir sprechen von der Forderung nach einer PUNKTWEISEN INVERSEN. Dieser Ausdruck hat seine Begründung in der späteren Optimierung auf einem diskreten Gitter, bei dem die inverse Transformation φ^{-1} nur an einer diskreten Menge an Punkten bestimmt wird, die deutlich kleiner ist als die der Transformation φ . Da die inverse Transformation aber nicht bekannt ist, erweitern wir unser Optimierungsproblem um die Unbekannte ψ . Für diese soll

$$c_I(\mathbf{x}; \varphi, \psi) := \varphi(\psi(\mathbf{x})) - \mathbf{x} = 0 \quad \forall \mathbf{x} \in S \quad (4.20)$$

gelten.

4.7.5. Vorverarbeitung

Um eine gute Vergleichbarkeit mit der volumenerhaltenden Registrierung aus dem vorherigen Abschnitt zu erhalten, verwenden wir dieselbe Vorverarbeitung und arbeiten auf den segmentierten Gefäßen (vgl. Abschnitt 4.6).

4.7.6. Distanzmaß

Analog zur Volumenerhaltung (vgl. Abschnitt 4.6) verwenden wir wiederum das SSD (Gleichung (3.5)) als Distanzmaß, d. h.

$$\mathcal{D}(\mathcal{R}, \mathcal{T}; \varphi) = \frac{1}{2} \int_{\Omega} (\mathcal{T}(\varphi(\mathbf{x})) - \mathcal{R}(\mathbf{x}))^2 \, d\mathbf{x}. \quad (4.21)$$

4.7.7. Regularisierung des Funktionals

Wie bei der volumenerhaltenden Registrierung wollen wir eine hohe, globale Glattheit bevorzugen. Wir wählen deshalb wiederum die curvature Regularisierung (siehe Abschnitt 3.6.1) für die Deformation φ . Zusätzlich zu der globalen Glattheit verlangen wir allerdings noch eine lokale Glattheit auf den Skeletten der Gefäßbäume. Wir fordern, dass deren Krümmung möglichst gering ist. Praktisch hat sich gezeigt, dass eine einfache Regularisierung nicht zu den gewünschten Anforderungen führt. Wir erweitern den curvature-Regularisierungsterm daher um

$$\mathcal{S}(\psi) = \int_0^{L(\kappa)} \|K(\kappa_B(t))\|^2 \, dt.$$

Aufgrund der Anschaulichkeit und späteren Diskretisierung nehmen wir hier die Umparameterisierung nach der Bogenlänge vor. Dieser Term ist nur für die Regularisierung der punktweisen Inversen zuständig und kann damit lokal wirken.

Es ergibt sich als Regularisierungsterm für die Deformation und die Approximation ihrer Inversen

$$\mathcal{S}(\varphi, \psi) = \frac{1}{2} \int_{\Omega} \sum_{i=1}^2 \left(\Delta \left(\varphi_i(\mathbf{x}) - \varphi_i^{(0)}(\mathbf{x}) \right) \right)^2 dx + \int_0^{L(\kappa)} \|K(\kappa_B(t))\|^2 dt. \quad (4.22)$$

Die Erweiterung der Idee auf Gefäßbäume mit Verzweigungspunkten läuft über eine Summation der Krümmungen der einzelnen Skelette. Dadurch, dass die Verzweigungspunkte in den angrenzenden Skeletten vorhanden sind, wird auch der Zusammenhalt zwischen den einzelnen Gefäßzweigen garantiert.

4.7.8. Kontinuierliche Formulierung der Längenerhaltenden Registrierung

Im vorangehenden Abschnitt haben wir die einzelnen Elemente des Modells vorgestellt. Damit können wir nun das Registrierungsproblem als Ganzes formulieren.

Aus den obigen Überlegungen mit (4.21), (4.22), (4.18) und (4.20) erhalten wir das Zielfunktional

$$\mathcal{J}(\varphi, \psi) = \mathcal{D}(\mathcal{R}, \mathcal{T}; \varphi) + \alpha \mathcal{S}(\varphi, \psi) \quad \text{u. d. NB} \quad c_I(\varphi, \psi) \wedge c_L(\psi). \quad (4.23)$$

4.7.9. Diskretisierung

In diesem Abschnitt gehen wir im Detail auf die Diskretisierung der neuen Nebenbedingungen ein. Die nötigen Modifikationen des Distanzmaßes und des Regularisierers erfolgen kurz zum Abschluss des Abschnitts. Wir arbeiten wiederum auf dem Skelett eines Gefäßbaumes ohne Verzweigungspunkte. Zu Beginn möchten wir zudem noch einmal kurz die Eigenschaften der gegebenen Daten aufführen. Gegeben sind also $l \in \mathbb{N}$ der Bogenlänge nach äquidistant verteilte Punkte $\mathbf{p}_i = \kappa_B(t_i)$, $i \in \{1, \dots, l\}$ auf dem Skelett. Diese definieren einen Polygonzug, dessen Ausgangslänge mit

$$L(\kappa_B) \approx \sum_{i=1}^{l-1} \|\kappa_B(t_{i+1}) - \kappa_B(t_i)\|_2 \quad (4.24)$$

als Approximation an die Skelettlänge bekannt ist. Diese Darstellung wollen wir im Folgenden zur Aufstellung der längenerhaltenden Nebenbedingungen nutzen. Als Anmerkung sei erwähnt, dass die Länge $L(\kappa)$ (Gleichung (4.12)) der Grenzwert der

Längen der Polygonzüge für $l \rightarrow \infty$ ist (siehe [30]). Ein Beispiel ist in Abbildung 4.24 zu finden.

Da wir die Invertierungsanforderungen c_I für die Längenerhaltungsnebenbedingungen c_L benötigen, starten wir mit diesen. Die Ausführungen erfolgen für den zweidimensionalen Fall. Die Übertragung in den dreidimensionalen Fall erfolgt analog.

Die Diskretisierte Punktweise Inverse Die Forderung auf dem Polygonzug verlangt, dass $\varphi(\psi(\mathbf{p}_i)) = \mathbf{p}_i$, $i \in \{1, \dots, l\}$ mit $\mathbf{p}_i = (p_i^1, p_i^2)^T$. Sei dann $\boldsymbol{\psi}_i \approx \varphi^{-1}(\mathbf{p}_i)$, $\boldsymbol{\psi}_i = (\psi_i^1, \psi_i^2)$, die diskretisierte Approximation an die Inverse, die es zu optimieren gilt. Wir führen zunächst analog zur diskreten Transformation $\boldsymbol{\varphi}$ den Vektor

$$\boldsymbol{\psi} = \begin{pmatrix} \boldsymbol{\psi}^1 \\ \boldsymbol{\psi}^2 \end{pmatrix} \in \mathbb{R}^{2l}, \quad \boldsymbol{\psi}^i = (\psi_1^i \dots \psi_l^i)^T, \quad i \in \{1, 2\},$$

als Diskretisierung von der approximierten Inversen ψ zu den Punkten des Polygonzuges \mathbf{p}_i , sowie

$$\mathbf{p} = \begin{pmatrix} \mathbf{p}^1 \\ \mathbf{p}^2 \end{pmatrix} \in \mathbb{R}^{2l}, \quad \mathbf{p}^i = (p_1^i \dots p_l^i)^T, \quad i \in \{1, 2\},$$

für die Punkte des Polygonzuges ein. Zur besseren Orientierung stellt Abbildung 4.24 die folgenden Berechnungen grafisch dar. Gegeben seien nun $\boldsymbol{\varphi}, \boldsymbol{\psi}$ approximativ. Durch die Interpolation

$$\begin{aligned} [\mathbf{p}_{\boldsymbol{\psi}}^1, \mathbf{d}\mathbf{p}_{\boldsymbol{\psi}}^1] &= \text{inter}(\boldsymbol{\psi}, \boldsymbol{\varphi}^1, 3) \\ [\mathbf{p}_{\boldsymbol{\psi}}^2, \mathbf{d}\mathbf{p}_{\boldsymbol{\psi}}^2] &= \text{inter}(\boldsymbol{\psi}, \boldsymbol{\varphi}^2, 3) \end{aligned} \tag{4.25}$$

erhalten wir die Approximation von $\varphi(\psi(\mathbf{x}))$ für die Punkte des Polygonzuges. Wir schreiben dann

$$\mathbf{p}_{\boldsymbol{\psi}} = \begin{pmatrix} \mathbf{p}_{\boldsymbol{\psi}}^1 \\ \mathbf{p}_{\boldsymbol{\psi}}^2 \end{pmatrix}.$$

Die Interpolation liefert uns zudem in $\mathbf{d}\mathbf{p}_{\boldsymbol{\psi}}^1, \mathbf{d}\mathbf{p}_{\boldsymbol{\psi}}^2$ die partiellen Ableitungen unserer Nebenbedingungen bezüglich $\boldsymbol{\psi}$. Die Diskretisierung der Invertierungsnebenbedingungen (4.20) ist dann gegeben durch

$$\mathbf{c}_I = \mathbf{p} - \mathbf{p}_{\boldsymbol{\psi}},$$

da bei Gleichheitsforderung komponentenweise Gleichheit gelten muss. Für die Ableitung $\mathbf{d}\mathbf{c}_I \in \mathbb{R}^{(l-1) \times 2(n+l)}$ schreiben wir

$$\mathbf{d}\mathbf{c}_I = \begin{pmatrix} \frac{\partial \mathbf{c}_I}{\partial \boldsymbol{\varphi}^1} & \frac{\partial \mathbf{c}_I}{\partial \boldsymbol{\varphi}^2} & \frac{\partial \mathbf{c}_I}{\partial \boldsymbol{\psi}^1} & \frac{\partial \mathbf{c}_I}{\partial \boldsymbol{\psi}^2} \end{pmatrix}.$$

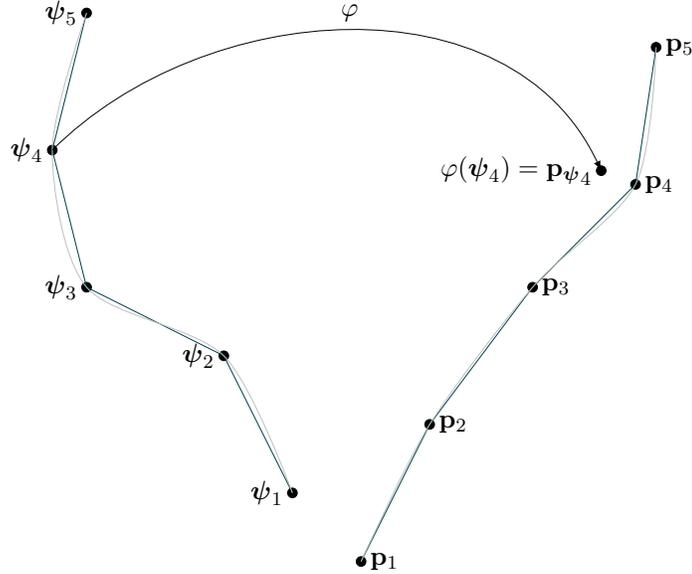


Abbildung 4.24.: Diskrete Situation zur Berechnung der Nebenbedingungen der punktweisen Inversen. Wir sehen die Punkte \mathbf{p}_i des gegebenen Skelettes, sowie die Punkte ψ der korrespondierenden approximierten Inversen. Für den Index $i = 4$ ist zusätzlich die Anwendung der Transformation φ auf die Approximation der Inversen eingezeichnet. Diese ist mit \mathbf{p}_{ψ_4} bezeichnet und sollte im Optimalfall direkt auf den Punkt \mathbf{p}_4 abbilden. Die Nebenbedingungen c_I betrachten die Differenz zwischen diesen beiden Punkten.

Wie oben erwähnt, haben wir die partiellen Ableitungen $\frac{\partial c_I}{\partial \psi^1}, \frac{\partial c_I}{\partial \psi^2}$ gegeben über die Interpolation (4.25), wir schreiben

$$d\mathbf{p}_\psi = \begin{pmatrix} d\mathbf{p}_\psi^1 \\ d\mathbf{p}_\psi^2 \end{pmatrix}.$$

Im Folgenden beschäftigen wir uns mit der Ableitung der Nebenbedingung nach der diskreten Transformation φ . Dabei orientieren wir uns an dem Ansatz im Kontext der Landmarkenregistrierung aus [76] und betrachten einen anderen Zugang zur Auswertung von $\varphi(\psi_i)$, bei dem sich die Ableitung nach der Transformation φ direkt ergibt. Da die Punkte des Polygonzuges im Allgemeinen nicht auf dem Gitter unserer Diskretisierung liegen, nutzen wir eine diskrete Punktauswertefunktion (engl. point evaluation function, PEF).

Starten wir im eindimensionalen Fall. Sei $X^{cc}(\Omega, m)$ ein zellzentriertes Gitter mit n Punkten und $\psi_i \in \mathbb{R}$ die Approximation der Invertierungsforderung $\varphi^{-1}(\mathbf{p}_i)$. Sei $\mathbf{P}_d \in \mathbb{R}^{l \times n}$. Wir suchen eine lineare Darstellung

$$\mathbf{P}_d \bar{\mathbf{x}}^{cc} = \psi. \quad (4.26)$$

Die Idee entspricht der Gewichteberrechnung bei der linearen Interpolation. Im Gegensatz zu (4.25) wählen wir hier der Einfachheit halber eine lineare Darstellung und nehmen Unterschiede in der Glattheit dabei in Kauf. Sei $\xi_i = \max\{j \in \mathbb{Z} : \mathbf{x}_j \leq \psi_i, \mathbf{x}_j \in X^{cc}(\Omega, m)\}$. Die Matrix $\mathbf{P}_d = (p_{ij})$ ist dann gegeben durch

$$p_{ij} = \begin{cases} \frac{1}{h} (\psi_i - \mathbf{x}_{\xi_i}) & j = \xi_i, \\ \frac{1}{h} (\mathbf{x}_{\xi_i} - \psi_i) & j = \xi_i + 1, \\ 0 & \text{sonst.} \end{cases}$$

Die Übertragung in den zweidimensionalen Fall geschieht komponentenweise. Es folgt

$$\mathbf{P} = \begin{pmatrix} \mathbf{P}_d & \\ & \mathbf{P}_d \end{pmatrix} \in \mathbb{R}^{2l \times 2n}$$

mit $n = m_1 m_2$. Damit erhalten wir die Darstellung unserer Approximation ψ über

$$\mathbf{P} \bar{\mathbf{x}}^{cc} = \psi.$$

Für die transformierten Punkte gilt folglich

$$\varphi(\psi) \approx \mathbf{P} \varphi.$$

Wir erhalten die partielle Ableitung nach der diskreten Transformation φ durch

$$\begin{pmatrix} \frac{\partial \mathbf{c}_I}{\partial \varphi^1} & \frac{\partial \mathbf{c}_I}{\partial \varphi^2} \end{pmatrix} = \mathbf{P}$$

und mit

$$d\mathbf{c}_I = (\mathbf{P} \quad d\mathbf{p}_\psi)$$

die Ableitung der Nebenbedingungen bezüglich der punktweisen Inversen.

Die Diskretisierten Längenerhaltungsnebenbedingungen Wir betrachten die längenerhaltenden Nebenbedingungen c_L auf einem zellzentrierten Gitter $X^{cc}(\Omega, m)$ für den zweidimensionalen Fall auf dem Polygonzug, der über über die Punkte $\mathbf{p}_1, \dots, \mathbf{p}_l$ gegeben ist. Die längenerhaltende Forderung (4.18) wird approximiert über die Längenerhaltung der linearen Segmente aus (4.24), d. h.

$$\mathbf{c}_i = \|\mathbf{p}_{i+1} - \mathbf{p}_i\|_2^2 - \|\psi_{i+1} - \psi_i\|_2^2, \quad i = 1, \dots, l-1.$$

Die einzelnen Längenerhaltungsnebenbedingungen fassen wir zusammen in dem Vektor

$$\mathbf{c}_L = \begin{pmatrix} \mathbf{c}_1 \\ \mathbf{c}_2 \\ \vdots \\ \mathbf{c}_{l-1} \end{pmatrix} = \begin{pmatrix} \|\mathbf{p}_2 - \mathbf{p}_1\|_2^2 - \|\boldsymbol{\psi}_2 - \boldsymbol{\psi}_1\|_2^2 \\ \|\mathbf{p}_3 - \mathbf{p}_2\|_2^2 - \|\boldsymbol{\psi}_3 - \boldsymbol{\psi}_2\|_2^2 \\ \vdots \\ \|\mathbf{p}_l - \mathbf{p}_{l-1}\|_2^2 - \|\boldsymbol{\psi}_l - \boldsymbol{\psi}_{l-1}\|_2^2 \end{pmatrix}.$$

Wir führen nun eine geeignete Notation ein, die es erlaubt, die Nebenbedingungen sowie die Ableitung kompakt und übersichtlich darzustellen. Grundlage dafür ist die Berechnung der Differenzen zwischen benachbarten Punkten. Wir betrachten dafür wieder anfänglich den eindimensionalen Fall mit dem Vektor $\mathbf{p} = (p_1, \dots, p_l)^T \in \mathbb{R}^l$. Ähnlich zur Berechnung der Ableitung des NGF (3.39) über finite Differenzen führen wir die Matrix

$$\mathbf{D} = \begin{pmatrix} 1 & -1 & & & \\ & 1 & -1 & & \\ & & \ddots & \ddots & \\ & & & 1 & -1 \end{pmatrix} \in \mathbb{R}^{(l-1) \times l}$$

ein, die uns mit \mathbf{Dp} die Differenzen $(p_{i+1} - p_i)$ liefert. Die Vektordarstellung der dazugehörigen Quadrierungen erhalten wir dann mit

$$\begin{pmatrix} (p_2 - p_1)^2 \\ (p_3 - p_2)^2 \\ \vdots \\ (p_l - p_{l-1})^2 \end{pmatrix} = \text{diag}(\mathbf{Dp})\mathbf{Dp}.$$

Im zweidimensionalen Fall können die Nebenbedingungen geschrieben werden als

$$\mathbf{c}_L = \text{diag}(\mathbf{Dp}^1)\mathbf{Dp}^1 + \text{diag}(\mathbf{Dp}^2)\mathbf{Dp}^2 - (\text{diag}(\mathbf{D}\boldsymbol{\psi}^1)\mathbf{D}\boldsymbol{\psi}^1 + \text{diag}(\mathbf{D}\boldsymbol{\psi}^2)\mathbf{D}\boldsymbol{\psi}^2).$$

Mit Hilfe der Kettenregel erhalten wir als Ableitung

$$d(\text{diag}(\mathbf{Dp}^i)\mathbf{Dp}^i) = 2 \text{diag}(\mathbf{Dp}^i)\mathbf{D}, \quad i \in \{1, 2\}.$$

Es ergibt sich mit den diskreten partiellen Ableitungen

$$\begin{aligned} d\mathbf{c}_L^\varphi &:= \frac{\partial \mathbf{c}_L}{\partial \boldsymbol{\varphi}^1} = \mathbf{0} \in \mathbb{R}^{(l-1) \times 2n} \quad \text{und} \\ d\mathbf{c}_L^\psi &:= \frac{\partial \mathbf{c}_L}{\partial \boldsymbol{\psi}^1} = (-2 \text{diag}(\mathbf{D}\boldsymbol{\psi}_1)\mathbf{D} - 2 \text{diag}(\mathbf{D}\boldsymbol{\psi}_2)\mathbf{D}) \in \mathbb{R}^{(l-1) \times 2l} \end{aligned}$$

die diskrete Ableitung der längenerhaltenden Nebenbedingung

$$d\mathbf{c}_L = \begin{pmatrix} d\mathbf{c}_L^\varphi & d\mathbf{c}_L^\psi \end{pmatrix}.$$

Diskretisierung von Distanzmaß und Regularisierer Die Diskretisierung für das Distanzmaß und den Regularisierer erfolgt auf einem zellzentrierten Gitter und ist dem Registrierungskapitel 3 zu entnehmen. Da wir nun allerdings sowohl über φ als auch ψ optimieren, müssen wir dies in der Ableitung einbeziehen. Es ergibt sich für die beiden ersten Ableitungen

$$\begin{aligned} d\mathbf{D} &= (d\mathbf{D}^\varphi \quad \mathbf{0})^T, \\ d\mathbf{S} &= \begin{pmatrix} \mathbf{A}^{\text{curv}} & \\ & \mathbf{A}^\psi \end{pmatrix} \begin{pmatrix} \varphi \\ \psi \end{pmatrix} \end{aligned}$$

und

$$\begin{aligned} d^2\mathbf{D} &= \begin{pmatrix} d^2\mathbf{D}^\varphi & \\ & \mathbf{0} \end{pmatrix}, \\ d^2\mathbf{S} &= \begin{pmatrix} \mathbf{A}^{\text{curv}} & \\ & \mathbf{A}^\psi \end{pmatrix} \end{aligned}$$

für die zweiten Ableitungen. Hierbei ist $\mathbf{D}^\varphi \in \mathbb{R}^{n \times 2n}$ die bekannte Ableitung des Distanzmaßes nach der diskreten Transformation φ (Abschnitt 3.5, Gleichung (3.34)). Der Anteil, der auf ψ wirkt, wird Null gesetzt, da die Approximation der Inversen nicht in die Distanzmaßberechnung mit einfließt. Für die Diskretisierung des Anteils von ψ im Regularisierungsterm (4.22) folgen wir Implementierungen im Bereich der Segmentierung und verwenden finite Differenzen (siehe Gleichung 3.17) zur Erstellung der Regularisierungsmatrix $\mathbf{A}^\psi \in \mathbb{R}^{2l \times 2l}$. Wichtige Grundlage hierbei ist die nach der Bogenlänge äquidistante Verteilung der Diskretisierungspunkte. Siehe hierzu z. B. [54]. Die Regularisierungsmatrix $\mathbf{A}^{\text{curv}} \in \mathbb{R}^{2n \times 2n}$ bezüglich der Deformation φ wurde in Abschnitt 3.6.1, Gleichung (3.46) eingeführt.

Damit sind die nötigen Grundlagen für die längenerhaltende Registrierung gelegt und wir kommen zum praktischen Teil des Abschnitts.

4.7.10. Vergleich der Unrestringierten Optimierungsverfahren

Wie im Fall der volumenerhaltenden Registrierung wollen wir auch bei der längenerhaltenden Registrierung die Auswirkung der Wahl des unrestringierten Optimierungsverfahrens im Augmented Lagrangian Ansatz betrachten. Hierfür testen wir die Verfahren wieder an den Datensätzen 1-3, vorgestellt in Abschnitt 4.4. Zuerst betrachten wir den Einsatz des Gauß-Newton-Verfahrens in dem Multilevel Ansatz mit Bildgrößen $m^{\text{grob}} = (16, 16, 16)$ und $m^{\text{fein}} = (32, 32, 32)$ mit Startstrafparametern $\mu^{(0)} = 10^4$ bzw. $\mu^{(0)} = 10^6$. Für $\lambda^{(0)}$ erwies sich der Nullvektor als geeignete

Wahl, da sich die einzelnen Komponenten stark unterschieden und Extrema im positiven und negativen Bereich aufwiesen. Eine Prolongation des Lagrange-Vektors musste nicht durchgeführt werden, da wir immer die gleiche Anzahl an Nebenbedingungen betrachtet haben. Die Wahl des Regularisierungsparameters für Datensatz 1 fiel auf $\alpha = 10^8$, für die anderen beiden Datensätze jeweils auf $\alpha = 10^9$. Die Ergebnisse sind in Abbildung 4.25 visualisiert.

Wir sehen, dass diese sich im Rahmen der volumenerhaltenden Registrierung befinden. Im Schnitt konnten sie sogar leicht verbessert werden. Die relative Distanz für den Datensatz 1 erlangt einen Wert von $d_{\text{rel}} = 0.2519$ nach der maximalen Anzahl an Iterationschritten mit einer maximalen Längenänderung von 0.0706mm. Dies ist völlig ausreichend, betrachtet man die Voxelgröße $h = (6.9066, 5.4880, 5.8750)$ auf der höchsten Auflösung - der Toleranzparameter τ_c für die Nebenbedingungen wurde dementsprechend gewählt. Bei Datensatz 2 wurde $d_{\text{rel}} = 0.2131$ erzielt bei einer maximalen Längenänderung von 0.1668 mm. Für Datensatz 2 ergibt sich $d_{\text{rel}} = 0.6406$. Die maximale Längenänderung beträgt hierbei 0.1612 mm. Die Laufzeit des Verfahrens hatte im Schnitt einen Faktor von 0.5 im Vergleich zur volumenerhaltenden Registrierung. Damit wurden die beiden gesetzten Ziele erreicht. Besonders die Laufzeitverbesserung konnte zur größten Zufriedenheit erreicht werden, während die Distanzmaßauswertung auf dem Gefäßbaum gleichwertige Resultate liefert. Bleibt der Vergleich innerhalb des Verfahrens mit dem L-BFGS getriebenen Augmented Lagrangian Ansatz.

Hierbei stellt sich bereits zu Beginn heraus, dass das Verfahren stark abhängig von der Parameterwahl ist und für die meisten Wahlen keine gleichwertigen Ergebnisse liefert. Bereits bei der volumenerhaltenden Registrierung hatten wir feststellen müssen, dass das L-BFGS-Verfahren weniger robust als das Gauß-Newton-Verfahren ist. Hier zeigt sich diese Eigenschaft jedoch deutlich stärker. Die in Abbildung 4.26 erzielten Ergebnisse gehen ebenfalls auf einen Multilevel Ansatz zurück mit den Bildgrößen $m^{\text{grob}} = (16, 16, 16)$ und $m^{\text{fein}} = (32, 32, 32)$. Die Startstrafparameter mussten mit $\mu^{(0)} = 10^7$ bzw. $\mu^{(0)} = 10^8$ bereits zu Beginn wieder sehr hoch gewählt werden. Eine höher Wahl führte zu instabilen Iterationsverläufen. Der Regularisierungsparameter war für alle drei Testdatensätze $\alpha = 5 \cdot 10^8$. Als maximale Anzahl der äußeren Iterationsschritte wurde 35 auf dem groben und 15 auf dem feinen Level gewählt. Es hat sich gezeigt, dass sich der Nullvektor für die Startschätzung des Lagrange Vektors $\lambda^{(0)}$ am robustesten verhält.

Wie bereits angedeutet erhalten wir deutlich schlechtere Ergebnisse nach der maximalen Anzahl an Iterationschritten. Die Gefäße werden nicht gut aufeinander ausgerichtet. Dies spiegelt sich auch in der relativen Distanz (4.11) wider. Für Datensatz 1 erhalten wir $d_{\text{rel}} = 0.8613$ bei einer maximalen Längenänderung von 0.3360 mm. Diese Änderung ist bezüglich der Voxelgröße zwar noch verträglich,

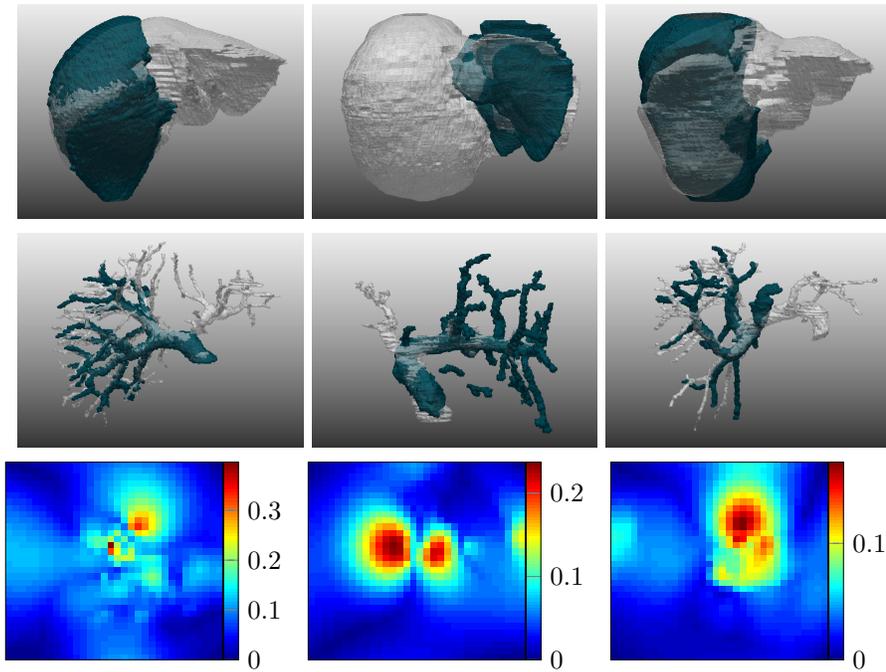


Abbildung 4.25.: Ergebnisse der längenerhaltenden Registrierung mit dem Gauß-Newton-Verfahren innerhalb des Augmented Lagrangian Ansatzes. Die Berechnungen erfolgten auf den Segmentierungen des Gefäßbaumes PV bei einer höchsten Auflösung der Transformation von $m^{\text{fein}} = (32, 32, 32)$. Im Vergleich zur rigiden Registrierung konnten wie bei der volumenerhaltenden Registrierung deutliche Verbesserungen erzielt werden. In der untersten Reihe ist die Volumenänderungen über eine Projektion des maximalen Wertes über die erste Dimension dargestellt. Es zeigt sich hier, dass die Transformation faltungsfrei ist.

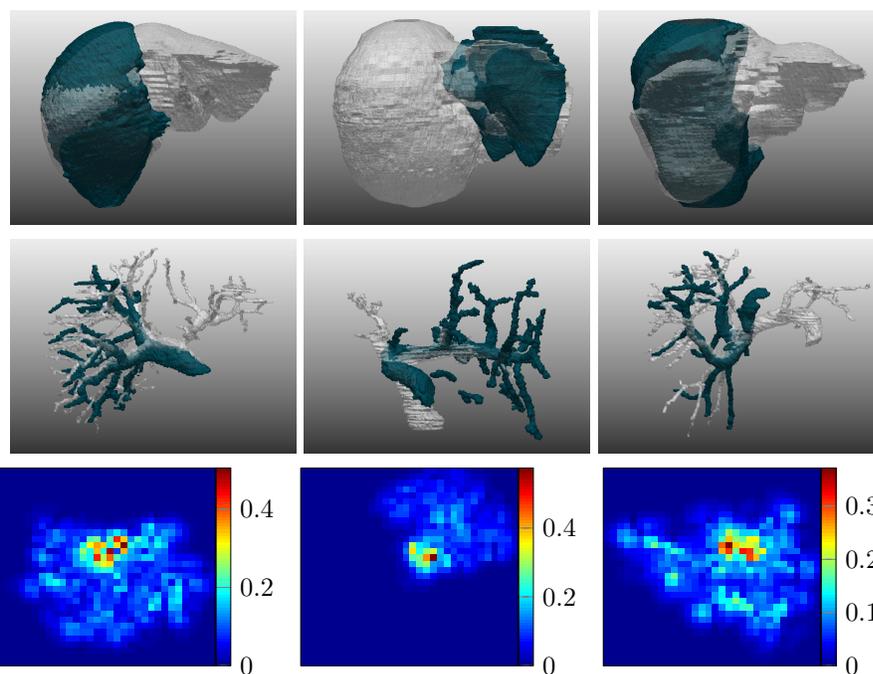


Abbildung 4.26.: Ergebnisse der längenerhaltenden Registrierung mit dem L-BFGS-Verfahren und $\alpha = 2 \cdot 10^8$. Berechnungen erfolgten auf den Segmentierungen des Gefäßbaumes PV bei einer höchsten Auflösung der Transformation von $m^{\text{fein}} = (32, 32, 32)$. Die Überlagerungen der Gefäße sind deutlich schlechter als mit den bereits vorgestellten Verfahren. In der untersten Reihe ist die Volumenänderung über eine Projektion des maximalen Wertes über die erste Dimension dargestellt. Im Vergleich zum Gauss-Newton-Verfahren sind die Volumenänderung deutlich erhöht. Allerdings ist das Ergebnis weiterhin faltungsfrei

jedoch haben wir kaum eine Verbesserung zur rigiden Registrierung. Datensatz 2 erzielt mit $d_{\text{rel}} = 0.9105$ mit einer maximalen Längenänderung von 0.3495 mm genauso wie Datensatz 2 mit $d_{\text{rel}} = 0.9785$ mit einer maximalen Längenänderung von 0.1195 mm eine schwache Verbesserung zur Ausgangssituation.

4.7.11. Fazit und Ausblick

Wir haben im vorangegangenen Abschnitt gesehen, dass die längenerhaltende der volumenerhaltenden Registrierung in Punkto Laufzeit und Genauigkeit der Gefäßbäume überlegen sein kann. Andererseits stellt sich auch heraus, dass es Probleme gibt. Die Stabilität des Verfahrens, insbesondere beim L-BFGS-Verfahren, kann noch deutlich verbessert werden. Dies kann daran liegen, dass die Bedingungen auf einer eindimensionalen Menge gestellt werden, wir uns jedoch in einem dreidimensionalen Raum befinden. Um das Verfahren zu verbessern, können z. B. weitere Regularisierungsmethoden untersucht werden. Die ersten Erfolge rechtfertigen jedoch einen weiteren Blick auf die diese Nebenbedingungen, die viel Potential in dieser Anwendung haben. Vorstellbar ist auch die Wiederverwertung der Invertitätsforderung, z. B. im Bereich der von Christensen vorgeschlagenen "Consistent Image Registration" [18]. Hierbei läuft die Optimierung dann sowohl über die eigentliche Transformation als auch deren Inversen. Im nächsten Kapitel werden wir auch eine andere Möglichkeit des Einsatzes dieser Bedingung kennenlernen. Diese zielt auf bestimmte Regionen beschränkte Nebenbedingungen ab.

4.8. Ausblick der Prä/Post-Registrierung

Zum Abschluss des Kapitels über die Registrierung von prä- und postoperativen Leberdaten wollen wir noch einmal die wichtigsten Punkte resümieren.

Wir haben gesehen, dass die standardmäßige initiale Registrierung mit einem rigiden Transformationsmodell bereits große Probleme aufgibt und keine ausreichende Robustheit aufweist. Um diese Robustheit gewähren zu können, zeigt sich die landmarkenbasierte rigide Registrierung als eine kompetente Wahl. Bereits fünf Landmarkenpaare haben ausgereicht, eine gute Ausrichtung - auch innerhalb der Leber - zu erzielen. Der Knackpunkt bei diesem Ansatz ist jedoch, dass Landmarkenpaare gefunden werden müssen. Bisher wurden diese von Hand gesetzt. Eine automatische Methode würde das gesamte vorgestellte Verfahren automatisieren. Ein solches Verfahren zu entwickeln scheint jedoch gut möglich. So existieren bereits Algorithmen, die Verzweigungspunkte von Gefäßbäumen der Leber oder Lunge automatisch aufeinander abbilden [89, 90, 65, 35, 15]. Hierbei werden die Gefäße

Verfahren	Datensatz 1	Datensatz 2	Datensatz 3
Volumenerhaltend mit GN	0.3761	0.1909	0.7092
Volumenerhaltend mit L-BFGS	0.4144	0.2248	0.7309
Längenerhaltend mit GN	0.2519	0.2131	0.6406
Längenerhaltend mit L-BFGS	0.8613	0.9105	0.9785

Tabelle 4.2.: Tabelle mit den relativen Verbesserungen der Distanz (4.11) für die längen- und volumenerhaltende Registrierung ausgewertet für die drei Testdatensätze. Dabei wird zudem zwischen den unrestringierten Optimierungsverfahren innerhalb des Augmented Lagrangian Ansatzes unterschieden. Hierbei steht GN für das Gauß-Newton-Verfahren und L-BFGS für das L-BFGS-Verfahren. Farbig hinterlegt sind hierbei die besten Werte der einzelnen Datensätze

als Graphen modelliert und es wird mit Hilfe von Attributen eine fehlertolerante Korrespondenz zwischen den Graphenknoten (Verzweigungspunkten) versucht zu finden. In Fall der prä- und postoperativen Registrierung, müsste der postoperative Graph ein Untergraph des präoperativen sein. Durch die Kenntnis der VCI kann zudem eine Startkorrespondenz hergestellt werden. Diese Informationen können auch zusätzlich in der nichtlinearen Hauptregistrierung als Nebenbedingungen genutzt werden [61, 73].

Mit einer robusten Vorregistrierung konnten starke Verbesserungen sowohl mit der volumen- als auch der längenerhaltenden Registrierung gezeigt werden. Tabelle 4.2 gibt einen Überblick über die relativen Verbesserungen der Distanz (4.11) auf dem Gefäßbaum PV. Für die längenerhaltende Registrierung zeigt sich jedoch, dass das L-BFGS-Verfahren keine stabilen Ergebnisse liefert und daher in dieser Konfiguration nicht genutzt werden sollte. Der Vorteil der Längenerhaltung liegt in der Laufzeit der Algorithmen bzw. in der Auswertung der Nebenbedingungen. Während bei der Volumenerhaltung 16^3 bzw. 32^3 Nebenbedingungen gestellt werden, sind diese bei der Längenerhaltung konstant unter 5000. In der Laufzeit zeigt sich dieses in den Testdurchläufen durch eine halbierte Dauer (25 im Gegensatz zu 53 Minuten bei der Registrierung von Patient 1 mit dem Gauß-Newton-Verfahren). Im Punkt der Genauigkeit und Robustheit hebt sich unter den unrestringierten Optimierern das Gauß-Newton-Verfahren hervor.

Basierend auf dem vorgestellten Gerüst zur Registrierung prä- und postoperativer Leberdaten, kann eine Untersuchung verschiedener Distanzmaße noch zu besseren Ergebnissen führen. Bei Datensatz 3 stellt sich heraus, dass die große nichtlineare Deformation der Vene nicht eingefangen werden kann. Hier könnte z. B. die Auswertung der Distanz auf einer Distanzkarte der Gefäße erfolgen. Damit erhöht sich der

Einzugsbereich, um korrespondierende Punkte einzufangen. Auch könnte zusätzlich zum Gefäßbaum der Rest der Leber mit in die Distanzauswertung eingehen, um dadurch auch die Oberflächen besser auszurichten. Im diesem Umfeld bestehen viele weitere Möglichkeiten.

Interessant bei der volumenerhaltenden Registrierung ist zudem die Einschränkung der Nebenbedingungen genau auf den Bereich der Leber, gekoppelt mit einer regionalen Regularisierung. Durch diesen Schritt erhoffen wir uns die Möglichkeit, eine zusammengefallene postoperative Leber auf deren präoperativen Partner zu registrieren. Würden wir in diesem Fall die Nebenbedingungen auf dem gesamten Gebiet stellen mit einheitlicher Regularisierung, kann das Verschwinden von leeren Zwischenräumen nicht modelliert werden.

Wenden wir unseren Blick weg von der Leberresektion, so finden sich zudem ähnliche Anwendungen im Umfeld der Tumorresektion in Gehirnen. Auch hier fehlt ein Teil der Daten. In diesem Fall erscheint es jedoch nicht sinnvoll die Längenerhaltung von Gefäßen zu fordern, sondern sich mit der Volumenerhaltung zu beschäftigen. Prinzipiell ist dieser Ansatz auf kein spezifisches Organ ausgelegt und besitzt daher viele weitere Anwendungsgebiete in der Registrierung prä- und postoperativer Daten.

Wir sehen, dass es viele weitere Möglichkeiten gibt und die Registrierung prä- und postoperativer Daten ein spannendes Gebiet ist, in dem noch viel Potential steckt. Mit dieser Bemerkung wollen wir nun auch dieses Feld verlassen und uns kurz einer weiteren Anwendung zuwenden, in der Nebenbedingungen ein wichtigen Schritt zur Modellierung und Lösung des Problems beitragen.

5. Subtraktionsbildgebung bei Multipler Sklerose

T_2 -gewichtete Subtraktionsbildgebung hat sich als leistungsstarkes Outcome Measure bei Probandenversuchen mit Multipler Sklerose (MS) erwiesen [69]. Es kann sowohl zur Visualisierung und Charakterisierung von Läsionen und deren Volumenänderung als auch zur Begutachtung einer Behandlungseffizienz genutzt werden. Bei der Erstellung der Subtraktionsbilder ist eine genaue Registrierung essentiell. In der Subtraktionsbildgebung sollte eine möglichst hohe Korrespondenz der Strukturen des Gehirns, d. h. der grauen und weißen Substanz etc., erfolgen, so dass die Änderungen der Läsion deutlich hervorgehoben werden. Ist dies nicht der Fall, wird durch eine schlechte Überlappung zusätzliche, störende Information erzeugt. Diese nennen wir im Folgenden Rauschen. Das Rauschen erschwert eine Beurteilung der Bilder, in denen möglichst nur die Läsionsänderungen sichtbar sein sollten. Andernfalls könnten unzureichende Ausrichtungen auf den ersten Blick z. B. als neu entstandene Läsionen interpretiert werden. Abbildung 5.1 zeigt ein typisches, verrauschtes und ein vereinfachtes, optimales Subtraktionsbild.

Um dieses Rauschen zu unterdrücken, wollen wir eine nichtlineare Registrierung nutzen, die auch Bilddetails richtig aufeinander abbildet. Unglücklicherweise stellt sich dabei ein großes Problem, denn eine einfache nichtlineare Registrierung nach dem Standard würde auch die Läsionen möglichst genau aufeinander abbilden. Wir sind aber gerade bei diesen Änderungen daran interessiert sie zu erhalten. In der Literatur sind verschiedene Ansätze zu finden, die sich mit der Registrierung ähnlicher Fälle beschäftigen [9, 71, 55]. Allen Ansätzen ist gemein, dass sie eine Maske des Läsionsbereichs verwenden, die die eigentlichen Informationen darunter ausblenden. Ziel dabei ist es, den Einfluss dieser Region in der Registrierung zu beeinflussen. Keiner dieser Ansätze nutzt jedoch die gegebenen Informationen dieser Region im Algorithmus. Ein weiterer Ansatz, der keine Masken benötigt, nutzt eine Kombination von Registrierung und Segmentierung [4]. Kürzlich wurde allerdings gezeigt, dass dieser Ansatz zu weniger akkuraten Ergebnissen führt und daher der Gebrauch einer Maskierung der Distanzfunktion weiterhin benötigt wird [3].

In dieser Arbeit wollen wir ebenfalls einen Ansatz mit Maskierung der Region vorstellen. Im Gegensatz zu den oben genannten Methoden wollen wir allerdings die

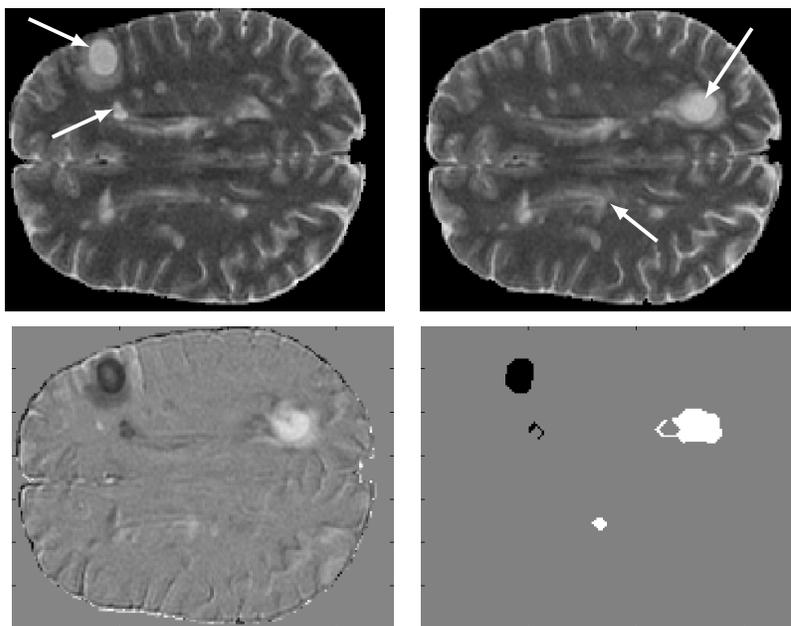


Abbildung 5.1.: Schematische Darstellung eines typischen (links unten) und eines optimalen (rechts unten) Subtraktionsbildes. Die für das optimale Subtraktionsbild vereinfachten betrachteten Läsionen sind dafür im Referenz- (links oben) und Templatebild (rechts oben) mit einem Pfeil markiert. Die aktiven Läsionen stechen deutlich hervor. Das sogenannte Rauschen, verursacht durch unzureichende Ausrichtung der grauen und weißen Substanz, ist dabei komplett unterdrückt. Die inaktiven Läsionen werden perfekt überlagert

Informationen unter dieser Maske im Distanzmaß nutzen und dadurch das Ergebnis der Registrierung zu einer erhöhten Genauigkeit führen. Dabei sollen gerade die Volumenänderungen der Läsion zwischen den beiden zu vergleichenden Zeitpunkten sichtbar gemacht werden. Um den Änderungen der Läsionen zu genügen, führen wir volumenerhaltende Nebenbedingungen im Bereich dieser ein. Dadurch wird eine angemessene Genauigkeit in der umliegenden Region erzielt. Zudem erlaubt die Registrierung keine Abbildung einer Läsion von einem Zeitpunkt auf die korrespondierende Läsion zum anderen Zeitpunkt, durch die aufgetretenen Volumenänderungen unterdrückt werden. Dadurch bleibt für die Kliniker die relevante Information erhalten.

Im Folgenden werden wir einen kurzen Einblick in die Multiple Sklerose und deren Erscheinungsbild in MR-Bildern geben. Nach einem Überblick über den Stand der Technik bei den Registrierungsverfahren gehen wir kurz auf die Vorverarbeitung ein und kommen dann zur Registrierung. Zum Abschluss präsentieren wir beispielhafte Ergebnisse und enden mit einem Ausblick.

5.1. Grundlagen der Multiplen Sklerose

Multiple Sklerose ist eine chronische Krankheit, die das zentrale Nervensystem betrifft. Die Erkrankung tritt besonders bei jungen Personen auf. Der erste Ausbruch erfolgt typischerweise im Alter von 20-40 Jahren. Frauen sind bei einem ungefähren Verhältnis von 2:1 von der Krankheit stärker betroffen als Männer. MS hat am Anfang der Erkrankung meist einen schubförmig remittierenden Verlauf, der Störungen z. B. im senso-motorischen oder visuellen Bereich erzeugt. Diese Störungen werden durch eine Demyelinisierung der weißen Substanz ausgelöst. Die entstehenden Läsionen treten an unterschiedlichen Stellen des zentralen Nervensystems auf, an denen Myelin existiert. Bevorzugt finden wir Läsionen an Rande der Ventrikel und am Corpus Callosum. Die Läsionen weisen verschiedene Verläufe auf. Die Verläufe lassen sich charakterisieren durch Phasen wie die Entzündung, Degeneration des Myelins und der Axone und einer eventuellen Wiederherstellung [64]. Die Größe der Läsionen variiert dabei stark. Die meisten Läsionen sind kleiner als ein Zentimeter im Durchmesser, können aber auch deutlich größer ausfallen. Ihre Form ist meistens rund bzw. oval und deutlich abgegrenzt vom umliegenden Gewebe. Die Läsionen sind gut in T_2 -gewichteten Magentresonanztomographie (MRT)- Aufnahmen sichtbar. Eine Schicht-Aufnahme mit mehreren Läsionen können wir in Abbildung 5.2 betrachten. Läsionen fallen im T_2 -gewichteten MRT durch Hyperintensitäten auf, welche allerdings nicht spezifisch sind. Die Hyperintensitäten können ihre Ursache in Entzündung, Demyelinisierung, Gliose oder Ödemen haben. Bei einer hohen Variabilität können wir 4-5 neue Läsionen pro Jahr im MRT feststellen, wobei nicht alle Läsionen mit klinischen Symptomen einher gehen. Mit fortschreitender Krankheit können neue und wachsende Läsionen gleichzeitig mit schrumpfenden Plaques auftreten. Abbildung 5.3 zeigt beispielhaft einen Verlauf über ein Jahr.

5.2. Stand der Technik

In der Literatur finden wir hauptsächlich Ansätze die ein affin-lineares oder rigides Transformationsmodell im Vorverarbeitungsschritt der Subtraktionsbilderzeugung

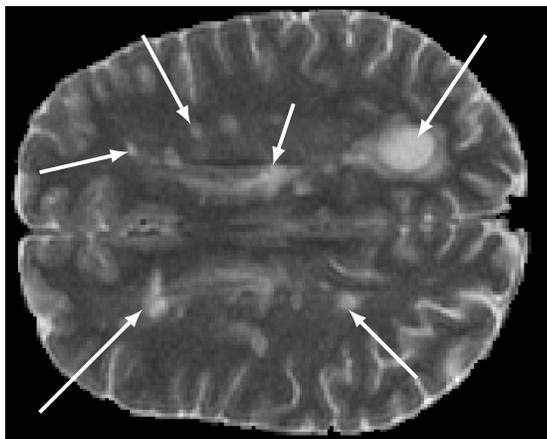


Abbildung 5.2.: Schicht aus einer T_2 -gewichteter MRT-Aufnahme mit diversen Läsionen. Einige der Läsionen sind mit Pfeilen markiert. Die Größe der Läsionen variiert stark, wobei kleine Läsionen deutlich häufiger auftreten. Auch das vermehrte Auftreten an den Ventrikeln ist gut zu erkennen

nutzen [69, 25, 70, 87, 86, 63]. Dies liegt in der nahezu rigiden Natur des Gehirns. Um jedoch ein Subtraktionsbild mit möglichst wenig Rauschen zu erzielen, ist ein lineares Transformationsmodell nicht ausreichend. Vorherige Arbeiten, die sich mit Pathologien und der daraus resultierenden Problematik der Registrierung beschäftigen, kommen vornehmlich aus dem Gebiet der Erstellung von Atlanten. Um das Problem der durch die Pathologie unterschiedlichen Daten zu umgehen, wurden spezielle Registrierungsansätze entwickelt. Eine einfache, aber effektive Methode wurde von Brett et al. 2001 eingeführt [9]. Dabei wird das Distanzmaß im Bereich der Läsion maskiert, um den Einfluss der Läsionsänderung auszublenden. Nachev et al. [71] nutzen dagegen die Symmetrie des Gehirns. Dabei wurde die Information aus der anderen Hemisphäre des Gehirns an die Stelle der Läsion gespiegelt. Hier ist das Auftreten der Läsion nur in einer Hemisphäre von essentieller Bedeutung. Eine ähnliche Idee verfolgt [84]. Die Läsion wurde mittels eines Inpainting aus den Daten entfernt, ohne Informationen aus der anderen Hemisphäre zu nutzen. Ein weiterer Ansatz [55] betrachtet das Problem als eines mit fehlenden Daten. Alle bisher vorgestellten Methoden leiden unter Nicht-Berücksichtigung der gesamten Information. Teils werden diese komplett ausgeblendet, teils geschätzt. Ein Verfahren, das die gesamte Information berücksichtigt ist [4]. Die Idee des Ansatzes besteht darin die Registrierung mit der Segmentierung zu kombinieren, um die vorherige Generierung der Läsionsmaske als ein Vorverarbeitungsschritt zu ver-

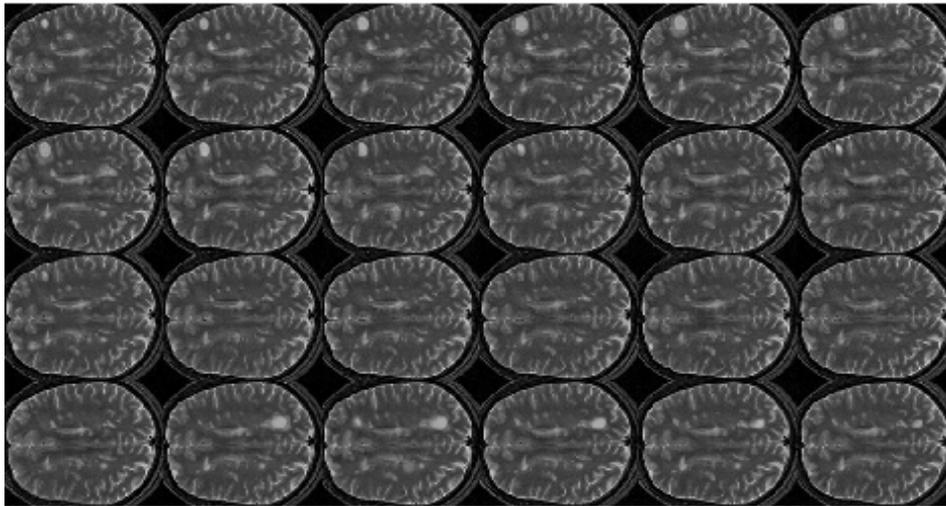


Abbildung 5.3.: Zeitreihe einer Schicht aus einer T_2 -gewichteten MRT-Aufnahme. Die Schichten sind angeordnet in zeitlicher Reihenfolge von oben links nach unten rechts. Die Aufnahmen sind über ein Jahr verteilt, aufgenommen in den Wochen 0, 1, 2, 3, 4, 5, 6, 7, 9, 11, 12, 13, 14, 16, 17, 19, 21, 23, 28, 34, 38, 42, 46 und 50. Läsionsänderungen, -neuerscheinungen und -rückgänge sind deutlich sichtbar.

meiden. Dieser Ansatz liefert, besonders bei großen Läsionen, jedoch keine genauen Ergebnisse [3].

Der neue, hier vorgestellte, Ansatz beruht ebenfalls auf Masken. Diese Masken müssen in einem vorherigen Schritt erstellt werden. Dies kann manuell, interaktiv oder automatisch erfolgen. Eine Möglichkeit der automatischen Segmentierung wollen wir in dieser Arbeit kurz beleuchten. Diese verwendet ausschließlich bisher vorgestellte Bausteine und ist darum besonders attraktiv. Durch eine vorangehende nichtlineare Registrierung und eine Analyse des Deformationsfeldes kann dabei die Segmentierung gewonnen werden. Der Ansatz hierfür orientiert sich an [88], der ebenfalls im Bereich der Multiplen Sklerose anzusiedeln ist. Stellen, an denen eine Veränderung der Läsion auftritt, weisen im Deformationsfeld eine Läsionsänderung auf. Diese nutzen wir zusammen mit Intensitätsinformationen zur Segmentierung. Durch die Glattheit der Deformationsfelder ist keine exakte Segmentierung notwendig, so dass eine grobe Segmentierung für die Anwendung ausreichend ist.

Im Folgenden wollen wir unseren Algorithmus erklären und an einem Beispiel vorstellen. Wir verfolgen dabei das Ziel, die Möglichkeiten dieses Ansatzes aufzuzeigen. Weitere Maßnahmen zur möglichen Verbesserung des Verfahrens folgen im Ausblick. Hauptziel ist eine stärkere Unterdrückung des Rauschen im Gegensatz zu bestehenden Verfahren, besonders in Regionen nahe der eigentlichen Läsionen.

5.3. Läsions-Volumenerhaltende Registrierung

Typischerweise werden Subtraktionsbilder mittels einer Bildverarbeitungskette generiert. Diese enthält Schritte wie das Entfernen des Schädels, Bias-Korrektur, Skalierung der Intensitäten und eine nachfolgende rigide Registrierung [69]. Hier schlagen wir eine Erweiterung bzw. Modifikation vor. Die Erweiterung besteht aus einem zusätzlichen nichtlinearen, restringierten Registrierungsschritt, der das Läsionsvolumen des Templatebildes unverändert lässt. Der Gewinn dadurch ist ein Subtraktionsbild, das deutlich rauschfreier ist und die Läsionsänderungen besser hervorhebt. Dazu benötigen wir eine grobe Segmentierung der Läsionen in beiden Bildern. Diese könnte durch eine vorangehende nichtlineare Registrierung, wie oben beschrieben, automatisch erzeugt werden. Die vorgeschlagene Bildverarbeitungskette besteht aus

1. einer Maskierung des Schädels,
2. einer Bias-Korrektur,
3. einer Intensitätskorrektur,

4. einer rigiden Registrierung,
5. einer nichtlinearen Registrierung zur Segmentierung der Läsionen und
6. einer abschließenden restringierten, nichtlinearen Registrierung.

Schritt 1-4 betrachten wir in dieser Arbeit nicht. Wir gehen von rigide ausgerichteten Daten aus und vernachlässigen Intensitätsunterschiede, die in den Punkten 2-3 behandelt werden. Eine Schädelsegmentierung nehmen wir ebenfalls als gegeben hin. Wir starten somit nach der rigiden Registrierung mit der Idee einer automatischen Segmentierung der Läsionen. Hauptaugenmerk bleibt jedoch der nichtlineare, restringierte Registrierungsschritt, der die Volumenerhaltung der Läsionen fordert.

Datenlage Wir betrachten im Folgenden die Berechnung des Subtraktionsbildes der Bilder \mathcal{R} und \mathcal{T} , aufgenommen zu den Zeitpunkten t_j, t_{j+1} . Ein Beispiel für diese beiden Aufnahmen und des Subtraktionsbildes nach der rigiden Registrierung können wir in Abbildung 5.4 betrachten. Dieses Beispiel wird uns noch weiter bei der Vorstellung des neuen Verfahrens verfolgen. Es handelt sich dabei um die Schichten von MRT-Aufnahmen mit jeweils einer Bildgröße $m = (140, 176, 49)$, aufgenommen mit einem zeitlichen Abstand von 30 Wochen. Dieser Datensatz ist sehr interessant, da sowohl wachsende, schrumpfende als auch neue bzw. abgebaute Läsionen vorhanden sind. Die 3D-Volumen sind bereits rigide aufeinander ausgerichtet. Die Daten wurden bereitgestellt von Charles Guttman und Dominik Meier vom Brigham and Women's Hospital und Harvard Medical School, Boston, USA. Der gesamte Datensatz besteht aus Aufnahmen zu 24 Zeitpunkten verteilt über ungefähr ein Jahr. Abbildung 5.3 zeigt eine Schicht über den gesamten Zeitverlauf.

Standardansätze Bei der Subtraktionsbildgebung für Multiple Sklerose wird meist auf eine rigide Registrierung zurückgegriffen (siehe z. B. [70]). Da sich das Gehirns größtenteils rigide verhält, erhalten wir hier bereits eine gute Lösung. Ein Schichtauszug ist in Abbildung 5.5 zu betrachten. Allerdings erhalten wir noch ein deutliches Rauschen, verursacht durch kleine, nichtlineare Deformationen. Im Folgenden werden wir dieses Ergebnis als Referenz für die visuelle Einstufung der Güte verwenden. Eine nichtlineare Registrierung kann zwar das Distanzmaß stark reduzieren, jedoch hilft das resultierende Subtraktionsbild nicht bei der Detektion von Läsionsänderungen. Läsionsänderungen, die beim Subtraktionsbild nach der rigiden Registrierung gut sichtbar waren, sind teilweise fast komplett verschwunden. Allerdings konnte das Rauschen in anderen Bereichen besser unterdrückt werden (siehe Abbildung 5.6). Dennoch, das Ergebnis ist klinisch nicht praktikabel. Verwenden wir

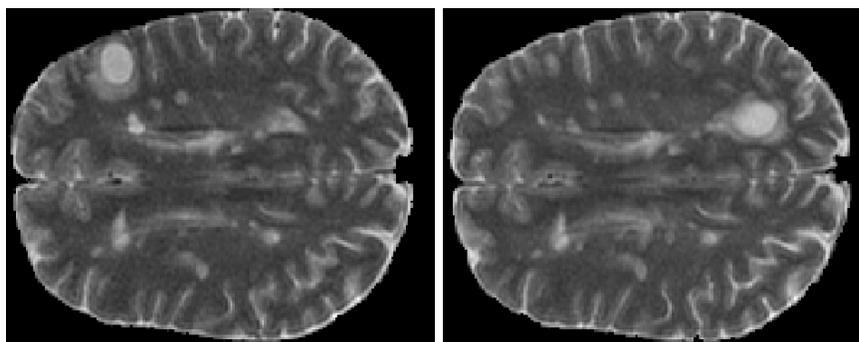


Abbildung 5.4.: Korrespondierende Schichten des Referenz- (Zeitpunkt t_j , links) und Templatebildes (Zeitpunkt t_{j+1} , rechts) aus dem Beispieldatensatz, die bereits rigide ausgerichtet sind. Es sind deutliche Läsionsänderungen zu sehen. Sowohl schrumpfende, wachsende, neue und verschwundene Läsionen können beobachtet werden.

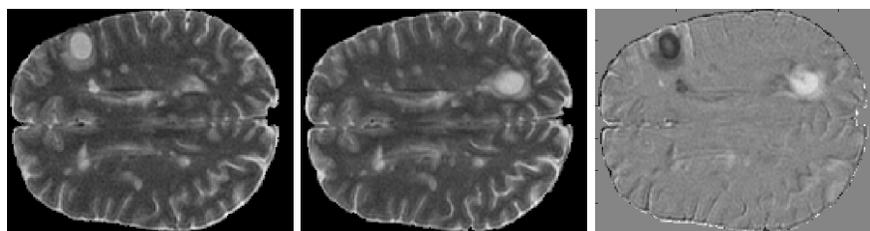


Abbildung 5.5.: Schichtauszug zweier T_2 -gewichteter MRT-Aufnahmen eines MS-Patienten mit einer zeitlichen Differenz von 30 Wochen. Oben sehen wir die Ausgangslage nach einer rigiden Registrierung. Links ist die Schicht des Referenzbildes, gefolgt vom Templatebild (Mitte) und dem Subtraktionsbild zu sehen. Die Läsionsänderungen sind in den Bildern durch helle und dunkle Flecken gegeben. Wir sehen allerdings durch unzureichende Ausrichtung ebenfalls Strukturen, an denen keine Läsionen vorhanden sind - das Rauschen.

hingegen den Ansatz, bei dem der Läsionsbereich im Distanzmaß maskiert wird, wie in [9], können wir das Rauschen unterdrücken und starke Volumenänderungen verhindern. Wie stark die Änderung unterdrückt wird, hängt dabei von der Größe der Maske ab. Dies wirkt sich wiederum auf die Rauschunterdrückung in dieser Region aus. Abbildung 5.7 zeigt die Subtraktionsbilder resultierend aus zwei maskierten, nichtlinearen Registrierungen mit unterschiedlicher Maskengröße. Die nichtlinearen

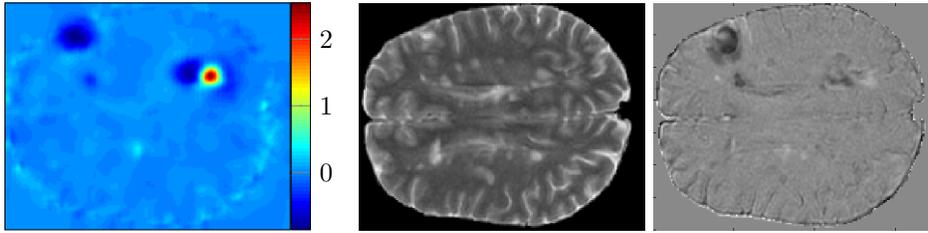


Abbildung 5.6.: Resultierendes Subtraktionsbild nach einer nichtlinearen Registrierung (rechts). Im deformierten Templatebild ist die Läsion oben rechts kaum noch zu erkennen (Mitte). Die Darstellung der Volumenänderung (links) weist an dieser Stelle starke Änderungen auf.

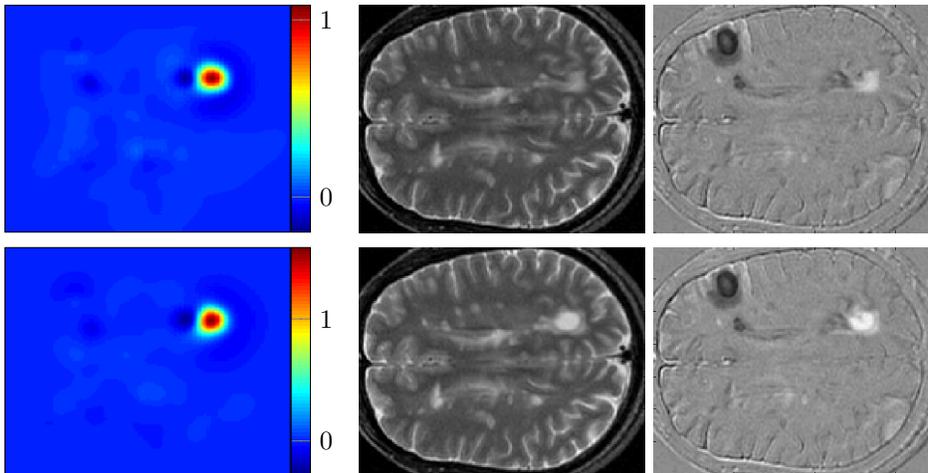


Abbildung 5.7.: Maskierter nichtlinearer Ansatz mit dem Subtraktionsbild (rechts). Die Masken wurden mit unterschiedlicher Standardabweichung geglättet, oben schwach, unten stärker. Durch eine größere Maske wird eine bessere Erhaltung der Läsion erreicht. Dies kann im deformierten Templatebild (Mitte) beobachtet werden. Links sind die Volumenänderungen dargestellt.

Ansätze wurden mit curvature Regularisierung bis zu einem Level mit der Auflösung $m = (128, 128, 49)$ berechnet. Die Betrachtung der Volumenänderung bezieht sich auf Abschnitt 3.4, Gleichung (3.4.1). Beim unmaskierten Fall traten minimale und maximale Volumenänderungen von $4.35 \cdot 10^{-4}$ bzw. 12.42 bei einer relativen Distanzmaßreduktion von $d_{\text{rel}} = 0.3037$. Der Regularisierungsparameter wurde mit $\alpha = 10^4$

gewählt. Für den maskierten Fall fiel die Wahl auf $\alpha = 10^5$. Die Masken wurden mit einem Gaußfilter (Gleichung (3.49)) mit unterschiedlicher Standardabweichung σ geglättet. Eine Veränderung des Maskenbereichs durch die Transformation wurde nicht berücksichtigt. Für kleineres σ erhalten wir noch eine Läsionsgrößenänderung im Templatebild. Dies kann daran liegen, dass die Veränderung des Maskenbereichs nicht berücksichtigt wurde. Minimale und maximale Volumenänderung, verursacht durch die Deformation, liegen bei 0.59 bzw. 2.68, bei $d_{\text{rel}} = 0.5965$. Für die größere Maske ergibt sich eine minimale und maximale Volumenänderung von 0.42 bzw. 2.61, bei $d_{\text{rel}} = 0.6333$. Die gemessenen Volumenänderungen treten selbst für große Masken auch im Bereich der Läsionen auf. Dies kann durch die Maskierung nicht verhindert werden. Um die Läsionsvolumina vollständig zu erhalten, werden wir im Folgenden eine Volumenerhaltung im Bereich der Läsionen fordern. Dabei soll auch die Deformation der Läsionsmaske berücksichtigt werden. Um den Weg zu einem komplett automatischen Verfahren zu zeigen, stellen wir eine Möglichkeit der automatischen Segmentierung aktiver Läsionen im Vorweg vor.

Automatische Segmentierung von Aktiven Läsionen Im Folgenden wollen wir die Möglichkeit einer automatischen Segmentierung und damit einer Automatisierung des gesamten Registrierungsschritts beleuchten. Dabei geht es uns vorerst mehr um eine Machbarkeitsanalyse denn eine perfekte Umsetzung.

Aufgrund der Glattheit der Deformation ist für den Algorithmus eine exakte Segmentierung nicht von Nöten. Daher beschäftigen wir uns hier mit einem einfachen automatischen Verfahren, um aktive, also sich über zwei Zeitpunkte verändernde Läsionen zu segmentieren. Aktive Läsion sind also Läsionen, die von einem zum anderen Zeitpunkt wachsen, schrumpfen (bis zur vollständigen Regeneration) oder neu entstehen. Die inaktiven Läsionen sind über beide Bilder bis auf leichte Deformationen konstant und können somit ohne weitere Probleme aufeinander abgebildet werden. Die grundsätzliche Idee für diesen Ansatz stammt bereits aus dem Jahr 1997 und wurde von Thirion und Calmon [88] vorgestellt. Betrachten wir zwei Zeitpunkte t_j, t_{j+1} . Seien zu diesen Zeitpunkten Bilder \mathcal{R} respektive \mathcal{T} gegeben. Auf diese Daten wenden wir eine nichtlineare Registrierung mit curvature Regularisierer (Gleichung (3.45)) mit SSD als Distanzmaß (Gleichung (3.5)) an, d. h.

$$\varphi^* = \min_{\varphi} \frac{1}{2} \int_{\Omega} (\mathcal{T}(\varphi(\mathbf{x})) - \mathcal{R}(\mathbf{x}))^2 \, d\mathbf{x} + \frac{\alpha}{2} \int_{\Omega} \sum_{i=1}^2 \left(\Delta \left(\varphi_i(\mathbf{x}) - \varphi_i^{(0)}(\mathbf{x}) \right) \right)^2 \, dx. \quad (5.1)$$

Dabei ist der Regularisierungsparameter α so zu wählen, dass sicher ein faltungsfreies Deformationsfeld erzielt wird. Die Referenztransformation φ_0 ist das Ergebnis der rigiden Registrierung. Es wurde SSD als Distanzmaß gewählt (siehe Abschnitt 3.5), da wir in einem monomodalen Fall sind und im Gegensatz zum LSD

keine Intensitäten geändert werden. Um die spätere Detektion der Läsionen als zusammenhängendes Gebiet zu erleichtern verwenden wir die curvature Regularisierung (vgl. Abschnitt 3.6.1), damit wir glatte Ergebnisse in der Volumenänderung erhalten. Von glatten Ergebnissen erhoffen wir uns eine sauberere Segmentierung durch besseren Zusammenhang der Gebiete.

Betrachten wir zeitlich versetzte Aufnahmen eines Gehirns ohne Pathologie, verhält sich dieses nahezu rigide. Volumenänderungen treten somit kaum auf bzw. sind sehr klein. Es gilt nahezu überall für die Funktionaldeterminante $\det(D\varphi^*) \approx 1$. Treten jedoch aktive Läsionen auf, wird es regional zu Volumenänderung kommen. Diese spiegeln sich in der Funktionaldeterminante wider, vergleiche Gleichung (3.26). Folglich gibt uns eine Inspektion der Transformation bzw. deren Funktionaldeterminante Aufschluss über Regionen, in denen aktive Läsionen anzusiedeln sind. Es sei angemerkt, dass dies bei MS vor allem für jüngere Personen bei kürzeren Abständen eintritt. Im höheren Alter ist eine deutliche Volumenänderung auch Hirnathropie zu erwarten (siehe z. B. [50]). Auf diese Fälle wollen wir im Folgenden nicht weiter eingehen. Zusätzlich weisen Läsionen in T_2 -gewichteten MR-Aufnahmen hohe Intensitäten auf. Kombinieren wir diese beiden Informationen, können wir gute und robuste Ergebnisse für eine Segmentierung der aktiven Läsionen erhalten. Wir unterscheiden dabei in Läsionen, die im Referenzbild aktiv sind und Läsionen, die im Templatebild aktiv sind und bezeichnen diese Mengen der dazugehörigen Punkte mit $\Lambda_{\mathcal{R}}$ und $\Lambda_{\mathcal{T}}$. Eine Läsion ist in nur einem Bild aktiv, wenn sie im Bild zum anderen Zeitpunkt gar nicht vorhanden ist. Folglich definieren wir die Menge der aktiven Läsionen im Referenzbild $\Lambda_{\mathcal{R}} \subset \Omega_T$ mit

$$\Lambda_{\mathcal{R}} := \{\mathbf{x} : f(\det(D\varphi^*)) > \tau_F \wedge \mathcal{R}(\mathbf{x}) > \tau_I\} \quad (5.2)$$

und die Menge der aktiven Läsionen im Templatebild $\Lambda_{\mathcal{T}} \subset \Omega_T$ mit

$$\Lambda_{\mathcal{T}} := \{\mathbf{x} : f(\det(D(\varphi^*)^{-1})) > \tau_F \wedge \mathcal{T}(\mathbf{x}) > \tau_I\} \quad (5.3)$$

Dabei stellen τ_F, τ_I die Schwellwerte für die Funktionaldeterminante bzw. Intensitäten. Die Funktion f misst die Abweichung von $\det(D\varphi) = 1$. Unsere Wahl von f fällt auf eine weitere Modifikation der symmetrische Varianten (4.7) und (4.8) des vorangegangenen Kapitels, d. h.

$$f(x) = \frac{x - 1}{x}. \quad (5.4)$$

Diese hat den Vorteil, dass sie weiterhin symmetrisch ist, jedoch stärker die Unterschiede nahe der Eins erkennt. Abbildung 5.8 visualisiert das Verhalten der drei uns bekannten Funktionen um den Punkt $x = 1$.

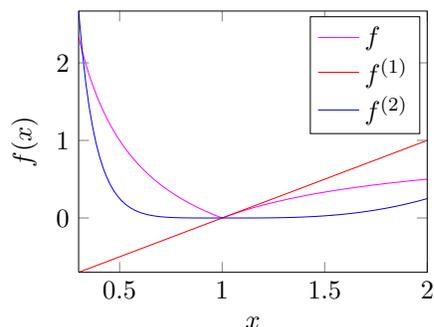


Abbildung 5.8.: Unterschiede der Funktionen für die Betrachtung Volumenänderung zur Segmentierung von aktiven Läsionen. Die gewählte Funktion f kombiniert die Vorteile von $f^{(1)}$ und $f^{(2)}$. Sie weist die Symmetrie von $f^{(2)}$ auf, bietet jedoch eine ähnlich gute Differenzierungsmöglichkeit um den Punkt $x = 1$.

Die Diskretisierung erfolgt nach den bekannten Mitteln. Für die Funktionaldeterminante verwenden wir wie bei der volumenerhaltenden Registrierung keine Unterteilung in Dreiecke/Tetraeder. Für diese Anwendung reicht dies mit einem sicher gewählten Regularisierungsparameter α aus, der Faltungen vermeidet.

Zur Demonstration der Idee betrachten wir das in Abschnitt 5.3 vorgestellte Beispiel. Dieses ist dargestellt in Abbildung 5.4. Hier treten sowohl inaktive als auch aktive Läsionen auf. Die Berechnungen erfolgten auf den Levels $m^{(1)} = (18, 22, 7)$, $m^{(2)} = (35, 44, 13)$, $m^{(3)} = (70, 88, 25)$ und $m^{(4)} = (140, 176, 49)$ mit Regularisierungsparameter $\alpha = 10^5$. Abbildung 5.9 zeigt die Ergebnisse der Segmentierung bei Nutzung der Funktion (5.4) zur symmetrischen Darstellung der Volumenänderung mit Schwellwert $\tau_F = 0.18$. Als Intensitätsschwellwert wurde $\tau_I = 500$ gewählt. Der Einfluss unterschiedlicher Funktionen zur Darstellung der Volumenänderung ist in Abbildung 5.10 gegeben. Die symmetrische Funktion (4.7) hebt zwar die Hauptänderungen deutlich hervor, jedoch ist die Separierbarkeit bei kleineren Änderungen gering. Dagegen weist die unsymmetrische Variante (4.7) eine schwierige Wahl des Schwellwerts auf. Die genutzte Funktion (5.4) ermöglicht eine gute Abgrenzung zwischen den Volumenänderungen, verursacht durch Läsionsänderungen und anderen Möglichkeiten. Es zeigt sich, dass eine automatische Erkennung von aktiven Läsionen mit den bekannten Methoden durchführbar ist. Jedoch müssen wir noch an der Robustheit des Verfahrens arbeiten, damit es ohne Probleme in der Bildverarbeitungskette zur Berechnung des Subtraktionsbildes eingesetzt werden kann. Im nächsten Abschnitt wollen wir uns nun dem zentralen Thema des Kapitels zuwenden - der nichtlinearen, restringierten Registrierung von MR-Aufnahmen

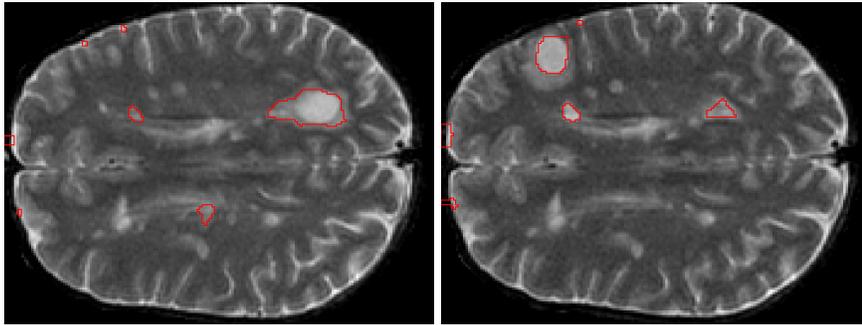


Abbildung 5.9.: Beispiel der automatischen Segmentierung von aktiven Läsionen mittels Deformationsanalyse und Intensitätsinformationen. Wir sehen ein zweidimensionales Beispiel mit wachsenden und schrumpfenden Läsionen von Zeitpunkt t_j nach t_{j+1} .

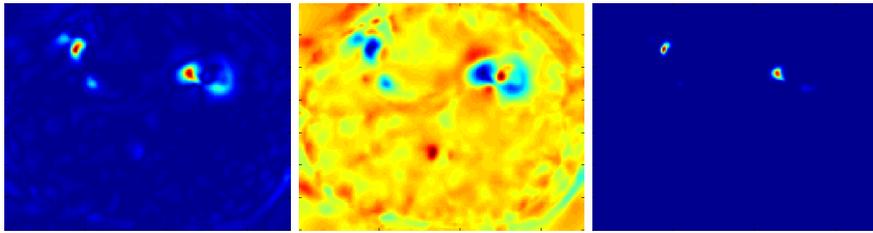


Abbildung 5.10.: Die Abbildung zeigt den Einfluss verschiedener Funktionen zur Betrachtung der Volumenänderung. Links ist die Verwendung der Funktion (5.4), in der Mitte der einfachen unsymmetrischen Variante (4.6) und rechts der symmetrischen Funktion (4.7) zu sehen. Während wir links eine gute Differenzierbarkeit auch bei kleinen Änderungen erkennen können, ist dies bei den anderen beiden Varianten nicht der Fall. Während in der Mitte das Rauschen zu stark ist, erkennen wir rechts nur große Volumenänderungen deutlich. Vergleiche hierzu auch Abbildung 5.8.

zur Subtraktionsbildgebung.

Läsions-Volumenerhaltende Registrierung Die Hauptaufgabe der Subtraktionsbildgebung ist es, Erkrankungsaktivität beobachtbar zu machen. Diese drückt sich bei MS unter anderem durch die (Volumen-)Änderung der Läsionen aus. Ein optimales Verfahren würde z. B. die graue und weiße Substanz genau aufeinander abbilden, um einzig Änderungen der Läsionen sichtbar zu machen. Eine rigide Registrierung

alleine kann nicht Sorge tragen für den ersten Aspekt. Deswegen nutzen wir ein nichtlineares Transformationsmodell. Für den zweiten Aspekt führen wir volumenerhaltende Nebenbedingungen in der Region der aktiven Läsionen des Templatebildes und des Referenzbildes ein. Durch den genutzten Euler Ansatz müssen wir - wie bei der Längenerhaltung (vgl. Abschnitt 4.7) - die Volumenbedingungen an die Inverse der Transformation für Λ_T des Templatebildes stellen. Die Nebenbedingungen sollen nämlich für Punkte gelten, die durch die Transformation in die Region der Läsion abgebildet werden, d. h. $\varphi(\mathbf{x}) \in \Lambda_T$ (Gleichung (5.3)) oder äquivalent $\mathbf{x} \in \varphi^{-1}(\Lambda_T)$. Wie bei der längenerhaltenden Registrierung benötigen wir die punktweise Inversitätsforderung 4.7. Diesmal jedoch auf einer Menge der gleichen Dimension. Für die Menge Λ_R (Gleichung (5.2)) der Läsionen im Referenzbild muss die Maske nicht transformiert werden. Da wir eine große globale Glattheit bevorzugen wollen, fällt die Wahl des Regularisierers \mathcal{S} auf den curvature-Term ((3.45)). Die Wahl des Distanzmaßes $\mathcal{D}(\mathcal{R}, \mathcal{T}; \varphi)$ werden wir in unseren Versuchen untersuchen. Dabei werden wir das SSD (3.33), NGF (3.38) und LSD (3.35) testen.

Es ergibt sich das Registrierungsproblem

$$\min_{\varphi} \mathcal{D}(\mathcal{R}, \mathcal{T}; \varphi) + \frac{\alpha}{2} \int_{\Omega} \sum_{i=1}^2 \left(\Delta \left(\varphi_i(\mathbf{x}) - \varphi_i^{(0)}(\mathbf{x}) \right) \right)^2 dx$$

u.d.NB $c(\mathbf{x}) = f(\det(D\varphi(\mathbf{x}))) = 0, \quad \mathbf{x} \in \varphi^{-1}(\Lambda_T) \cup \Lambda_R.$

Im Gegensatz zur längenerhaltenden Registrierung aus dem vorangegangenen Kapitel, führen wir keine zusätzliche Regularisierung für die inverse Transformation ψ ein. Dadurch, dass die inverse Transformation auf einem Gebiet derselben Dimension lebt, gehen wir von einer ausreichenden globalen Regularisierung aus und damit von einer implizierten Regularisierung von ψ durch die Regularisierung der Transformation φ .

Diskretisierung Die Diskretisierung des Registrierungsfunctionals samt Nebenbedingungen verläuft analog zum Kapitel der prä/post-Registrierung (Kapitel 4). Aufgrund der Volumenbedingungen arbeiten wir auf einem nodalen Gitter. Als Straffunktion f wählen wir die unsymmetrische Variante gegeben in Gleichung (4.6).

Optimierung Zur Optimierung verwenden wir wiederum einen Multilevel Augmented Lagrangian Ansatz wie in Abschnitt 4.6.6 beschrieben. Wir verwenden für die unrestringierte Optimierung das Gauss-Newton bzw. L-BFGS-Verfahren. Im Gegensatz zur längenerhaltenden Registrierung verändert sich bei uns allerdings die

Quelltext 5.1: Modell-Algorithmus zur Berechnung der inversen Deformation

```

1 % Deformation  $\varphi = \mathbf{x} + \mathbf{u}$ 
2 % Inverse Deformation  $\psi = \mathbf{x} + \mathbf{v}$ 
3 function  $\mathbf{v}^{\text{ende}} = \text{inverseDeformation}(\mathbf{u})$ 
4      $\mathbf{v} = 0$ 
5     for  $k=1:\text{maxIter}$ 
6          $\mathbf{v}(\mathbf{x}) = -\mathbf{u}(\mathbf{x} + \mathbf{v}(\mathbf{x}))$ 
7     end
8 end

```

Menge der Punkte, an denen die punktweise Inverse berechnet werden soll. Wir benötigen eine Prolongation dieser Punkte. Wir umgehen hier allerdings eine lineare Interpolation der Inversen auf einem unregelmäßigen Gitter und berechnen stattdessen beim Übergang vom groben auf das feine Level die komplette Inverse auf dem groben Gitter, prolongieren die resultierende inverse Deformation und greifen die entsprechenden Punkte im Gebiet unserer Läsionen ab. Zur Berechnung der inversen Deformation nutzen wir eine Fixpunktiteration, vorgestellt in [16]. Dieses Verfahren ist sehr leicht zu implementieren und ihm wird eine geringe Laufzeit nachgesagt. Der Algorithmus ist in Quelltext 5.1 angegeben. In der einführenden Arbeit wird als hinreichende Konvergenzeigenschaft Lipschitz-Stetigkeit der Verrückung \mathbf{u} vorausgesetzt, d. h. $\|\mathbf{u}(\mathbf{x}) - \mathbf{u}(\mathbf{y})\| \leq c\|\mathbf{x} - \mathbf{y}\|$, $c < 1$. Auch wenn diese nicht immer erfüllt ist, haben wir bisher keine Probleme des Verfahrens in der Praxis entdecken können.

5.4. Anwendung und Vergleich von Distanzmaßen

Zur Demonstration der Möglichkeiten mit dem regional volumenerhaltenden Ansatz befassen wir uns mit der Registrierung zweier extremer Zeitpunkte, bei denen starke Volumenänderungen vorhanden sind. Dafür nutzen wir die vorgestellten Zeitpunkte aus dem Abschnitt 5.3 der Datenlage, aufgenommen im Abstand von 30 Wochen. Aufgrund der größeren Robustheit des Gauss-Newton-Verfahrens, testen wir hiermit das Verfahren für unterschiedliche Distanzmaße. Dazu betrachten wir das SSD, LSD und NGF. Für SSD und NGF führen wir eine Aufdatierung der Schätzung des Lagrange-Vektors $\boldsymbol{\lambda}$ sowie des Strafparameters μ nach jedem unrestringierten Optimierungsschritt durch. Beim LSD zeigt es sich, dass diese Strategie nicht stabil genug ist. Daher verwenden wir zwischen den Aufdatierungen eine komplette unrestringierte Optimierung mit einer maximalen Anzahl von 20 Iterationsschritten. Die Registrierung lief auf den Leveln $m^{(1)} = (16, 16, 16)$, $m^{(2)} = (32, 32, 32)$ und

$m^{(3)} = (64, 64, 49)$. Für SSD haben wir $\mu^{(0)} = 10^2, 10^3, 10^4$ auf den drei Leveln sowie $\alpha = 5 \cdot 10^5$ gewählt. Beim Einsatz von LSD war $\mu^{(0)} = 1, 10, 10^2$ bei $\alpha = 10^{-2}$ und $\mu^{(0)} = 10, 10^2, 10^3$ mit $\alpha = 5 \cdot 10^{-2}$ bei der Anwendung von NGF. Bei der Verwendung von LSD wurden die Grauwerte auf das Intervall $[0, 1]$ normalisiert. Als Start für den Lagrange-Vektor fiel die Wahl auf $\lambda^{(0)} = \mathbf{0}$. Die Ergebnisse sind in Abbildung 5.11 zu betrachten. Es zeigt sich, dass SSD zu den besten Ergebnissen führt und dabei auch die geringste Laufzeit aufweist. Die wenig überzeugenden Ergebnisse bei der Nutzung von NGF könnten auch an der Schwierigkeit der Wahl des Rauschparameters ϵ liegen, der hier nicht optimiert wurde. Besonders die Ergebnisse des LSD enttäuschen, der für MRT-Aufnahmen im ersten Moment eine gute Wahl sein sollte. Globale Intensitätsunterschiede, die bei MRT-Aufnahmen auftreten, sollten korrigiert bzw. ignoriert werden. Durch die Änderungen des Bildes, verursacht durch die Hyperintensitäten der Läsionen, kann es bei der Optimierung der optimalen Grauwertabbildung allerdings zu Abbildungen zwischen nicht korrespondierenden Grauwerten kommen. Die relativen Distanzmaßreduktionen sind $d_{\text{rel}} = 0.6114$ für SSD, $d_{\text{rel}} = 0.9013$ für LSD und $d_{\text{rel}} = 0.9245$ für NGF. Das wir nicht nur die relative Distanzmaßreduktion betrachten dürfen, wird beim Ergebnis von LSD deutlich. Der optische Eindruck zeigt eine Verschlechterung des Ergebnisses. Mit den oben aufgeführten Gründen, scheint das LSD eindeutig das falsche Modell für diese Anwendung zu sein.

Wir können feststellen, dass SSD von den drei angewendeten Distanzmaßen die besten Ergebnisse sowohl rein optisch als auch bezüglich der relativen Distanzmaßreduktion liefert. Zudem ist SSD von der Rechenzeit das kostengünstigste Distanzmaß. Im Folgenden wollen wir daher die unrestringierten Optimierungsverfahren innerhalb des Augmented Lagrangian Ansatzes bei Verwendung von SSD testen.

5.5. Vergleich der Optimierungsverfahren

Da SSD das kostengünstigste Distanzmaß ist, gehen wir in diesem Abschnitt beim Multilevel Ansatz auf das feinste Level für die betrachteten Optimierungsverfahren. Die resultierenden Level sind nun $m^{(1)} = (16, 16, 16)$, $m^{(2)} = (32, 32, 32)$, $m^{(3)} = (64, 64, 49)$ und $m^4 = (179, 140, 49)$.

Bei Verwendung des Gauß-Newton-Verfahrens setzen wir wieder auf die Strategie, den Lagrange-Vektor nach jedem Optimierungsschritt aufzudatieren. Die maximale Anzahl an Iterationsschritten für die einzelnen Level war 200, 100, 50 bzw. 25. Als Start-Lagrange-Vektoren haben wir $\lambda^{(0)} = \mathbf{0}$ gewählt. Die Strafparameter über die Level waren gegeben durch $\mu^{(0)} = 10^2, 10^3, 10^4, 10^6$. Als Regularisierungsparameter

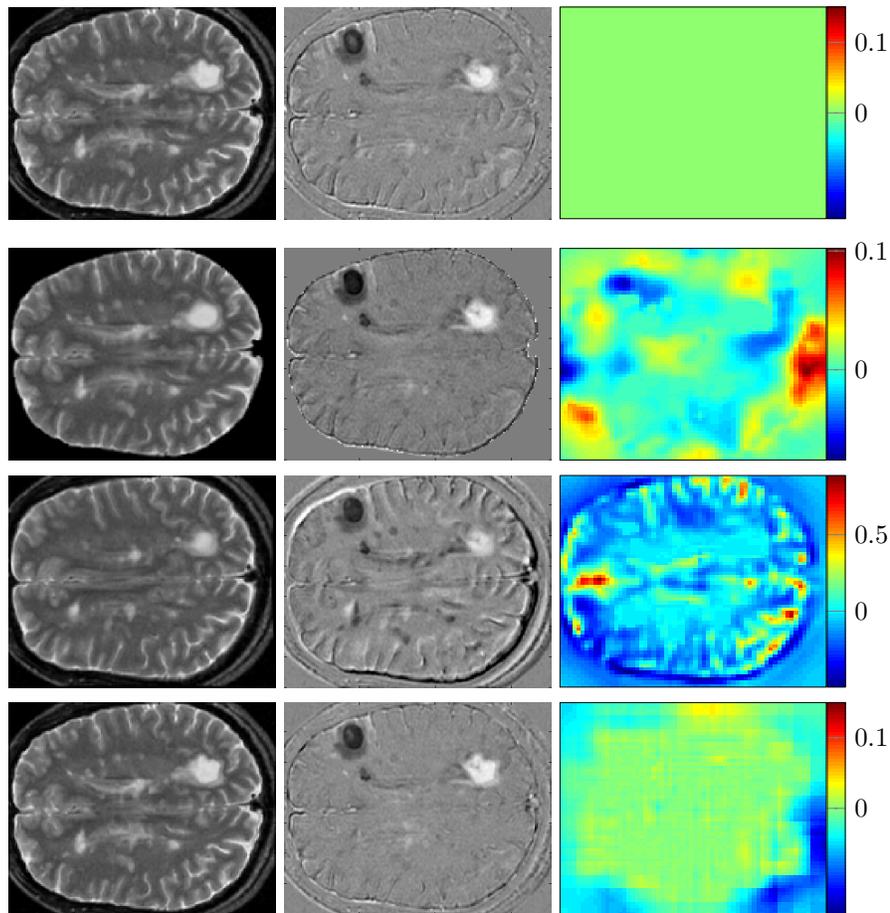


Abbildung 5.11.: Wir sehen einen Vergleich verschiedener Distanzmaße für die regional volumenerhaltende Registrierung. Von links nach rechts sind Volumenänderung, deformiertes Template und erzielt Subtraktionsbild zu betrachten. Oben ist als Referenz das Ergebnis bei einer rigiden Registrierung zu betrachten, bei der noch starkes Rauschen sichtbar ist. In den folgenden Zeilen sind von oben nach unten die Ergebnisse bei Verwendung des SSD, LSD und NGF zu sehen. Sowohl NGF als auch SSD erzielen gute Ergebnisse, wobei das SSD leicht überlegen ist. Das LSD liefert keine brauchbaren Ergebnisse.

fiel die Wahl auf $\alpha = 5 \cdot 10^5$. Der Algorithmus erreichte eine relative Verbesserung

der Distanz von $d_{\text{rel}} = 0.52$.

Im Vergleich dazu erreichte das L-BFGS-Verfahren $d_{\text{rel}} = 0.444$ bei einer äquivalenten Laufzeit zum Gauss-Newton-Verfahren. Es stellt sich heraus, dass im Gegensatz zur längenerhaltenden Registrierung das Verfahren an Stabilität deutlich besser performiert. Bei der Wahl des Regularisierungsparameters zeigte sich das Verfahren robuster als beim Gauss-Newton-Verfahren. Die Wahl fiel auf $\alpha = 10^3$. Die innere Iterationsschleife des L-BFGS-Verfahrens hatte dabei eine maximale Anzahl von 30 Iterationsschritten zwischen jeder Aufdatierung des Lagrange-Vektors und Strafparameters. Für die äußere Schleife wurde eine maximale Anzahl von 50, 40, 30 bzw. 20 Schritten festgelegt, abhängig vom betrachteten Level. Als Strafparameter wurde $\mu^{(0)} = 10^1, 10^2, 10^2$ und 10^3 für die entsprechenden Level und als Start-Lagrange-Vektor $\lambda^{(0)} = \mathbf{0}$ gewählt.

Die Ergebnisse der beiden Verfahren sind in Abbildung 5.12 abgebildet. Rein visuell lassen sich diese kaum unterscheiden, wobei das L-BFGS-Verfahren leicht überlegen ist. Die äquivalente Laufzeit und Verbesserung des Distanzmaßes sprechen allerdings zusammen, nach diesem ersten Test, deutlich für eine Anwendung des L-BFGS-Verfahrens innerhalb des Augmented Lagrangian Ansatzes.

5.6. Ausblick

Wir haben einen Registrierungsansatz zur Subtraktionsbildgebung bei MS-Patienten kennengelernt. Im Gegensatz zu Standard-Verfahren nutzt dieser ein nicht-lineares Deformationsmodell, ohne dabei Läsionsveränderungen komplett zu unterdrücken. Alleine die Volumenänderung wird verhindert. Dies wurde ermöglicht durch eine regionale Volumenerhaltung der Läsionen. Im vorangegangenen Abschnitt haben wir beobachten können, dass der vorgestellte Ansatz bei einem ersten Beispiel vielversprechende Ergebnisse liefert. Das sogenannte Rauschen, verursacht durch unzureichende Ausrichtung der grauen und weißen Substanz, konnte deutlich reduziert werden. Gleichzeitig bleiben die Läsionsvolumina erhalten. Die Läsionsänderungen werden nun noch deutlicher hervorgehoben.

Im Folgenden muss das Verfahren auf weiteren Daten getestet werden. Zudem kann die Implementierung effizienter gestaltet werden. Dieser Punkt wurde im Zuge dieser Arbeit nicht weiter beleuchtet. Hierunter fällt auch das Feintuning der einzelnen Parameter und eine Prolongation der Lagrange-Vektoren auf einem unregelmäßigen Gitter. Vorstellbar ist auch eine Erweiterung des Registrierungsansatzes auf die Registrierung ganzer Zeitreihen. Dies ermöglicht eventuell ein besseres Verständnis der Evolution von MS-Läsionen.

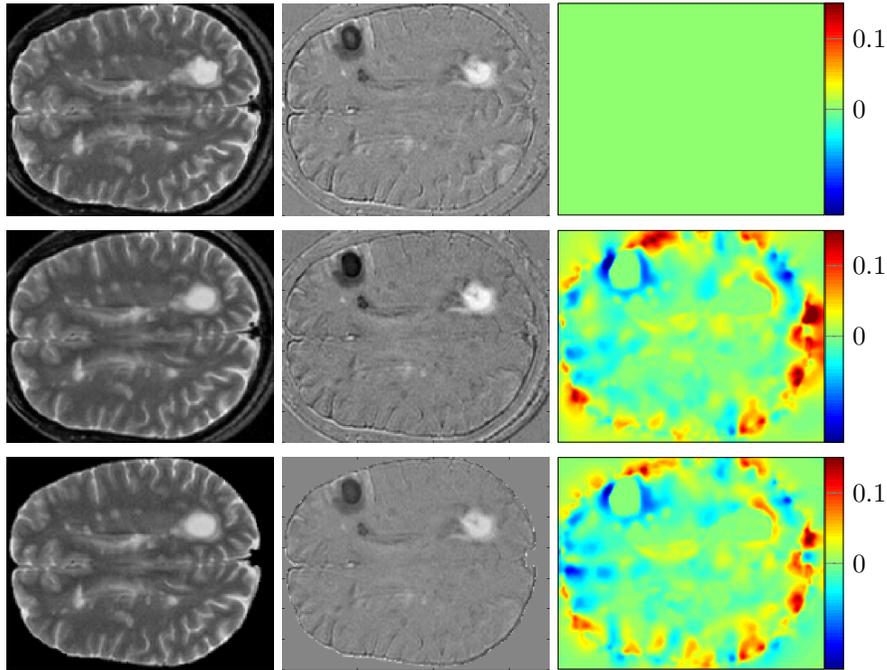


Abbildung 5.12.: Vergleich der unrestringierten Optimierungsverfahren innerhalb des Augmented Lagrangian Ansatzes für den vorgestellten Registrierungsalgorithmus zur Subtraktionsbildgebung bei MS-Patienten. Von links nach rechts sind die Volumenveränderungen, das deformierte Templatebild und das resultierende Subtraktionsbild zu den Daten aus Abbildung 5.4 visualisiert. Oben sehen wir die Ausgangslage nach einer starren Registrierung. In der Mitte sehen wir das Ergebnis bei Verwendung des Gauß-Newton-Verfahrens bzw. unten bei Verwendung des L-BFGS-Verfahrens. Beim L-BFGS-Verfahren werden sowohl die Läsionen sehr gut erhalten als auch das Rauschen stark unterdrückt. Das Gauss-Newton-Verfahren erreicht ebenfalls ein sehr gutes Ergebnis, allerdings ist noch mehr Rauschen im Bild zu betrachten.

6. Diskussion und Ausblick

Nach den Ausblicken in den einzelnen Anwendungskapiteln wollen wir abschließend noch einige allgemeine Sachverhalte der Dissertation ansprechen. Wir haben im Zuge dieser Arbeit gesehen, dass das Einbeziehen von Zusatzinformationen in die Modellierung des Registrierungsproblems teilweise unabdingbar ist. Die hier vorgestellten Probleme bezogen sich dabei auf zeitlich versetzte Aufnahmen desselben Patienten, bei dem sich die Bilder aufgrund eines chirurgischen Eingriffs bzw. eines Krankheitsverlaufes deutlich geändert haben. Aber auch gerade bei der Registrierung von Daten unterschiedlicher Patienten mit variierender Anatomie und pathologischen Fällen sind in den Daten nicht für alle Punkte in dem einen Bild auch korrespondierende Punkte in dem anderen Bild vorhanden. Durch das Einbringen geeigneter Informationen konnten wir für unsere Beispielfälle sehr gute erste Resultate erzeugen, die Standardansätzen deutlich überlegen sind. Erstes Anwendungsfeld hierfür war die Registrierung prä- und postoperativer Leberdaten. Bei nichtlinearen Standardansätzen kommt es hierbei zu einem Nachwachsen der Leber. Neben den bekannten Nebenbedingungen für eine Volumenerhaltung [41, 79] haben wir dafür neue Nebenbedingungen formuliert, die eine Längenerhaltung von Gefäßen fordern (Abschnitt 4.7). Im Zuge dieser Nebenbedingungen wurde zudem eine punktweise Inversitätsforderung vorgestellt (Abschnitt 4.7.4). Diese erweist sich als vielseitig einsetzbar, z. B. bei regional gestellten Nebenbedingungen. So konnten wir die punktweisen Inversitätsforderungen für eine Volumenerhaltung von Läsionen bei zeitlich versetzten MR-Aufnahmen von MS-Patienten nutzen.

Aus diesen Beispielen werden Vorteile der Nutzung des diskretisieren-dann-optimieren Ansatzes in der Registrierung deutlich. Zum einen können wir einen Grundbaustein, bestehend aus Distanzmaß und Regularisierer, leicht erweitern um bekannte und neue Nebenbedingungen. Zum anderen existieren bereits eine Menge an effizienten Methoden aus der numerischen Optimierung zur Lösung des diskretisierten Problems. In dieser Arbeit haben wir für diesen Schritt den Augmented Lagrangian (Abschnitt 2.2.2 und 4.6.6) verwendet. Es stellt sich dabei heraus, dass die Performance deutlich von Problemstellung und verwendetem unrestringierten Optimierungsverfahren innerhalb des Ansatzes abhängt. An diesem Punkt steckt zudem noch viel Potential für die Effizienz der Algorithmen. Wir haben gesehen, dass die Startwerte und Aufdatierungsstrategien eine wichtige Rolle spielen. Diese

können noch weiter optimiert werden. Aber auch andere Verfahren zur Optimierung von restringierten Problemen mit Gleichheitsnebenbedingungen wie z. B. das Sequential Quadratic Programming (SQP, [72]) sind interessante Alternativen.

Ein weiterer Schritt ist das Einbringen von Ungleichheitsnebenbedingungen in das Modell. So können z. B. Toleranzen bei der Volumenerhaltung der Leber angegeben werden. Dadurch können Variationen des Blutflusses und damit Volumenänderungen der Gefäße oder Schwellungen am Rande der Leber eingefangen werden. Dabei bleibt der Schritt der Diskretisierung gleich, jedoch müssen andere Optimierungsmethoden genutzt werden, die die Toleranzbereiche mit einbeziehen.

Für den Einsatz im klinischen Bereich ist zudem eine Optimierung der Laufzeit interessant. Bisher wurden die Verfahren als “Proof-of-Concept“ in MATLAB umgesetzt ohne das Ziel einer besonders performanten Implementierung. Neben den genutzten effizienten Optimierungsalgorithmen kann hier über eine matrixfreie Implementierung der (approximierten) zweiten Ableitung und eine Prädiktionierung der auftretenden linearen Gleichungssysteme nachgedacht werden. Zudem würde eine Implementierung in einer hardware-näheren Programmiersprache unter Verwendung der GPU eine deutliche Laufzeitverbesserung mit sich bringen. Daher sehen wir auch keine Schranken dafür, dass die vorgestellten Verfahren in den medizinischen Alltag einfließen können.

Im Ganzen bleibt abschließend zu sagen, dass moderne Registrierungsalgorithmen typischerweise eine feinere Modellierung aufweisen müssen, um das Problem angemessen angehen zu können. In dieser Arbeit haben wir zwei dieser Fälle gezeigt, doch das Problem von fehlenden oder nicht eindeutigen Korrespondenzen tritt in vielen weiteren Fällen auf. Gerade bei der Verfolgung von Krankheitsverläufen ist dies der Fall.

Literaturverzeichnis

- [1] *BrainWeb: Simulated brain database*, Oktober 2011, <http://mouldy.bic.mni.mcgill.ca/brainweb/>.
- [2] E. H. Adelson, C. H. Anderson, J. R. Bergen und P. J. Burt, *Pyramid methods in image processing*, RCA engineer **29** (1984), Nr. 6, 33–41.
- [3] S. M. Andersen, S. Z. Rapcsak und P. M. Beeson, *Cost function masking during normalization of brains with focal lesions: Still a necessity?*, Neuroimage **53** (2010), Nr. 1, 78–84.
- [4] J. Ashburner und K. J. Friston, *Unified segmentation*, Neuroimage **26** (2005), Nr. 3, 839–851.
- [5] D. P. Bertsekas, *Constrained optimization and Lagrange multiplier methods*, Athena Scientific, 1996.
- [6] P. J. Besl und H. D. McKay, *A method for registration of 3-D shapes*, IEEE Transactions on Pattern Analysis and Machine Intelligence **14** (1992), Nr. 2, 239–256.
- [7] S. Bock, C. Kühnel, T. Boskamp und H. O. Peitgen, *Robust vessel segmentation*, Proc. SPIE, Medical Imaging 2008: Computer-Aided Diagnosis, Bd. 6915, 2008, S. 691539–691539–9.
- [8] F. L. Bookstein, *Principal warps: thin-plate splines and the decomposition of deformations*, IEEE Transactions on Pattern Analysis and Machine Intelligence **11** (1989), Nr. 6, 567–585.
- [9] M. Brett, A. P. Leff, C. Rorden und J. Ashburner, *Spatial normalization of brain images with focal lesions using cost function masking*, Neuroimage **14** (2001), Nr. 2, 486–500.
- [10] C. Broit, *Optimal registration of deformed images*, Dissertation, University of Pennsylvania, Philadelphia, 1981.
- [11] T. M. Buzug, *Einführung in die Computertomographie*, Mathematisch-physikalische Grundlagen der Bildrekonstruktion, Springer, 2005.

- [12] N. D. Cahill und J. A. Noble, *Extending the quadratic taxonomy of regularizers for nonparametric registration*, Proc. SPIE, Medical Imaging 2010: Image Processing, Bd. 7623, 2010, S. 76230B–76230B–12.
- [13] P. Campadelli, E. Casiraghi und A. Esposito, *Liver segmentation from computed tomography scans: A survey and a new algorithm*, Artificial Intelligence in Medicine **45** (2009), Nr. 2-3, 185–196.
- [14] D. Castaing, D. Azoulay und R. Adam, *Leberchirurgie und Chirurgie der portalen Hypertonie*, Elsevier, Urban & Fischer Verlag, August 2007.
- [15] A. Charnoz, V. Agnus, G. Malandain, C. Forest, M. Tajine und L. Soler, *Tree matching applied to vascular system*, GbRPR (Berlin, Heidelberg), Bd. 3750, Springer, 2005, S. 183–192.
- [16] M. Chen, W. Lu, Q. Chen, K. J. Ruchala und G. H. Olivera, *A simple fixed-point approach to invert a deformation field*, Medical Physics **35** (2008), Nr. 1, 81–88.
- [17] N. Chitphakdithai und J. S. Duncan, *Non-rigid registration with missing correspondences in preoperative and postresection brain images*, Medical Image Computing and Computer-Assisted Intervention - MICCAI, Bd. 6361/2010, Springer, 2010, S. 367–374.
- [18] G. E. Christensen und H. J. Johnson, *Consistent image registration*, IEEE Transactions on Medical Imaging **20** (2001), Nr. 7, 568–582.
- [19] Chris A. Cocosco, Vasken Kollokian, K. S. Kwan Remi und Alan Evans, *Brainweb: Online interface to a 3D MRI simulated brain database*, Neuroimage **5** (1997), Nr. 4, 425.
- [20] A. Collignon, F. Maes und D. Delaere, *Automated multi-modality image registration based on information theory*, Information Processing in Medical Imaging (1995), 263–274.
- [21] D. L. Collins, A. P. Zijdenbos, V. Kollokian, J. G. Sled, N. J. Kabani, C. J. Holmes und A. C. Evans, *Design and construction of a realistic digital brain phantom*, IEEE Transactions on Medical Imaging **17** (1998), Nr. 3, 463–468.
- [22] C. Couinaud, *Le foie: Etudes anatomiques et chirurgicales*, Masson, Paris, 1957.
- [23] W. R. Crum, O. Camara und D. J. Hawkes, *Methods for inverting dense displacement fields: Evaluation in brain image registration*, Medical Image Computing and Computer-Assisted Intervention, Bd. 4791, August 2007, S. 900–907.
- [24] A. Derksen, *Bildregistrierung mit Volumenbeschränkung*, Bachelorarbeit, Universität zu Lübeck, 2010.

-
- [25] Y. Duan, P. G. Hildenbrand, M. P. Sampat, D. F. Tate, I Csapo, B. Moraal, R. Bakshi, F. Barkhof, D. S. Meier und C. R. G. Guttmann, *Segmentation of subtraction images for the measurement of lesion change in multiple sclerosis*, American Journal of Neuroradiology **29** (2008), Nr. 2, 340–346.
- [26] P. Dumpuri, L. W. Clements und R. Li, *Comparison of pre/post-operative CT image volumes to preoperative digitization of partial hepatectomies: a feasibility study in surgical validation*, Proc. SPIE, Medical Imaging 2009: Visualization, Image-Guided Procedures, and Modeling, Bd. 7261, 2009, S. 726122–726122–7.
- [27] B. Fischer und J. Modersitzki, *A super fast registration algorithm*, Bildverarbeitung für die Medizin, Bd. 169–173, 2001.
- [28] ———, *Curvature based registration with applications to MR-mammography*, Proceedings of the International Conference on Computational Science-Part III (London, UK), ICCS '02, Springer, 2002, S. 202–206.
- [29] ———, *Fast diffusion registration*, Contemporary Mathematics **313** (2002), 117–129.
- [30] O. Forster, *Analysis 2*, Differentialrechnung im \mathbb{R}^n , gewöhnliche Differentialgleichungen, Vieweg +Teubner, 2008.
- [31] ———, *Analysis 3*, Integralrechnung im \mathbb{R}^n mit Anwendungen, Vieweg +Teubner, 2009.
- [32] C. Geiger und C. Kanzow, *Numerische Verfahren zur Lösung unrestringierter Optimierungsaufgaben*, 1. Aufl., Springer Berlin Heidelberg, 1999.
- [33] F. Gigengack, L. Ruthotto, M. Burger, C. H. Wolters, X. Jiang und K. P. Schafers, *motion correction in dual gated cardiac PET using mass-preserving image registration*, IEEE Transactions on Medical Imaging **31** (2012), Nr. 3, 698 – 712.
- [34] P. E. Gill, W. Murray und M. H. Wright, *Practical optimization*, Academic Press, 1981.
- [35] M. W. Graham und W. E. Higgins, *Globally optimal model-based matching of anatomical trees*, Proc. SPIE, Medical Imaging 2006: Image Processing, Bd. 6144, 2006, S. 373–387.
- [36] M. Groher, *2D-3D registration of vascular images*, Dissertation, Technische Universität München, München, 2008.

- [37] M. Groher, D. Zikic und N. Navab, *Deformable 2D-3D registration of vascular structures in a one view scenario*, IEEE Transactions on Medical Imaging **28** (2009), Nr. 6, 847–860.
- [38] E. Haber, S. Heldmann und J. Modersitzki, *A computational framework for image-based constrained registration*, Linear algebra and its applications **431** (2009), 459–470.
- [39] E. Haber, R. Horesh und J. Modersitzki, *Numerical optimization for constrained image registration*, Numerical Linear Algebra with Applications **17** (2010), Nr. 2-3, 343–359.
- [40] E. Haber und J. Modersitzki, *Numerical methods for volume preserving image registration*, Inverse Problems **20** (2004), Nr. 5, 1621–1638.
- [41] ———, *Volume preserving image registration*, Medical Image Computing and Computer-Assisted Intervention **3216** (2004), 591–598.
- [42] ———, *Volume preserving image registration*, Medical Image Computing and Computer-Assisted Intervention - MICCAI (Haynor D.R. Barillot, C. und Hellier P., Hrsg.), Bd. 3216, Springer, 2004, S. 591–598.
- [43] ———, *Intensity gradient based registration and fusion of multi-modal images*, Methods of information in medicine **46** (2007), Nr. 3, 292–299.
- [44] ———, *Image registration with a guaranteed displacement regularity*, International Journal of Computer Vision **1**, 361–372.
- [45] H. Handels, *Medizinische Bildverarbeitung: Bildanalyse, Mustererkennung und Visualisierung für die computergestützte ärztliche Diagnostik und Therapie*, 2. Aufl., Vieweg +Teubner, 2009.
- [46] M. Hanke-Bourgeois, *Grundlagen der numerischen Mathematik und des wissenschaftlichen Rechnens*, 2. Aufl., Vieweg +Teubner, Dezember 2008.
- [47] S. Heldmann, *Multimodal registration of MR images with a novel least-squares distance measure*, Proc. SPIE **7623** (2010), 76230A–76230A–8.
- [48] S. Heldmann, Olesch J. Beuthien, B. und B. Fischer, *Improved minimal-invasive laparoscopic liver surgery by registration of 3D CT and 2D ultrasound slices*, BMT 2010 - FAL-Sessions, Hot-Topic Sessions, Projektssessions (BMT 2010 - FAL-Sessions - Hot-Topic Sessions - Projektssessions), 2010.
- [49] S. Heldmann und N. Papenberg, *A scale-space approach for image registration of vessel structures*, Bildverarbeitung für die Medizin (2009), 137–141.

-
- [50] M. J. Hohol, C. R. G. Guttmann, J. Orav, G. A. Mackin, R. Kikinis, S. J. Khoury, F. A. Jolesz und H. L. Weiner, *Serial neuropsychological assessment and magnetic resonance imaging analysis in multiple sclerosis.*, Archives of neurology **54** (1997), Nr. 8, 1018–1025.
- [51] K. Höllig, *Finite element methods with B-splines*, Society for Industrial Mathematics, 2003.
- [52] B. K. P. Horn und B. G. Schunck, *Determining optical flow*, Artificial intelligence **17** (1981), Nr. 1-3, 185–203.
- [53] A. K. Jain, *Fundamentals of digital image processing*, Prentice Hall, 1988.
- [54] M. Kass, A. Witkin und D. Terzopoulos, *Snakes: Active contour models*, International Journal of Computer Vision **1** (1988), Nr. 4, 321–331.
- [55] J. Kim, *Spatial normalization of brains with traumatic brain injury: A visual introduction*, Techn. Ber., Moss Rehabilitation Research Institute, 2009.
- [56] L. König, *Bildregistrierung mit Volumenbeschränkung*, Bachelorarbeit, Universität zu Lübeck, 2009.
- [57] R. K. S. Kwan, A. C. Evans und G. B. Pike, *MRI simulation-based evaluation of image-processing and classification methods.*, IEEE Transactions on Medical Imaging **18** (1999), Nr. 11, 1085–1097.
- [58] R. K. S. Kwan, A. Evans und G. B. Pike, *An extensible MRI simulator for post-processing evaluation*, Visualization in Biomedical Computing, Springer, 1996, S. 135–140.
- [59] Konrad Königsberger, *Analysis 1*, 6. auflage Aufl., Springer, 2003.
- [60] H. Lamecker und T. Lange, *Segmentation of the liver using a 3d statistical shape model*, Techn. Ber., ZIB, 2004.
- [61] T. Lange, N. Papenberg, J. Olesch, B. Fischer und P. M. Schlag, *Landmark constrained non-rigid image registration with anisotropic tolerances*, IFMBE Proceedings (Berlin, Heidelberg) (O. Dössel und W. C. Schlegel, Hrsg.), Bd. 25/4, Springer, 2010, S. 2238–2241.
- [62] T. Lange, S. Wörz, K. Rohr und P. M. Schlag, *Landmark-based 3D elastic registration of pre- and postoperative liver CT data an experimental comparison*, Bildverarbeitung für die Medizin (Berlin, Heidelberg) (W. Brauer, H. P. Meinzer, T. M. Deserno, H. Handels und T. Tolxdorff, Hrsg.), Springer, 2009.

- [63] M. Liguori, D. S. Meier, P. G. Hildenbrand, B. C. Healy, T. Chitnis, N. F. Baruch, S. J. Khoury, H. L. Weiner, R. Bakshi, F. Barkhof und C. R. G. Guttmann, *One year activity on subtraction MRI predicts subsequent 4 year activity and progression in multiple sclerosis*, Journal of Neurology, Neurosurgery & Psychiatry **82** (2011), Nr. 10, 1125–1131.
- [64] D. S. Meier und C. R. G. Guttmann, *MRI time series modeling of MS lesion development*, Neuroimage **32** (2006), Nr. 2, 531–537.
- [65] J. H. Metzen, T. Kröger, A. Schenk, S. Zidowitz, H. O. Peitgen und X. Jiang, *Matching of anatomical tree structures for registration of medical images*, Image and Vision Computing **27** (2009), Nr. 7, 923–933.
- [66] J. Modersitzki, *Numerical methods for image registration*, Oxford University Press, USA, 2004.
- [67] ———, *FLIRT with rigidity—image registration with a local non-rigidity penalty*, International Journal of Computer Vision **76** (2007), Nr. 2, 153–163.
- [68] J. Modersitzki, *FAIR: flexible algorithms for image registration*, SIAM, Philadelphia, 2009.
- [69] B. Moraal, I. J. van den Elskamp, D. L. Knol, B. M. J. Uitdehaag, J. J. G. Geurts, H. Vrenken, P. J. W. Pouwels, R. A. van Schijndel, D. S. Meier, C. R. G. Guttmann und F. Barkhof, *Long-interval T2w subtraction MRI: A powerful new outcome measure in MS trials*, Annals of Neurology **67** (2009), Nr. 5, 667–675.
- [70] B. Moraal, M. P. Wattjes, J. J. G. Geurts, D. L. Knol, R. A. van Schijndel, P. J. W. Pouwels, H. Vrenken und F. Barkhof, *Improved detection of active multiple sclerosis lesions: 3D subtraction imaging*, Radiology **255** (2010), Nr. 1, 154–163.
- [71] P. Nachev, E. Coulthard, H. R. Jäger, C. Kennard und M. Husain, *Enantiomorphic normalization of focally lesioned brains*, Neuroimage **39** (2008), Nr. 3, 1215–1226.
- [72] J. Nocedal und S. J. Wright, *Numerical optimization*, 2. Aufl., Springer Series in Operations Research and Financial Engineering, Springer, 2006.
- [73] J. Olesch, N. Papenberg, T. Lange, M. Conrad und B. Fischer, *Matching CT and ultrasound data of the liver by landmark constrained image registration*, Proc. SPIE, Medical Imaging 2009: Visualization, Image-Guided Procedures, and Modeling, Bd. 7261, 2009, S. 72610G–72610G–7.

-
- [74] G. Opfer, *Numerische Mathematik für Anfänger*, Eine Einführung für Mathematiker, Ingenieure und Informatiker., Vieweg + Teubner, Mai 2001.
- [75] A. V. Oppenheim, R. W. Schafer und J. R. Buck, *Zeitdiskrete Signalverarbeitung*, 2. Aufl., Pearson Studium, 2004.
- [76] N. Papenberg, *Ein genereller Registrierungsansatz mit Anwendung in der navigierten Leberchirurgie*, Dissertation, Universität zu Lübeck, Lübeck, 2008.
- [77] S. Periaswamy und H. Farid, *Medical image registration with partial data*, Medical Image Analysis **10** (2006), Nr. 3, 452–464.
- [78] P. Risholm, E. Samset, I. F. Talos und W. Wells, *A non-rigid registration framework that accommodates resection and retraction*, Information Processing in Medical Imaging (Berlin, Heidelberg), Bd. 5636, Springer, 2009, S. 447–458.
- [79] T. Rohlfing, C. R. Maurer, D. A. Bluemke und M. A. Jacobs, *Volume-preserving nonrigid registration of MR breast images using free-form deformation with an incompressibility constraint*, IEEE Transactions on Medical Imaging **22** (2003), Nr. 6, 730–741.
- [80] M. Rumpf und A. Telea, *A continuous skeletonization method based on level sets*, Proceedings of the symposium on Data Visualisation, VISSYM '02, 2002.
- [81] P. M. Schlag, S. Eulenstein, T. Lange und M. Kleemann, *Computerassistierte Chirurgie*, Urban Fischer in Elsevier, 2010.
- [82] M. Schünke, E. Schulte, U. Schumacher und M. Voll, *Prometheus Lernatlas der Anatomie*, 2. Aufl., Innere Organe, Georg Thieme Verlag, 2009.
- [83] M. Sdika, *A fast nonrigid image registration with constraints on the Jacobian using large scale constrained optimization*, IEEE Transactions on Medical Imaging **27** (2008), Nr. 2, 271–281.
- [84] M. Sdika und D. Pelletier, *Nonrigid registration of multiple sclerosis brain images using lesion inpainting for morphometry or lesion mapping*, Human Brain Mapping **30** (2009), Nr. 4, 1060–1067.
- [85] M. Staring, S. Klein und J. P. W. Pluim, *A rigidity penalty term for nonrigid registration*, Medical Physics **34** (2007), Nr. 11, 4098.
- [86] I.L. Tan und R. A. van Schijndel, *Serial isotropic three-dimensional fast FLAIR imaging: Using image registration and subtraction to reveal active multiple sclerosis lesions*, American Journal **179** (2002), Nr. 3, 777–782.

- [87] I.L. Tan, R. A. van Schijndel, F. Fazekas und M. Filippi, *Image registration and subtraction to detect active T2 lesions in MS: An interobserver study*, Journal of Neurology **249** (2002), Nr. 6, 767–773.
- [88] J. P. Thirion und G. Calmon, *Deformation analysis to detect and quantify active lesions in 3D medical image sequences*, Research Report, INRIA, 1997.
- [89] J. Tschirren, E. A. Hoffman und G. McLennan, *Branchpoint labeling and matching in human airway trees*, Proc. SPIE, Medical Imaging 2003: Physiology and Function: Methods, Systems, and Applications, Bd. 5031, 2003, S. 187–194.
- [90] J. Tschirren, G. McLennan, K. Palagyi, E. A. Hoffman und M. Sonka, *Matching and anatomical labeling of human airway tree*, IEEE Transactions on Medical Imaging **24** (2005), Nr. 12, 1540–1547.
- [91] N. J. Tustison, B. B. Avants, P. A. Cook, Yuanjie Z., A. Egan, P. A. Yushkevich und J. C. Gee, *N4ITK: Improved N3 bias correction*, IEEE Transactions on Medical Imaging **29** (2010), Nr. 6, 1310–1320.
- [92] K. Van Leemput, F. Maes, D. Vandermeulen und P. Suetens, *Automated model-based bias field correction of MR images of the brain.*, IEEE Transactions on Medical Imaging **18** (1999), Nr. 10, 885–896.
- [93] P. A. Viola, *Alignment by maximization of mutual information*, Dissertation, Massachusetts Institute of Technology, Boston, 1995.
- [94] J. Werner, *Numerische Mathematik*, Vieweg, 1992.
- [95] S. Wörz und K. Rohr, *Physics-based elastic registration using non-radial basis functions and including landmark localization uncertainties*, Computer Vision and Image Understanding **111** (2008), Nr. 3, 263–274.
- [96] D. Zikic, M. Groher und A. Khamene, *Deformable registration of 3D vessel structures to a single projection image*, Proc. SPIE, Bd. 6914, Medical Imaging 2008: Image Processing, Nr. 691412, 2008, S. 691412–691412–12.
- [97] B. Zitova und J. Flusser, *Image registration methods: A survey*, Image and Vision Computing **21** (2003), Nr. 11, 977–1000.