



UNIVERSITÄT ZU LÜBECK
INSTITUTE OF MATHEMATICS AND
IMAGE COMPUTING

Bildregistrierung mit Hilfe der robusten Hauptkomponentenanalyse

*Image registration using
robust principal component analysis*

Masterarbeit

im Rahmen des Studiengangs
Mathematik in Medizin und Lebenswissenschaften
der Universität zu Lübeck

vorgelegt von
Christoph Schulz

ausgegeben und betreut von
Prof. Dr. rer. nat. Jan Modersitzki
Institute of Mathematics and Image Computing

mit Unterstützung von
Kai Brehmer, M.Sc.
Institute of Mathematics and Image Computing

Lübeck, den 26. Juli 2019

Eidesstattliche Erklärung

Ich versichere an Eides statt, die vorliegende Arbeit selbstständig und nur unter Benutzung der angegebenen Hilfsmittel angefertigt zu haben.

Ort, Datum

Christoph Schulz

Kurzfassung

Die Bewegungskorrektur für kontrastmittelbasierte MRT Daten ist eine Herausforderung in der Bildregistrierung [13]. In dieser Arbeit wird ein möglicher Registrierungsansatz von Hamy u. a. [13] implementiert und mit Zeitreihenaufnahmen der menschlichen Niere getestet. Der Algorithmus liefert ähnliche Ergebnisse zum globalen Ansatz und bessere als der sequentielle Ansatz von Brehmer u. a. [3]. Darüber hinaus wird der Ansatz auf einem künstlich erzeugten Datensatz getestet, um die Tauglichkeit zu überprüfen. In einem ersten Schritt wird eine robuste Hauptkomponentenanalyse durchgeführt, welche eine gegebene Zeitreihe von Bildern in eine Matrix mit geringem Rang und eine dünn besetzte Matrix zerlegt. Dies ermöglicht die Bewegungskomponenten von den Intensitätsschwankungen zu trennen. Im zweiten Schritt wird die Bewegungskomponente unabhängig von dem Kontrastmittel registriert. In dieser Arbeit wird eine Einführung in die Bildregistrierung gegeben und das Registrierungsmodell beschrieben. Des Weiteren wird eine mögliche Erweiterung auf 3D beschrieben und zukünftige Aufgaben diskutiert.

Abstract

Motion correction for contrast enhanced MRI data is a challenge in image registration [13]. In the course of this work, a possible approach of Hamy et al. [13] was implemented and tested with time lapse images of the human kidney. The algorithm provides similar results to the global approach and better than the sequential approach from Brehmer et al. [3]. In addition, the implementation was tested on an artificially generated dataset to verify its suitability. In Hamy's approach, a robust principal component analysis is performed in a first step, which decomposes a given time series of images into a low rank matrix and a sparse matrix. This makes it possible to separate the motion components from the intensity fluctuations. In the second step, the motion component is registered independently of the contrast medium. In this thesis, an introduction to image registration is given first and then the registration model is described. Furthermore, a possible extension to 3D is described and tasks for future work are discussed.

Danksagung

An dieser Stelle möchte ich mich bei all denjenigen bedanken, die mich während der Anfertigung dieser Arbeit unterstützt haben.

Ein ganz besonderer Dank geht an meinen Betreuer Kai Brehmer, zu dem ich jeder Zeit mit Fragen ins Büro schneien konnte und der mir immer wieder neue Denkanstöße gegeben hat. Außerdem möchte ich mich bei Stephanie Häger für die Zeit im Studentenpool bedanken. Wir haben uns gegenseitig motiviert und dank ihr ist der Spaß bei der Arbeit nie zu kurz gekommen.

Des Weiteren geht ein großer Dank an meine Freundin die mich auch in Motivationslöchern immer wieder aus dem Bett gescheucht hat und die immer für mich da war. Zu guter Letzt möchte ich mich bei meiner Familie bedanken, die mich nicht nur während der Masterarbeit sondern während des gesamten Studiums begleitet und unterstützt hat.

Inhaltsverzeichnis

1	Einleitung	1
1.1	Motivation	2
1.2	Aufbau der Arbeit	3
2	Mathematische Grundlagen	5
2.1	Grundlagen der Bildregistrierung	5
2.2	Mathematische Optimierung	18
2.3	Robuste PCA	22
3	Registrierungsmodell	25
3.1	Modellübersicht	25
3.2	Methoden	27
3.2.1	Herleitung der Residual Complexity (RC)	27
3.2.2	Registrierungsmodell	31
4	Implementierung und Ergebnisse	33
4.1	Tests auf realen Daten	33
4.2	Tests auf künstlich erzeugten Daten	41
5	Fazit und Ausblick	45
	Literaturverzeichnis	50

1

Einleitung

Diese Arbeit beschäftigt sich mit der Bildregistrierung von medizinischen Bilddaten. Genauer werden Bilder aus einem dynamischen- kontrastmittelbasierten MRT verwendet. In der Bildregistrierung ist es das Ziel Bilder durch Deformationen möglichst ähnlich zu bekommen. Dabei können ganz unterschiedliche Ansätze verwendet werden. Allgemein lassen sich diese in zwei Arten unterscheiden. Zum einen die merkmalsbasierten Ansätze: Bei dieser Art der Herangehensweise erfolgt die Registrierung allein über merkmalspezifische Punkte, welche Landmarken genannt werden. Zum Anderen die intensitätsbasierten Ansätze: Bei diesen Ansätzen wird ein Distanzmaß zur Bestimmung der Ähnlichkeit definiert. Dieses operiert auf den Grauwerten der Bilder operiert und soll in der Regel minimiert werden. Da es sich hierbei um ein schlecht gestelltes Problem handelt, muss ein Regularisierer eingefügt werden, der die Plausibilität der Deformation beeinflusst.

In dieser Arbeit wird der von Hamy u. a. [13] eingeführte Ansatz vorgestellt und auf Daten der Niere und einem künstlich erzeugten Datensatz angewendet. Ziel dieses Ansatzes ist eine Bewegungskorrektur einer Zeitreihe von Bildern durchzuführen. Dazu werden die Bilder vor der Registrierung in zwei Anteile zerlegt. Für die Daten der menschlichen Niere soll der eine Anteil einen geringen Rang haben und hauptsächlich die Bewegungskomponenten enthalten. Der zweite Anteil soll wiederum die, vom Kontrastmittel hervorgerufenen, starken Unterschiede in den Intensitäten enthalten. Da diese Trennung in der Realität nicht exakt ist, wird über Zerlegung und Registrierung iteriert. Die aus diesem Ansatz erhaltenen Ergebnisse werden anschließend mit Ergebnissen aus Brehmer u. a. [3] verglichen und beurteilt.

1.1 Motivation

Die Bildregistrierung ist ein wichtiges Gebiet der Bildverarbeitung und findet in vielen Bereichen, wie Astronomie, Kunst, Biologie, Chemie und Medizin seine Anwendung [18]. In der Medizin sind die Bilder wegen verschiedenster Einflüsse, wie Herzschlag, Atmung, unterschiedliche Ausrichtung des Patienten etc. nicht immer direkt vergleichbar. So spielt die Bildregistrierung bei der Analyse solcher Daten unter anderem in der Tumorerkennung, OP-Planung oder beim Beobachten eines Krankheitsverlaufes eine immer größer werdende Rolle [2].

In der quantitativen Magnetresonanztomographie werden auf einer Zeitreihe von Bildern basierend quantitative Bildmerkmale bestimmt, welche die Gewebeeigenschaften charakterisieren. Wichtig dabei ist, dass die Bilder räumlich ausgerichtet sind, um eine zuverlässige Berechnung gewährleisten zu können [10]. Auch hier kann die Bildregistrierung helfen.

Ein üblicher Ansatz ist die paarweise Registrierung per Mutual Information, da die Aufnahmen häufig mit einem Kontrastmittel durchgeführt werden und so hohe Unterschiede in den Intensitäten auftreten können [10]. Dabei wird immer ein bewegtes Bild auf ein festes Referenzbild ausgerichtet. Sind mehr als zwei Bilder gegeben, ist die Frage wie das Referenzbild gewählt wird. In der Literatur sind dafür verschiedene Lösungen zu finden. Bei einigen Ansätzen wie z.B. dem von Seghers u. a. [22] wird eine paarweise Registrierung aller möglichen Bildpaare und die erhaltenen Transformationen miteinander kombiniert. Andere wie Guyader u. a. [10] oder Brehmer u. a. [3] versuchen eine gruppenweise durchzuführen, bei denen mehrere Bilder gleichzeitig registriert werden können.

In dieser Arbeit wird der Ansatz von Hamy u. a. [13] vorgestellt, implementiert und mit den Ansätzen aus Brehmer u. a. [3] verglichen. Zusätzlich werden Tests auf einem künstlich erzeugten Datensatz durchgeführt. Dieser Ansatz ist eher mit dem Ansatz von Seghers u. a. [22] zu vergleichen. Es wird jedoch eine robuste Hauptkomponentenanalyse als Vorverarbeitung durchgeführt. Dadurch soll die Bewegungskomponente vom Kontrastmittel basierten Teil zu trennen.

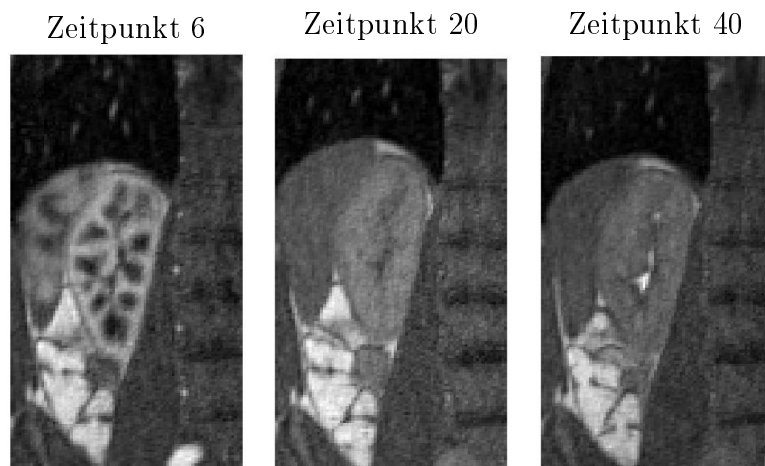


Abbildung 1.1. Drei repräsentative 2D Bilder des 4D Datensatzes der linken menschlichen Niere. Der Datensatz hat die Größe $146 \times 82 \times 52 \times 49$ und die abgebildeten Schichten die Größe 146×82

In Abbildung 1.1 sind beispielhaft drei Bilder einer Zeitreihe von kontrastmittelbasierten Daten der linken menschlichen Niere zu sehen, bei denen Hamys Algorithmus angewendet werden könnte. Es fällt auf, dass sich die Niere und umliegende Strukturen über die Zeit auf und ab bewegt. Des weiteren ist erkennbar, dass das Kontrastmittel zum Zeitpunkt 6 komplett in die Niere verteilt ist, während es zum Zeitpunkt 20 die Niere bereits verlassen hat. Aufgabe eines Ansatzes zur Bewegungskorrektur ist es nun, die Bewegungen des Gewebes zu registrieren, während die Dynamik, d.h. der Fluss des Kontrastmittels erhalten bleibt.

1.2 Aufbau der Arbeit

In Abschnitt 2 werden Grundlagen Bildregistrierung, Optimierung und zur robusten Hauptkomponentenanalyse gegeben. Diese sind für das spätere Verständnis des in dieser Arbeit verwendeten Modells nötig. Dann werden in Abschnitt 3 das Modell und die verwendeten Methoden vorgestellt, bevor in Abschnitt 4 die Ergebnisse der numerischen Experimente präsentiert werden.

Abschließend werden in Abschnitt 5 die Ergebnisse zusammengefasst und diskutiert, sowie ein Ausblick für spätere Arbeiten gegeben.

2

Mathematische Grundlagen

In diesem Kapitel werden die für das Modell von Hamy u. a. [13] nötigen mathematischen Grundlagen vorgestellt. Dazu werden die Daten vor der Registrierung in zwei Komponenten zerlegt. In der einen Komponente soll das Kontrastmittel und in der anderen die übrig gebliebenen Informationen enthalten sein. Dies geschieht über eine robuste Variante der Hauptkomponentenanalyse.

Für ein besseres Verständnis des in dieser Arbeit verwendeten Modells, wird in diesem Kapitel zunächst ein kurzer Einblick in die Grundlagen der nicht-parametrischen Bildregistrierung [17, 18] und der mathematischen Optimierung [20] gegeben. Dazu werden Bilder im mathematischen Sinne definiert und erklärt wie mit ihnen gearbeitet werden kann. Des Weiteren werden Beispiele für Distanzmaße und Regularisierer gegeben. Anschließend wird die robuste Variante der Hauptkomponentenanalyse [5] eingeführt.

2.1 Grundlagen der Bildregistrierung

Die Bildregistrierung ist eine sehr anspruchsvolle Aufgabe in der Bildverarbeitung. In diesem Abschnitt wird eine Einführung in die Grundlagen der nicht-parametrischen Bildregistrierung und die in dieser Arbeit verwendeten Methoden gegeben. Falls nicht anders erwähnt, wird sich in den folgenden Abschnitten auf Modersitzki [18] und Modersitzki [17] bezogen. Auch für eine allgemeine Einführung in die Registrierung wird auf diese Werke verwiesen.

Bei klassischen Problemen der Bildregistrierung sind ein Templatebild T und ein Referenzbild R gegeben. Diese können z.B. aus unterschiedlichen Perspektiven, zu unterschiedlichen Zeitpunkten, mit anderen Modalitäten oder von anderen Individuen erzeugt worden sein. Dabei können in der Theorie d -dimensionale Bilder als stetige Funktionen aufgefasst werden und sind wie folgt definiert.

Definition 2.1 (d-dimensionales Bild).

Sei $d \in \mathbb{N}$. Eine Abbildung $T : \mathbb{R}^d \rightarrow \mathbb{R}$ heißt d -dimensionales Bild, wenn

1. T einen kompakten Träger hat,
2. $0 \leq T(x) < \infty$ für alle $x \in \mathbb{R}^d$,
3. $\int_{\mathbb{R}^d} T(x)^k dx < \infty$ endlich für alle $k > 0$.

Ziel der Bildregistrierung ist es, für das Templatebild T eine Transformation $y : \mathbb{R}^d \rightarrow \mathbb{R}^d$ zu finden, so dass die beiden Bilder möglichst ähnlich sind, d.h. $R \approx T(y(x))$. Dabei kann die Transformation y in einen Identitätsteil und einen Deformationsteil u mit $u : \mathbb{R}^d \rightarrow \mathbb{R}^d$ d.h.

$$y(x) = x - u(x) \tag{2.1}$$

aufgeteilt werden. Diese Aufteilung wird in Abschnitt 2.1 eine Rolle spielen. Ein möglicher Ansatz ein solches Problem zu lösen ist es, ein Distanzmaß D zwischen den zu registrierenden Bildern T und R zu definieren und dieses bezüglich der Deformation u zu minimieren:

$$D[R, T; u] := D[R, T(x - u(x))] \xrightarrow{u} \min.$$

Dabei ist es notwendig, dass das Distanzmaß sein Minimum annimmt, wenn sich die Bilder am ähnlichsten sind. Dieser intuitive Weg hat jedoch einige Nachteile und ist dadurch nach Hadamard [12] ein schlecht gestelltes Problem. Zum Einen können kleine Änderung der Eingabe zu einer großen Änderung der Ausgabe führen können, zum anderen kann die Lösung nicht eindeutig sein, da das Problem nicht konvex ist. Außerdem kann die Transformation sogar nichtstetig sein.

Definition 2.2 (Korrektgestellttheit Hadamard [12]).

Ein mathematisches Problem heißt korrekt gestellt, wenn

- eine Lösung existiert
- die Lösung eindeutig ist
- die Lösung stetig von der Eingabe abhängt

Nicht korrekt gestellte Probleme heißen schlecht gestellte Probleme

Aus diesem Grund ist es nötig einen zusätzlichen Term, den Regularisierer S , hinzuzufügen. Dieser ermöglicht es bestimmte Transformationen zu bevorzugen und nicht stetige auszuschließen.

Aus einer Linearkombination von Distanzmaß und Regularisierer lässt sich ein Zielfunktional $J : \mathbb{R}^d \rightarrow \mathbb{R}$ aufstellen welches optimiert wird, um das Registrierungsproblem zu lösen.

Definition 2.3 (Variationeller Ansatz der Bildregistrierung).

Seien $T, R : \mathbb{R}^d \rightarrow \mathbb{R}$ Bilder, $y : \mathbb{R}^d \rightarrow \mathbb{R}^d$ eine unbekannte Transformation und $d \in \mathbb{N}$. Dann wird

$$J(y) = D[T(y), R] + \alpha S[y] \xrightarrow{y} \min \quad (2.2)$$

Registrierungsproblem genannt. Dabei bezeichne D das Distanzmaß, S den Regularisierer und $\alpha \in \mathbb{R}_+$ den Regularisierungsparameter

In dieser Arbeit wird mit der FAIR-Toolbox [17] gearbeitet. Diese verwendet einen „Discretize-then-Optimize,, Ansatz, um das Registrierungsproblem zu lösen. Dies bedeutet, dass das Registrierungsproblem zunächst diskretisiert wird, bevor die Optimierungsprobleme mittels newtonbasierter Verfahren gelöst werden. Warum eine Diskretisierung nötig ist, wie das Registrierungsproblem 3.12 gelöst werden kann, wie die Ähnlichkeit von Bildern gemessen werden kann und wie die Regularisierung funktioniert, wird in den folgenden Abschnitten erklärt.

Diskretisierung

In der Theorie können Bilder als stetige Funktionen aufgefasst werden. Dies ist jedoch in der Praxis, unter anderem auf Grund von begrenzter Speicherkapazität, nicht möglich, so dass die Bilder in der Regel, durch eine Abtastung, diskretisiert vorliegen. Dabei wird jedem Wert ein bestimmter Punkt auf einem Gitter zugeordnet. Das Gitter wird auf einem d -dimensionalen Bildgebiet $\Omega = (\omega^1, \omega^2) \times \dots \times (\omega^{2d-1}, \omega^{2d})$ definiert. Das Standardgitter der Größe $m = (m^1, \dots, m^d)$ ist das zellzentrierte Gitter 2.4, welches auch in dieser Arbeit verwendet wird.

Definition 2.4 (Zellzentriertes Gitter).

Sei $\Omega = (\omega^1, \omega^2) \times \dots \times (\omega^{2d-1}, \omega^{2d})$ ein Bildgebiet und $m = [m^1, m^2]$ die Bildgröße. Dann wird durch $h = [h^1, \dots, h^d]$ mit

$$h^i = \frac{\omega^{2i} - \omega^{2i-1}}{m^i}$$

die Gitterweite definiert. Ferner seien für $j = [j^1, \dots, j^d]$ die Zellmittelpunkte durch

$$\xi_j^i = \omega^{2i-1} + (j - 0.5)h^i \quad \text{mit } \xi^i = [\xi_1^i, \dots, \xi_{m^i}^i] \in \mathbb{R}^{m^i}$$

definiert. Dann ist das zellzentrierte Gitter durch

$$x_j := (\xi_{j^1}^1, \dots, \xi_{j^d}^d) \quad \text{mit } j^i = 1, \dots, m^i \quad \text{und } i = 1, \dots, d$$

definiert.

Neben dem zellzentrierten Gitter sind in der Literatur unter anderem auch das nodale oder das gestaffelte Gitter zu finden. Bei diesen Varianten wird nicht über die Zellmittelpunkte sondern über die Kanten bzw. Eckpunkte interpoliert.

Für die praktische Anwendung ist es wichtig in welcher Reihenfolge die Punkte im Vektor x_j gespeichert werden. In dieser Arbeit (siehe 2.4) wurde die lexikographische Anordnung genutzt. Der Ablauf hierfür ist in Abbildung 2.1 erklärt.

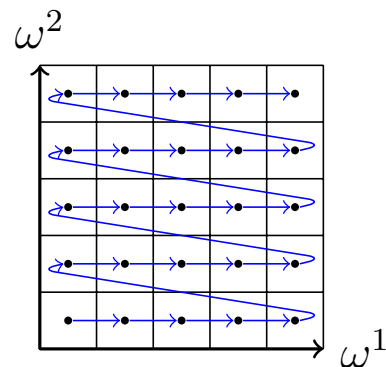


Abbildung 2.1. Die Punkte repräsentieren ein zellzentriertes Gitter und die Pfeile veranschaulichen die Reihenfolge, in der die Punkte im Vektor x_j bei einer lexikographischen Anordnung gespeichert werden.

Interpolation

Wie im vorherigen Abschnitt erwähnt ist das Referenzbild R lediglich auf einem Gitter gegeben. Um das transformierte Templatebild $T(y)$ am transformierten Gitter auswerten zu können, ist eine Interpolation nötig. Dazu wird eine Funktion \mathcal{T} mit $\mathcal{T}(x_i) = T(x_i) := T_i$ gesucht, welche im Hinblick auf die spätere Optimierung stetig und differenzierbar sein sollte. Im Folgenden werden verschiedene Ansätze der Interpolation für den 1D-Fall vorgestellt. Mit Hilfe des Kronecker Produktes können diese jedoch leicht in höhere Dimensionen überführt werden.

Definition 2.5 (Kronecker Produkt).

Seien zwei Matrizen $A \in \mathbb{R}^{m \times n}$ und $B \in \mathbb{R}^{p \times q}$ gegeben. Dann ist das Kronecker Produkt von A und B durch

$$A \otimes B := \begin{pmatrix} a_{1,1}B & \dots & a_{1,n}B \\ \vdots & \ddots & \vdots \\ a_{m,1}B & \dots & a_{m,n}B \end{pmatrix} \in \mathbb{R}^{mp \times nq}$$

definiert.

Nächste Nachbar Interpolation

Der einfachste Ansatz ist die Nächste-Nachbar-Interpolation. Dabei werden Punkte außerhalb des Bildgebietes auf 0 und Punkte innerhalb auf den Wert gesetzt zu dessen Zelle sie gehören.

Mathematisch lässt sich dies durch $\mathcal{T}^{NN} : \mathbb{R} \rightarrow \mathbb{R}$ mit

$$\mathcal{T}^{NN}(x) = \begin{cases} T_i & \text{falls } x \in \frac{-h}{2} \leq x - x_i \leq \frac{h}{2} \\ 0 & \text{sonst} \end{cases}$$

beschreiben. Die resultierende Interpolationsfunktion ist nicht stetig, an den Zellrändern nicht differenzierbar und die Ableitung innerhalb der Zellen ist stets 0. Diese Eigenschaften können während der später eingeführten Optimierung führen. Aus diesem Grund wird diese Art der Interpolation in dieser Arbeit nicht genutzt.

Lineare Interpolation

Eine weitere Methode ist die Lineare Interpolation. Diese ist durch $\mathcal{T}^{lin} : \mathbb{R} \rightarrow \mathbb{R}$ mit

$$\mathcal{T}^{lin}(x) = \begin{cases} (1 - \xi)T_i + \xi T_{i+1} & \text{falls } x_i \leq x \leq x_{i+1} \\ 0 & \text{sonst} \end{cases} \quad \text{für } \xi = \frac{x - x_i}{h}$$

definiert. Auch dieser Ansatz ist einfach zu implementieren und ist im Gegensatz zur Nächsten Nachbar Interpolation stetig. An den Gitterpunkten entstehen jedoch auch durch lineare Interpolation im Allgemeinen Stellen an denen nicht differenziert werden kann, weshalb in dieser Arbeit auch dieser Ansatz nicht verwendet wird.

Spline Interpolation

Im Sinne der eingangs erwähnten Bedingungen an \mathcal{T} liefert die Spline Interpolation die besten Ergebnisse. Ziel ist es eine Funktion $\mathcal{T}^{spline} : \mathbb{R} \rightarrow \mathbb{R}$ zu finden, welche die Daten interpoliert und die Biegeenergie

$$S[\mathcal{T}] = \int_{\Omega} (T''(x))^2 dx$$

minimiert. Die Lösung des Minimierungsproblems

$$S[\mathcal{T}] = \min \quad \text{u.d.N} \quad T(x_j) = T_j \quad j = 1, \dots, m$$

ist ein kubischer Spline, welcher durch eine Linearkombination von Koeffizienten c_j und Basisfunktionen b_j dargestellt werden kann. Durch Translationen eines sogenannten Muttersplines b_0 können die b_j berechnet werden. Für $b_j(x) = b_0(x - j)$ mit

$$b_0(x) = \begin{cases} (x+2)^3, & -2 \leq x < -1 \\ -x^3 - 2(x+1)^3 + 6(x+1), & -1 \leq x < 0 \\ x^3 + 2(x-1)^3 - 6(x-1), & 0 \leq x < 1 \\ (2-x)^3, & 1 \leq x < 2 \\ 0, & \text{sonst} \end{cases}$$

erhält man

$$\mathcal{T}^{spline}(x) = \sum_{j=1}^m c_j b_j(x).$$

Für die Berechnung der Koeffizienten $c = [c_1, \dots, c_m]$ muss das Gleichungssystem

$$B_m c = T,$$

gelöst werden, wobei T das diskretisierte Bild mit allen werten T_i repräsentiert. Dabei hat $B_m \in \mathbb{R}^{m \times m}$ folgende Gestalt:

$$B_m = \begin{pmatrix} 4 & 1 & & & 0 \\ 1 & 4 & 1 & & \\ & \ddots & \ddots & \ddots & \\ & & 1 & 4 & 1 \\ 0 & & & 1 & 4 \end{pmatrix}$$

Da $b_0(x) = 0$ für $x \notin (-2, 2)$ gilt für jeden Punkt $y = p + \xi$

$$\mathcal{T}^{spline} = c_{p-1}b(\xi + 1) + c_p b(\xi) + c_{p+1}b(\xi - 1) + c_{p+2}b(\xi - 2).$$

Im Gegensatz zu den bereits eingeführten Interpolationsansätzen, liefert die Spline Interpolation nicht nur ein stetiges sondern auch ein überall differenzierbares Ergebnis. Da bei späteren Arbeiten die Ableitung nötig ist, wird dieses Verfahren in dieser Arbeit genutzt.

Bis hier her wurde erläutert wie Bilder in der Theorie definiert werden können und wie mit ihnen gearbeitet werden kann. Im Hinblick auf die spätere Registrierung werden im Folgenden einige Ansätze beschrieben, wie die Distanz zwischen zwei Bildern berechnet werden kann.

Distanzmaße

Wie bereits erwähnt wird durch das Distanzmaß D die Ähnlichkeit zwischen zwei Bildern gemessen. Dabei sind je nach Ausgangslage verschiedene Methoden hilfreich. Grundlegend sind die merkmalsbasierten und die intensitätsbasierten Distanzmaße zu unterscheiden. Ein Beispiel für die merkmalsbasierten Distanzmaße ist der Abstand zwischen Landmarken. Dabei hängt das Ergebnis allein von der Wahl und dem Abstand korrespondierender Landmarken zueinander ab. In [18] werden solche Ansätze genauer betrachtet. Bei Intensitätsbasierten Distanzmaßen wird die Ähnlichkeit zweier Bilder über deren Grauwerte bestimmt. Einige Beispiele wie die Summe der quadrierten Differenzen (SSD), Normalisierte Gradientenfelder (NGF), Normalisierte Kreuzkorrelation (NCC), Mutual Information (MI) sind in der Literatur zu finden [18] und werden im Folgenden kurz vorgestellt.

Summe quadrierter Differenzen

Ein sehr intuitives Distanzmaß ist das SSD. Dieses vergleicht die Grauwertunterschiede in jedem Pixel, um die Ähnlichkeit zweier Bilder zu messen.

Definition 2.6 (SSD).

Sei $d \in \mathbb{N}$ und T, R Bilder. Das Distanzmaß der Summe der quadrierten Differenzen ist als

$$D^{SSD}[T, R] := \frac{1}{2} \int_{\mathbb{R}} (T(x) - R(x))^2 dx \quad (2.3)$$

definiert.

Ein großer Vorteil dieses Distanzmaßes ist es, dass es leicht zu interpretieren und implementieren ist. Die Einfachheit bringt allerdings auch Nachteile mit sich. Sind die zu registrierenden Bilder durch unterschiedliche Modalitäten erzeugt, können die Grauwerte möglicherweise nicht ohne weiteres miteinander verglichen werden. Des Weiteren können vereinzelt hohe Intensitätsunterschiede, die z.B. durch ein Kontrastmittel hervorgerufen werden, das Ergebnis stark beeinflussen.

Normalisierte Kreuzkorrelation

Ein Problem der SSD liegt im direkten Vergleich der Intensitäten von $T(y)$ und R . Ein Distanzmaß mit dem dieser direkte Zusammenhang geschwächt wird, ist die normalisierte Kreuzkorrelation [17]. Dabei werden die Grauwerte unter bestimmten Skalierungsfaktoren miteinander verglichen, wodurch das Maß robuster gegenüber unterschiedlichen Modalitäten wird. Mehr Informationen dazu sind in [17] zu finden.

Definition 2.7 (NCC).

Seien $d \in \mathbb{N}$ und T, R Bilder. Das Distanzmaß der Kreuzkorrelation ist als

$$D^{NCC}[T, R] = 1 - \frac{\langle T, R \rangle^2}{\|T\|^2 \|R\|^2} \quad (2.4)$$

gegeben, wobei $\|T\| = \sqrt{\langle T, T \rangle}$ bezeichne.

Normalisiertes Gradienten Feld

Eine weiteres Distanzmaß, welches das Problem der SSD bei multimodalen Bildern behebt, ist das normalisierte Gradienten Feld (NGF). Dieses Distanzmaß wurde erstmals von Haber, Modersitzki [11] vorgestellt und operiert auf den Gradienten der Bilder. Dabei wird ausgenutzt, dass Bilder als ähnlich angesehen werden können, wenn sie an den gleichen Stellen Änderungen der Intensität aufweisen. Es wird also versucht in den Bildern Kanten gleicher Richtung zu finden. Dabei muss der Grauwertunterschied an diesen Stellen nicht zwingend gleich groß sein. Aus diesem Grund wird eine Normalisierung durchgeführt.

Definition 2.8 (Normalisiertes Gradienten Feld).

Seien T und R Bilder. Dann ist das Distanzmaß NGF über

$$D^{NGF}[T, R] = \int_{\Omega} 1 - \left(\frac{\nabla T(y(x))^T \nabla R(x)}{\|\nabla T(y(x))\| \|\nabla R(x)\|} \right)^2 dx$$

definiert.

Mutual Information

Eine ganz andere Idee verfolgt die sogenannte Mutual Information (MI). Dieses Distanzmaß basiert auf der Entropie und kann wie folgt definiert werden [18].

Definition 2.9 (Entropie).

Sei $q \in \mathbb{N}$ und ρ mit $\rho : \mathbb{R}^d \rightarrow \mathbb{R}$ sei eine Dichte auf \mathbb{R}^d . Dann ist die Entropie über

$$H(\rho) := - \int_{\mathbb{R}^d} \rho \log \rho dx$$

definiert.

Definition 2.10 (Mutual Information).

Seien $d \in \mathbb{N}$ und R, T Bilder. Dann ist das Distanzmaß der Mutual Information D^{MI} durch

$$D[R, T] := H(\rho_R) + H(\rho_T) - H(\rho_{R,T})$$

definiert.

Ein Problem bei diesem Distanzmaß ist es, dass es hochgradig nicht konvex ist, d.h. viele lokale Minima auftreten und es wesentlich schwieriger zu implementieren ist als z.B. SSD. Auch die Parametereinstellungen sind nicht trivial. Trotzdem wird es häufig genutzt, da es meist sehr gute Ergebnisse auch für multimodale Daten liefert.

Regularisierer

Wie bereits einführend erwähnt ist ein Regularisierer notwendig, um eine plausible Transformation als Lösung des Registrierungsproblems zu erhalten. Durch das Hinzufügen des Terms $\alpha S[y]$ in 3.12 werden dem Problem zusätzliche Informationen gegeben. Der Parameter α soll dafür sorgen, dass die Lösung möglichst eindeutig ist und das Problem im Idealfall konvex wird. Dadurch soll sichergestellt werden, dass das Registrierungsproblem 3.12 korrekt gestellt 2.2 ist.

Je nach Bedarf können so unterschiedliche Eigenschaften der Transformation y erzwungen werden. Die meisten Ansätze basieren auf Varianten von L_2 Normen und beziehen sich meistens auf die in Abschnitt 2.1 beschriebene Deformation u . Im Folgenden werden einige mögliche Regularisierer vorgestellt.

Elastische Regularisierung

Die von Broit 1981 eingeführte elastische Regularisierung beruht auf Gleichungen aus der Elastizitätstheorie und ist damit physikalisch motiviert.

$$\mathcal{S}^{\text{elas}}[y] = \frac{1}{2} \int_{\Omega} \sum_{j=1}^d \mu \|\nabla u_j\|^2 + (\lambda + \mu)(\nabla u)^2 dx \quad (2.5)$$

Durch die Konstanten λ und μ , auch Lamé Konstanten genannt, können die Eigenschaften von elastischen Materialien modelliert werden. Für tiefer gehende Informationen wird auf [18] verwiesen.

Diffuse Regularisierung

Der Diffuse Regularisierer wurde erstmals von Fischer und Modersitzki [8] in die Bildregistrierung eingeführt. Seinen Ursprung hat dieser Regularisierer von den Euler-Lagrange Gleichungen und beschreibt eine generalisierte Diffusionsgleichung.

$$\mathcal{S}^{\text{diff}}[y] = \frac{1}{2} \sum_{j=1}^d \int_{\Omega} \|\nabla u_j\|^2 dx \quad (2.6)$$

Dieser Regularisierer bestraft den Gradienten der Transformation in jede Raumrichtung, wodurch Oszillationen in der Transformation unterdrückt werden können.

Krümmungsbasierte Regularisierung

Ein weiterer Regularisierer ist der krümmungsbasierte Regularisierer. Dabei wird durch Verwendung des Laplace Operators Δ versucht die Krümmung der Deformation zu beeinflussen.

$$\mathcal{S}^{\text{curv}}(y) = \frac{1}{2} \sum_{j=1}^d \int_{\Omega} \|\Delta u_j\|^2 dx \quad (2.7)$$

Dieser Ansatz wurde erstmals von Fischer und Modersitzki [8] in die Bildregistrierung eingeführt. Durch Betrachtung der zweiten Ableitungen anstelle der ersten, sind die Transformationen deutlich glatter als die der diffusen oder elastischen Regularisierung.

Multilevelregistrierung

Für eine effiziente Registrierung ist es häufig sinnvoll eine Vorregistrierung auf einer größeren Auflösung bzw. sogenannten Level der Bilder durchzuführen.

Definition 2.11 (Multileveldarstellung von 2D Daten [17]).

Seien $m^1 = m^2 = 2^L$ mit $L \in \mathbb{N}$ und ein Bild $T \in \mathbb{R}^{m^1 \times m^2}$ gegeben. Dann heißt $T^l, l = 0, \dots, L$ mit

$$T^{l-1} = (T^l(1 : 2 : m^1 - 1, 1 : 2 : m^2 - 1) + T^l(2 : 2 : m^1, 1 : 2 : m^2 - 1) + T^l(1 : 2 : m^1 - 1, 2 : 2 : m^2) + T^l(2 : 2 : m^1, 2 : 2 : m^2))/4$$

für $l = L : -1 : 1$ Multileveldarstellung von T

Dies bringt einige Vorteile mit sich. Erstens sind die Optimierungsprobleme auf größeren Auflösungen leichter und schneller zu lösen, da weniger Pixel miteinander verglichen werden müssen. Zweitens können die Ergebnisse aus Registrierungen auf größeren Auflösungen als Startwert für die Registrierung der nächst feineren Auflösung genutzt werden, wodurch auch hier Zeit gespart wird. Drittens sind auf größeren Auflösungen weniger Details zu erkennen, wodurch die Registrierung robuster wird und die Gefahr in lokalen Minima stecken zu bleiben reduziert wird.

2.2 Mathematische Optimierung

Bis hierher wurde erklärt wie die Ähnlichkeit von Bildern gemessen wird, daraus ein Registrierungsproblem aufgestellt werden kann und wie es reguliert werden kann. Offen ist wie ein solches Problem gelöst wird. In der Regel können gewöhnliche Verfahren der Optimierung das Registrierungsproblem 3.12 lösen. Im Folgenden Abschnitt wird eine kurze Einführung in die Optimierung gegeben und genauer auf das in dieser Arbeit verwendete Gauß Newton Verfahren eingegangen. Für tiefer gehende Informationen und weitere Ansätze der Optimierung wird auf [20] verwiesen.

Die Optimierung ist ein wichtiges Werkzeug der Entscheidungstheorie und findet in allen Bereichen der Naturwissenschaften seine Anwendung. Mathematisch gesehen kann die Optimierung als Minimierung oder Maximierung einer Zielfunktion bzgl. ihrer Variablen, welche gewissen Einschränkungen unterliegen können, beschrieben werden (2.12).

Definition 2.12 (Restringiertes Optimierungsproblem).

Seien $d \in \mathbb{N}$, $x \in \mathbb{R}^d$ der Vektor an unbekanntem Variablen, $f : \mathbb{R}^d \rightarrow \mathbb{R}$ die Zielfunktion und $c_i(x)$ mit $c : \mathbb{R}^d \rightarrow \mathbb{R}$ die Nebenbedingungen. Dann kann ein Optimierungsproblem als

$$\min_{x \in \mathbb{R}^d} f(x) \quad \text{u.d.N} \quad \begin{array}{ll} c_i(x) = 0 & i \in \mathcal{E} \\ c_i(x) \leq 0 & j \in \mathcal{I} \end{array} \quad (2.8)$$

definiert werden

Eine Möglichkeit Probleme der Form 2.8 zu unterscheiden ist die Aufteilung in sogenannte restringierte und unrestringierte Optimierungsprobleme.

Unrestringierte Probleme sind solche Probleme bei denen es keine Nebenbedingungen gibt oder sie keinen Einfluss auf das Ergebnis haben und ignoriert werden können d.h. $\mathcal{E} = \mathcal{I} = \emptyset$ in 2.8. In restringierten Problemen hingegen spielen die Nebenbedingungen eine zentrale Rolle und bestimmen die zulässige Menge, auf der optimiert wird.

Eine wichtige Bedeutung in der Optimierung hat das Konzept der Konvexität.

Definition 2.13 (Konvexität).

Eine Menge $A \subset \mathbb{R}^d$ heißt konvex, wenn

$$\lambda x + (1 - \lambda)y \in A$$

für alle $x, y \in A$ und $\lambda \in (0, 1)$.

Analog heißt eine Funktion f konvex, wenn

$$f(\lambda x + (1 - \lambda)y) \leq (1 - \lambda)f(x) + \lambda f(y) \quad \forall x, y \in \mathbb{R}^d, \lambda \in (0, 1) \quad (2.9)$$

gilt und strikt konvex falls $<$ in 2.9 gilt.

Mit Hilfe der Konvexität können Aussagen über die Lösung eines Optimierungsproblems getroffen werden. So ist jedes lokale Minimum gleichzeitig auch ein globales Minimum, falls die Zielfunktion konvex ist. Ist die Zielfunktion strikt konvex, ist die Lösung sogar eindeutig.

Optimierungsalgorithmen

Optimierungsalgorithmen sind iterative Algorithmen, bei denen ein Startwert x_0 so lange aktualisiert bzw. verbessert wird, bis $f(x_k) = f(x^*)$ im Optimalfall das Minimum annimmt. Gute Algorithmen sollten robust, effizient und möglichst genau sein. In den meisten Fällen muss ein Kompromiss zwischen diesen Eigenschaften gefunden werden, da z.B. sehr schnell konvergierende Algorithmen sehr ungenau sein können oder robuste Ansätze möglicherweise sehr langsam sind. Die Algorithmen unterscheiden sich in der Regel darin wie von einer Iteration in die nächste gegangen wird. Dabei nutzen die meisten Strategien die Werte der Zielfunktion f , die Nebenbedingungen c_i und die ersten und zweiten Ableitungen der Zielfunktion.

Bei sogenannten Line Search Ansätzen wird in jeder Iterierten eine Suchrichtung $p_k \in \mathbb{R}^d$ berechnet und festgelegt wie weit in diese Richtung gegangen wird. Seien f und x wie in Definition 2.12 gegeben. Dann können die Iterierten über

$$x_{k+1} = x_k + \alpha_k p_k$$

berechnet werden, wobei $\alpha_k \in \mathbb{R}$ die Schrittweite angibt. Beim Newtonverfahren wird die Suchrichtung über die Taylorentwicklung zweiter Ordnung der Zielfunktion

$$f(x_k + p) = f_k p^T \nabla f_k + \frac{1}{2} p^T \nabla^2 f_k p = m_k(p)$$

berechnet. Angenommen $\nabla^2 f_k$ ist positiv definit, dann entspricht die Suchrichtung dem Vektor p , welcher $m_k(p)$ minimiert. Durch differenzieren und nullsetzen erhält man

$$p_k = -(\nabla^2 f_k)^{-1} \nabla f_k$$

als Suchrichtung des Newtonverfahrens. Für den Fall, dass $\nabla^2 f_k$ nicht positiv definit ist, ist die Suchrichtung möglicherweise nicht definiert oder erfüllt die Abstiegseigenschaft $\nabla f_k \nabla^2 f_k < 0$ nicht mehr und muss angepasst werden. Zudem ist es oft umständlich, fehleranfällig oder teuer die Hessematrix $\nabla^2 f_k$ zu bestimmen.

Diesen Nachteil umgehen Quasi-Newton verfahren, indem sie die Hessematrix lediglich approximieren. Eine Möglichkeit wird beim Gauss-Newton Verfahren genutzt, bei welchem die Residuen r_k der gegebenen Funktion via kleinste Quadrate Methode minimiert werden. Die Hessematrix wird hierbei über die Jakobimatrix J mit $\nabla^2 f_k \approx J_k^T J_k$ approximiert. Anschließend wird die Suchrichtung nicht direkt über die Newton Gleichung $\nabla^2 f(x_k) p = -\nabla f(x_k)$ sondern durch Lösen des Gleichungssystems

$$J_k^T J_k p_k^{GN} = -J_k^T r_k$$

berechnet.

Wie weit in diese Richtung gegangen wird, kann auf verschiedene Arten festgelegt werden. In dieser Arbeit wurde die Armijo Schrittweite

$$f(x_k + \alpha p_k) \leq f(x_k) + c_1 \alpha \nabla f_k^T p_k,$$

mit $c \in (0, 1)$ genutzt, d.h. der Abstieg von f soll proportional zu der Schrittweite α_k und der Richtungsableitung $\nabla f_k^T p_k$ sein. Für weitere Details wird hier auf [20] verwiesen.

Für die restringierte Optimierung ist ein möglicher Lösungsweg die Umformulierung der Zielfunktion. Dabei werden die Nebenbedingungen direkt in die umformulierte Zielfunktion aufgenommen. Einen solchen Ansatz beschreiben die sogenannten Lagrange Multiplikatoren. Dabei wird die neue Zielfunktion aus einer Linearkombination der eigentlichen Zielfunktion und den Nebenbedingungen zusammengesetzt. Die Faktoren der Nebenbedingung werden Lagrange Multiplikatoren genannt. Betrachtet man z.B. das lineare Optimierungsproblem [6]

$$\min_{x \in \mathbb{R}^n} a^T x \quad \text{u.d.N} \quad \begin{array}{l} Ax \leq b \\ x \geq 0 \end{array}$$

mit $A \in \mathbb{R}^{m \times n}$, $x \in \mathbb{R}^n$, $b \in \mathbb{R}^m$, $x \in \mathbb{R}^n$, erhält man die Lagrangefunktion $L(x, \lambda) = c^T x + \lambda^T (Ax - b)$. Hier ist $\lambda \in \mathbb{R}$ der Lagrange Multiplikator [6]. Sobald jedoch nicht lineare Optimierungsprobleme betrachtet werden, wird die Theorie der Lagrange Multiplikatoren etwas umfangreicher. Eine Erweiterung dieses Verfahrens ist die sogenannte augmented Lagrange Methode. Bei dieser Methode wird der Lagrangefunktion zusätzlich ein quadratischer Strafterm hinzugefügt. Dieser soll die Konvexität des Optimierungsproblems beeinflussen und das Lösen damit erleichtern. Angenommen es sei ein Optimierungsproblem wie in Definition 2.12 gegeben. Dann ist die augmented Lagrangefunktion [20] durch

$$l(x, \lambda, \mu) := f(x) - \sum_{i \in \mathcal{E}} \lambda_i c_i(x) + \frac{\mu}{2} \sum_{i \in \mathcal{E}} c_i^2(x) \quad (2.10)$$

mit $\lambda, \mu \in \mathbb{N}$ definiert. Lösungsansätze, welche auf dieser Funktion basieren, fixieren zunächst λ , um eine Näherung für den Lagrangemultiplikator zu erhalten. Anschließend wird μ fixiert, um einen Wert für x zu finden, welcher l ungefähr minimiert. Diese Lösungsmethode ist ein Sonderfall einer allgemeineren Klasse von Algorithmen die als sogenannte alternating directions Methode (ADM) bekannt ist [5]. Für genauere Informationen hierzu wird auf [23] verwiesen.

2.3 Robuste PCA

Die Hauptkomponentenanalyse (engl. principal component analysis kurz: PCA) ist ein wichtiges Verfahren um große Datensätze zu strukturieren, zu veranschaulichen und vor allem die Dimensionen zu reduzieren. Allerdings ist der klassische Ansatz sehr empfindlich gegenüber Rauschen und Ausreißern [5]. Heutzutage sind verrauschte Daten allgegenwärtig und gerade bei Zeitreihen von kontrastmittelbasierten MRT können die Intensitätsunterschiede sehr groß sein.

Aus diesem Grund ist es wichtig dieses Verfahren zu verbessern und robuster gegenüber solchen Fehlerquellen zu machen. In diesem Abschnitt wird eine robuste Variante der PCA von Candès u. a. [5] eingeführt, welche eine gegebene Matrix in eine Matrix mit geringem Rang (Low-rank) und einen dünn besetzten bzw. Sparseanteil zerlegt.

Gegeben sei eine Matrix M von der angenommen wird, dass sie in

$$M = L_0 + S_0,$$

zerlegt werden kann. Dabei sei L_0 der Teil mit geringem Rang und S_0 der dünn besetzte Anteil. Wichtig zu erwähnen ist, dass die Gestalt des niedrigdimensionalen Spalten- bzw. Zeilenraumes von M und der Ort, wo sich die von Null verschiedenen Einträge in S befinden, unbekannt sind. Dieser Ansatz könnte für verschiedene Anwendungen hilfreich sein. So könnte man in der Videoüberwachung die einzelnen Frames als Spalten in der Matrix M speichern und sich den Hintergrund als Low-rank Komponente und sich bewegende Objekte als Sparse Komponente vorstellen. Ein ähnlicher Ansatz wird in dieser Arbeit betrachtet. Die Matrix M wird mit Daten aus einem dynamischen, kontrastmittelbasierten MRT gefüllt, das Kontrastmittel soll in die Sparse Komponente und der Rest in dem Low-rank Anteil gespeichert werden.

Zunächst scheint es unmöglich ein solches Problem zu lösen, da die Menge an Unbekannten in L_0 und S_0 doppelt so groß ist wie die Anzahl an Messwerten. Überraschenderweise ist dieses Problem durch eine konvexe Optimierung lösbar.

Dazu bezeichne

$$\|M\|_* := \sum_i \sigma_i(M)$$

die Nukleornorm von M , wobei $\sigma_i(M)$ die Singulärwerte von M sind. Diese entstammen einer Singulärwertzerlegung von M , welche im Folgenden definiert wird.

Definition 2.14 (Singulärwertzerlegung).

Sei $M \in \mathbb{R}^{m \times n}$ mit $m > n$ und $m, n \in \mathbb{N}$. Dann existieren Matrizen $U \in \mathbb{R}^{m \times m}$ und $V \in \mathbb{R}^{n \times n}$ sowie eine Diagonalmatrix $\Sigma = \text{diag}(\sigma_1, \dots, \sigma_r) \in \mathbb{R}^{m \times n}$, so dass

$$M = U \Sigma V^T \tag{2.11}$$

gilt. Weiter gilt $\sigma_1, \dots, \sigma_r > 0$ und somit $\text{rang}(M) = r \in \mathbb{N}$

Diese Norm kann als konvexe Annäherung der Rangfunktion verstanden werden. Im Hinblick darauf, dass die eine Komponente geringen Rang haben soll, macht die Verwendung dieser Norm Sinn. Mit

$$\|M\|_1 := \sum_{ij} |M_{ij}|$$

wird die unserem Fall die 1-Norm von M bezeichnet. Diese dient dazu zu messen wie dünn besetzt eine Matrix ist. Mit Hilfe dieser Normen, können L und S durch lösen von

$$\begin{aligned} \min \quad & \|L\|_* + \lambda \|S\|_1 \\ \text{u.d.N.d} \quad & L + S = M \end{aligned} \tag{2.12}$$

exakt aus M gewonnen werden, wobei $\lambda \in \mathbb{N}$. Für genauere Informationen wann eine solche Zerlegung Sinn macht und unter welchen Voraussetzungen diese Zerlegung exakt ist, wird auf Candès u. a. [5] verwiesen. Zum Lösen des Problems 2.12 wird die in Abschnitt 2.2 eingeführte augmented Lagrange Methode [16, 20] genutzt. Diese erzeugt eine Lösung durch Optimieren der erweiterten Lagrangefunktion [1, 20]

$$l(L, S, Y) = \|L\|_* + \lambda \|S\|_1 + \langle Y, M - L - S \rangle + \frac{\mu}{2} \|M - L - S\|_F^2. \tag{2.13}$$

Ein Ansatz ist es iterativ zunächst $(L_k, S_k) \in \arg \min_{L,S} l(L, S, Y_k)$ zu setzen und anschließend den Lagrange Multiplikator $Y_{k+1} = Y_k + \mu(M - L_k - S_k)$ zu aktualisieren. Details zu diesem Ansatz sind in [20] zu finden. Wie in [23] nachzulesen ist, haben $\min_L l(L, S, Y)$ und $\min_S l(L, S, Y)$ einfache Lösungen.

Sei $S_\tau : \mathbb{R} \rightarrow \mathbb{R}$ mit $S_\tau[x] = \text{sgn}(x) \max(|x| - \tau, 0)$ der sogenannte Shrinkage Operator, welcher für Matrizen eintragsweise zu verstehen ist, dann gilt

$$\operatorname{argmin}_S l(L, S, Y) \ni S_{\lambda/\mu}(M - L + \mu^{-1}Y).$$

Analog sei $\mathcal{D}_\tau(X)$ mit $\mathcal{D}_\tau(X) = US_\tau(\Sigma)V^*$ der Singulär-Schwellwert Operator, wobei $X = U\Sigma V^*$ eine Singulärwertzerlegung von X ist, dann gilt

$$\operatorname{argmin}_L l(L, S, Y) \ni D_{1/\mu}(M - S + \mu^{-1}Y).$$

Der daraus resultierende iterative Ansatz, l zunächst bzgl. L bei festem S zu minimieren, dann l bzgl. S bei festem L zu minimieren und anschließend den Lagrangemultiplikator Y zu aktualisieren, ist in Algorithmus 1 zusammengefasst.

Algorithm 1: Principal Component Pursuit [5]

Data: Matrix M

Result: Low-rank Matrix L und Sparse Matrix S

initialize: $S_0 = Y_0 = 0, \mu \in \mathbb{R}$ mit $\mu > 0$

while not converged **do**

$$L_{k+1} = D_{1/\mu}(M - S_k + \mu^{-1}Y_k)$$

$$S_{k+1} = S_{\mu/\lambda}(M - L_{k+1} + \mu^{-1}Y_k)$$

$$Y_{k+1} = Y_k + \mu(M - L_{k+1} - S_{k+1})$$

end while

Wie die Parameter μ, λ und die Abbruchbedingung gewählt wurden, wird in Abschnitt 3 beschrieben.

3

Registrierungsmodell

Im folgenden Abschnitt wird der von Hamy u. a. [13] entwickelte Algorithmus für die Bewegungskorrektur von Daten aus einem dynamischen kontrastmittelbasierten MRT (Robust Data Decomposition Registration", kurz RDDR) vorgestellt. Dazu wird zunächst ein grober Überblick über das Modell gegeben und anschließend genauer auf die verwendeten Methoden eingegangen.

3.1 Modellübersicht

Der Prozess des RDDR wird in Abbildung 3.1 veranschaulicht und kann wie folgt beschrieben werden. Gegeben sei eine Zeitreihe von kontrastmittelbasierten MRT Bildern, welche in eine Matrix M überführt wird und dann durch eine robuste PCA mit Startwert λ_{init} zerlegt wird. Anschließend werden die so erhaltenen Matrizen mit geringem Rang aufeinander registriert. Die dadurch erhaltenen Deformationsfelder werden auf die Original Zeitreihe angewandt, so dass ein Teil der Bewegung entfernt werden kann. Dieser Ablauf wird dann mit steigendem λ über eine fixe Anzahl an Iterationen wiederholt.

Da durch die robuste PCA die Bewegungskomponenten nicht exakt aus den Daten gefiltert werden können, ist der iterativer Algorithmus gewählt worden. Der mit den Iterationen steigende Parameter λ ermöglicht es den Anteil der Bewegungskomponente in der Low-rank Matrix zu regulieren.

Dieser Algorithmus wurde im Zuge dieser Arbeit in Matlab implementiert und kann als Ergänzung der FAIR Toolbox [17] verwendet werden.

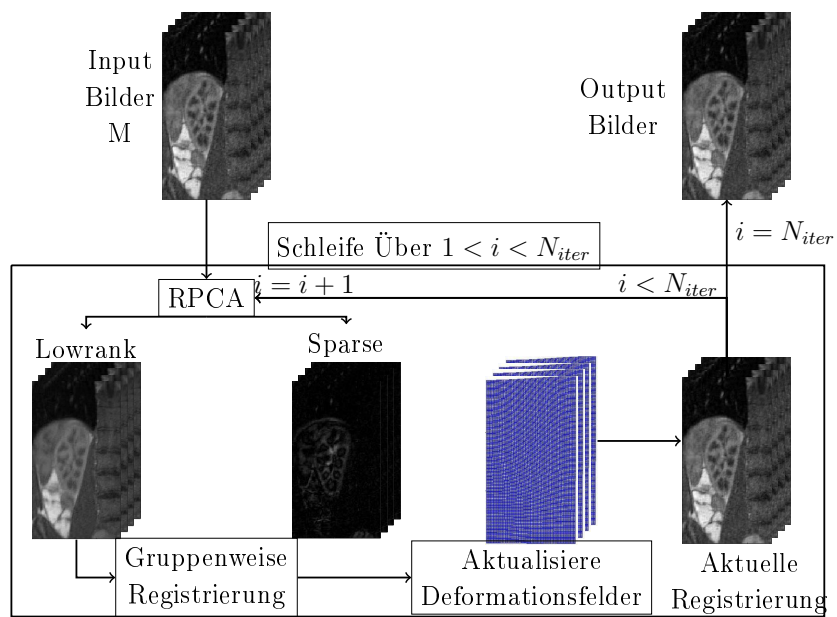


Abbildung 3.1. Diagramm welches den Algorithmus von Hamy u. a. [13] veranschaulicht.[13]

3.2 Methoden

Im Folgenden wird genauer auf die verwendeten Methoden eingegangen. Die robuste PCA wurde naiv nach dem in Abschnitt 2.3 eingeführten Algorithmus 1 implementiert. Für die Regularisierung wurde der krümmungsbasierte Regularisierer Gleichung 2.7 und als Distanzmaß wurde eines, welches der SSD 2.3 ähnlich ist, verwendet. Da die robuste PCA und der Regularisierer schon in den vorherigen Abschnitten eingeführt wurden, wird in diesem Abschnitt lediglich Residual Complexity als neues Distanzmaß vorgestellt.

3.2.1 Herleitung der Residual Complexity (RC)

Wie in Abschnitt 2.1 erwähnt wurde, ist es sinnvoll bei Bildern mit sehr unterschiedlichen Intensitätswerten ein Distanzmaß zu nutzen, welches diesen Unterschied kompensiert. Einige mögliche Ansätze die den direkten Vergleich der Intensitäten zu umgehen, ist es andere Bildmerkmale, wie z.B. Kanten, zum Vergleich heranzuziehen (vgl. Abschnitt 2.1). In diesem Abschnitt wird ein weiterer Ansatz, welcher auf SSD beruht, beschrieben und hergeleitet. Dabei soll der Intensitätsunterschied durch eine Korrektur I ausgeglichen werden und die Einfachheit von SSD bewahrt werden.

Es seien zwei Bilder $R, T \in \mathbb{R}^{d \times d}$ gegeben, wobei der Zusammenhang

$$R = T(y) + I$$

angenommen wird. Dabei bezeichne $y : \mathbb{R}^d \rightarrow \mathbb{R}^d$ die Transformation und $I \in \mathbb{R}^{(d \times d)}$ die Intensitätskorrektur. Dann kann ein zur SSD ähnliches Distanzmaß wie folgt definiert werden:

$$E(I, y) = \min \|R - T(y) - I\|^2 + \beta \|PI\|_2, \quad (3.1)$$

wobei $\|\cdot\|$ die Euklidische Norm und P den Regularisierungsoperator für I beschreibt. Durch analytisches Lösen nach I , kann dieser Term aus der Distanzfunktion eliminiert werden. Differenzieren und lösen nach I liefert:

$$\nabla_I E(I, y) = -2(R - T(y) - I) + 2\beta P^T P I \stackrel{!}{=} 0 \quad (3.2)$$

$$\Leftrightarrow I = \underbrace{(Id + \beta P^T P)^{-1}}_V r. \quad (3.3)$$

Hierbei bezeichne $r = R - T(y)$ das Differenzbild von R und $T(y)$ und Id die Einheitsmatrix. Es ist sichergestellt das V existiert, da $P^T P$ positiv semidefinit ist. Des Weiteren ist V symmetrisch, wodurch einsetzen von 3.2 in 3.1

$$\begin{aligned} E(y) &= \|r - Vr\|^2 + \beta \|P^T P r\|^2 \\ &= r^T ((Id - V)^T (Id - V)) r + \beta r^T (V^T P^T P V) r \\ &= r^T (Id - 2V + V^T (Id + \beta P^T P) V) r \\ &= r^T (Id - 2V + V) r \\ &= r^T (Id - V) r \\ &= r^T (Id - (Id + \beta P^T P)^{-1}) r \end{aligned} \quad (3.4)$$

als Distanzfunktion liefert, welche nun nicht mehr von S abhängig ist. Um 3.4 weiter zu vereinfachen, wird nun die Matrix $P^T P$ genauer betrachtet. Da $P^T P$ in jedem Fall quadratisch, symmetrisch und positiv semidefinit ist, existiert eine Eigenwertzerlegung

$$P^T P = Q \Lambda Q^T \quad \Lambda = [\lambda_1, \dots, \lambda_N], \quad \lambda_i \geq 0 \quad (3.5)$$

mit einer Diagonalmatrix Λ , welche die Eigenwerte beinhaltet und einer orthogonalen Matrix Q , welche die Eigenvektoren von $P^T P$ enthält. Durch Substituieren von 3.5 in 3.4 wird die Distanzfunktion weiter vereinfacht und man erhält

$$\begin{aligned} E(y) &= r^T Q \text{diag} \left(1 - \frac{1}{1 + \beta \lambda_i} \right) Q^T r \\ &= r^T Q \text{diag} \left(\frac{\beta \lambda_i}{1 + \beta \lambda_i} \right) Q^T r \\ &= r^T Q L Q^T r. \end{aligned}$$

3.2 Methoden

Wenn nun ein geeigneter Regularisierungsoperator P gewählt wird, ist $A = QLQ^T$ bekannt und das Minimierungsproblem

$$\begin{aligned} \min_y E(y) &= \min_y (R - T(y))^T A (R - T(y)) \\ &= \min_y (Qr)^T L Q^T r \end{aligned} \quad (3.6)$$

kann bezüglich y gelöst werden. Bei diesem Ansatz wird jedoch lediglich Q im Vorwege festgesetzt und L während der Optimierung berechnet. Dazu wird

$$E(y, L) = (Q^T r)^T L (Q^T r)$$

betrachtet. Sucht man nach dem Minimum bezüglich der Eigenwerte von L , so ist eine trivial Lösung die Nullmatrix, welche jedoch nicht von Interesse ist. Um dies zu verhindern wird eine Regularisierung für die Eigenwerte in L eingeführt, die dafür sorgt, dass die Einträge in L möglichst gleichverteilt sind. Dabei sei bemerkt, dass, falls L der Einheitsmatrix entspricht, sich die Distanzfunktion 3.6 zur im vorherigen Abschnitt eingeführten SSD vereinfacht:

$$E(y, Id) = (Q^T r)^T Id (Q^T r) = r^T r.$$

Dies entspricht dem Fall, dass keine Intensitätskorrektur, sprich $S = 0$ durchgeführt wird. SSD liefert optimale Ergebnisse, wenn die Daten mit einem unabhängigen identisch verteilten Rauschen behaftet sind.

Je mehr die Verteilung der Eigenspektren in L von der Gleichverteilung abweicht, desto mehr Werte enthält QLQ^T auf den Nebendiagonalen. Dies bedeutet, dass das Rauschen auf den Daten stärker korreliert ist. Da die Verteilung des Rauschens im Allgemeinen unbekannt ist, wird hier die Annahme getroffen, dass das Rauschen möglichst unkorreliert ist, d.h die Werte in L sollen möglichst gleichverteilt sein. Dies kann durch einen, auf der Kullback-Leibler Divergenz beruhenden Regularisierungsterm zwischen der Gleichverteilung und der Verteilung der Eigenwerte in L

$$KL(L) = - \sum_i \log(l_i) + \sum_i l_i \quad (3.7)$$

erreicht werden. Der erste Term sorgt dafür, dass die Werte in L nicht negativ und gleichverteilt sind, während der zweite Term dafür sorgt, dass die

Eigenwerte nicht zu groß werden. Durch diese Art der Regularisierung wird die Nähe zu den Daten gewährleistet, so dass eine Informationsmessung und keine Entfernungsmessung durchgeführt wird. Durch Einfügen des Regularisierungsterm 3.7 erhält man

$$E(L, y) = (Q^T r)^T L(Q^T r) + \alpha KL(L) \quad (3.8)$$

als Distanzmaß, wobei der Parameter $\alpha \in \mathbb{N}_0$ ist. Durch Differenzieren bzgl. l_i und gleichsetzen mit 0 erhält man

$$(q_i r)^2 - \alpha \frac{1}{l_i} + \alpha = 0 \Rightarrow l_i = \frac{1}{(q_i^T r)^2 / \alpha + 1}. \quad (3.9)$$

Dabei bezeichnen q_i die Eigenvektoren in $Q = [q_1, \dots, q_N]$. Setzt man 3.9 in 3.8 ein so erhält man

$$E(y) = \alpha N - \alpha \sum_{n=1}^N \log \frac{1}{(q_n^T r)^2 / \alpha + 1}$$

Die Terme, welche nicht von y abhängen können ignoriert werden, das sie keinen Einfluss auf die Lage des Minimums haben. Dadurch erhält man

$$E(y) = \sum_{n=1}^N \log ((q_n^T r)^2 / \alpha + 1) \quad (3.10)$$

als vorläufiges Distanzmaß. Allerdings ist dies noch nicht die finale Form des Distanzmaßes, da die genaue Form der Eigenvektoren in Q noch nicht definiert wurde.

Die orthogonale Matrix Q besteht aus den Eigenvektoren von $P^T P = Q^T \Lambda Q^T$, wobei P der Regularisierungsoperator für das Korrekurfeld I ist. Als Q wird die diskrete Kosinus Transformation (DCT) gewählt. Dies hat zwei Vorteile. Für den Fall einer Matrixmultiplikation entspricht $Q^T r$ der DCT von r und ist mittels FFT schnell zu berechnen, d.h.

$$\text{DCT}(r) := Q^T r.$$

Zum einen korrespondieren die DCT Eigenvektoren zum diskreten ableitungsbasierten Regularisierer P und zum anderen sind es Eigenvektoren der Kovarianzmatrix eines schwach stationären Prozessen, was die Annahme eines schwach stationären Residuums impliziert. Das daraus erhaltene Distanzmaß wird Residual Complexity genannt.

Definition 3.1 (Residual Complexity).

Seien $d \in \mathbb{N}$ und $R, T \in \mathbb{R}^d$ vektorisierte Bilder. Ferner sei mit $N \in \mathbb{N}$ die Anzahl an Eigenvektoren von Q aus Gleichung 3.5 bezeichnet. Dann ist das Distanzmaß der Residual Complexity $D^{RC} : \mathbb{R}^d \rightarrow \mathbb{R}$ durch

$$D^{RC}[R, T] := \sum_{i=1}^N \log \left(\frac{c^2}{\alpha} + 1 \right) \quad (3.11)$$

mit $c = DCT(r)$ und $r = R - T(y)$ definiert.

Analog zu SSD oder NGF kann auch für dieses Distanzmaß eine intuitive Erklärung für die Idee gegeben werden.

Seien mit x die DCT-Koeffizienten des Differenzbildes r bezeichnet. Dann kann das Bild stark komprimiert werden, wenn nur wenige Koeffizienten ungleich von null sind. Daraus lässt sich ableiten, dass zwei Bilder als ähnlich angesehen werden können, wenn die Kosinustransformation des Differenzbildes möglichst wenige Koeffizienten besitzt.

Das in Definition 3.1 eingeführte Distanzmaß verfolgt genau diese Idee. Es bestraft kleine Koeffizienten überproportional stark zu den großen. Dadurch erreicht das Distanzmaß sein Optimum, wenn die Kosinustransformierte des Differenzbildes möglichst wenige Koeffizienten besitzt. Diese Lösung korrespondiert dann zu der richtigen Ausrichtung von Templatebild zu Referenzbild.

3.2.2 Registrierungsmodell

In jeder Iteration wurde eine ‘‘Gruppenweise‘‘ Multilevel Registrierung durchgeführt. Dabei wurden die Low-rank Bilder der einzelnen Zeitpunkte auf den Durchschnitt aller Low-rank Bilder in der jeweiligen Iteration registriert. Als Ausgangspunkt wurde Level 4 gewählt und bis zur feinsten Auflösung auf Level 7 iteriert. Als Distanzmaß wurde das in 3.2.1 beschriebene Distanzmaß verwendet und als Regularisierer der krümmungsbasierte Regularisierer aus 2.7 verwendet. Damit lässt sich das Registrierungsproblem 3.12 in unserem Fall wie folgt formulieren.

Definition 3.2 (Registrierungsproblem).

Seien $T : \mathbb{R}^d \rightarrow \mathbb{R}$ die Low-rank Komponenten eines Bildes einer Zeitreihe von Bildern, $R : \mathbb{R}^d \rightarrow \mathbb{R}$ der Durchschnitt aller Low-rank Komponenten, $y : \mathbb{R}^d \rightarrow \mathbb{R}^d$ eine unbekannte Transformation und $d \in \mathbb{N}$. Dann wird durch

$$J(y) = D^{RC}[T(y), R] + \beta S^{curv}[y] \xrightarrow{y} \min \quad (3.12)$$

mit $\beta \in \mathbb{N}$ das in dieser Arbeit verwendete Registrierungsproblem definiert .

4

Implementierung und Ergebnisse

Im Folgenden werden die während der Anfertigung dieser Arbeit durchgeführten Tests vorgestellt. Zunächst wird auf die Experimente mit realen Daten aus einem dynamischen kontrastmittelbasierten MRT eingegangen. Anschließend werden die Ergebnisse für Tests mit einem künstlich erzeugtem Datensatz vorgestellt

4.1 Tests auf realen Daten

In diesem Abschnitt werden alle durchgeführten Experimente und Ergebnisse bei Verwendung von realen MRT Daten präsentiert. Zunächst werden die Tests der robusten PCA beschrieben bevor auf die Ergebnisse des Algorithmus von Hamy u. a. [13] eingegangen wird. Dabei wird auch genauer auf die Wahl der jeweiligen Parameter eingegangen.

Zunächst wurden Tests mit einem 4D Datensatz, welcher durch ein kontrastmittelbasiertes MRT erzeugt wurde, durchgeführt. Dieser besteht aus an 49 Zeitpunkten aufgenommenen 3D Bildern der Größe $146 \times 82 \times 52$ Voxel von der Niere, welche mit freundlicher Genehmigung von Jarle Rørvik vom Universitätskrankenhaus Haukeland in Bergen in Norwegen bereitgestellt wurde. Des Weiteren wurden die Algorithmen in Matlab (Version 9.6 R2019a) implementiert.

Zunächst wurde die robuste PCA getestet. Die Implementierung erfolgte Naiv nach dem in Abschnitt 2.3 vorgestellten Algorithmus. Da es sich bei dem in

Hamy beschriebenen Verfahren um einen 2D Ansatz handelt, wurde zunächst manuell eine Schicht, in der ein großer Querschnitt der Niere über alle Zeitpunkte zu sehen ist, aus dem Datensatz ausgewählt. Hierdurch wurden 49 2D Bilder der Größe 146×82 Pixel erzeugt. Die Matrix $M \in \mathbb{R}^{146 \times 82 \times 49}$ wurde anschließend spaltenweise mit den vektorisierten Bildern der einzelnen Zeitpunkte gefüllt.

Nach dem Durchlaufen des Algorithmus erhält man damit eine Matrix $L \in \mathbb{R}^{146 \times 82 \times 49}$ mit geringem Rang und eine dünn besetzte Matrix $S \in \mathbb{R}^{146 \times 82 \times 49}$. Per reshape können daraus wieder 49 Bilder der Größe 146×82 erhalten werden. Der Parameter $\mu = n_1 n_2 / 4 \|M\|$ mit $(n_1, n_2) = (146, 82)$ und die Abbruchbedingung

$$\|M - L - S\| \leq \delta \|M\|_F$$

mit $\delta = 10^{-7}$ wurden wie in [5] gewählt.

In Abbildung 4.1 sind die Originalbilder der Zeitpunkte 6, 20 bzw. 40 und korrespondierenden Matrizen mit geringem Rang bzw. die dünn besetzte Matrix nach durchlaufen der robusten PCA für das laut [5] optimale $\lambda_c = 1/\sqrt{\max(n_1, n_2)}$ gezeigt. Auffällig bei den Ergebnissen ist, dass neben den durch das Kontrastmittel hervorgerufenen hohen Intensitäten im Bereich der Niere auch Anteile der Bewegung in der Sparse Matrix S auftauchen. Des Weiteren sieht man, dass die Komponente mit geringem Rang wesentlich weniger Rauschen als die Originaldaten enthält. Auch dieser Anteil lässt sich in der Sparse Komponente wiederfinden.

Bei weiteren Tests, wurden für λ verschiedene Werte getestet. Dadurch konnte ein Eindruck über den Einfluss des Parameters λ gewonnen werden und ein geeigneter Startwert λ_{start} für den Algorithmus von Hamy 3.1 gefunden werden. In Abbildung 4.2 ist der Rang der Matrix M in Abhängigkeit von λ gezeigt.

Die Wahl von λ beeinflusst wie viel der Details bzw. der Bewegung in L zu sehen ist. In Abbildung 4.3 ist dies gut zu erkennen. Dort ist ein Profil des zeitlichen Verlaufes von L für verschiedene Lambda und der Originaldaten M (ganz links) zu sehen. Je größer λ gewählt wird, desto mehr Details werden

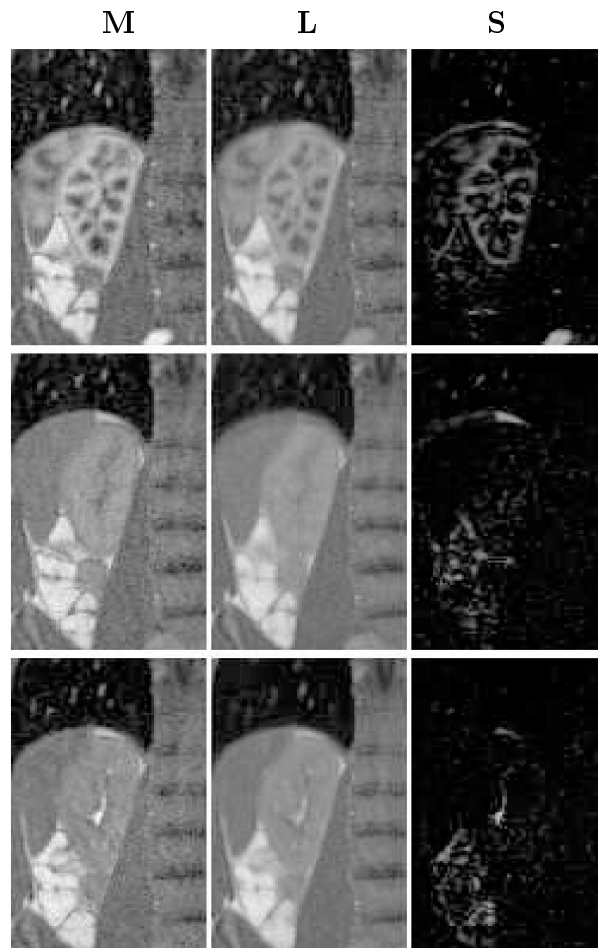


Abbildung 4.1. Ergebnisse der robusten PCA mit $\lambda = \lambda_c$. In den Zeilen sind jeweils das Originalbild (links), die Komponente mit geringem Rang L (Mitte) und der dünn besetzte Anteil S (rechts) für verschiedene Zeitpunkte zu sehen. In der ersten Zeile für Zeitpunkt 6, in der zweiten Zeile für Zeitpunkt 20 und in der dritten Zeile für Zeitpunkt 40.

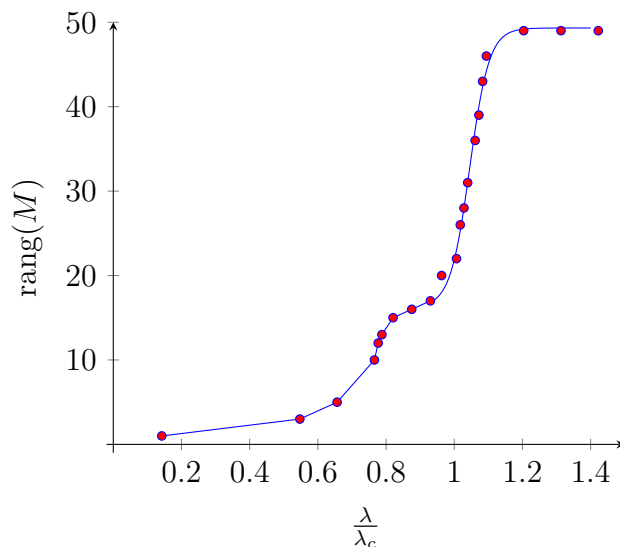


Abbildung 4.2. In dieser Abbildung ist der Plot des Rang der Matrix M in Abhängigkeit des Trade-Off-Parameters λ zu sehen. Dabei bezeichne λ_c das in [5] optimale λ .

in L sichtbar. Außerdem fällt auf, dass auch die Bewegungskomponente für wachsendes Lambda in L verschwindet.

Aus den Abbildungen 4.2 und 4.3 wurde empirisch λ_{start} so gewählt, dass der Rang der Matrix M ein viertel so groß ist wie die Anzahl an Zeitpunkten. Somit soll gewährleistet sein, dass ein Teil der Bewegung in L enthalten ist und der Großteil des Kontrastmittels in S steckt. Auf Grund des exponentiellen Verlaufes der Kurve in 4.2 wurde λ anschließend in jedem Iterationschritt logarithmisch erhöht. Dies soll dazu führen, dass die Menge an Details in L von Iteration zu Iteration nicht sprunghaft ansteigt. Die Gesamtzahl an Iterationen wurde wie in Hamy u. a. [13] vorgeschlagen auf 10 festgesetzt.

Bei den Experimenten des gesamten Algorithmus 3.1 wurden verschiedene Werte sowohl für den Regularisierungsparameter β im Distanzmaß 3.11 als auch für der Regularisierungsparameter für den Krümmungsbasierten Regularisierer getestet. In Abbildung 4.4 ist eine Profilansicht der besten Ergebnisse für den Algorithmus von [13] bei Anwendung auf die Daten der Niere gezeigt. Dabei wurde das α aus 3.11 als 200 festgelegt. Der Regularisierungs-

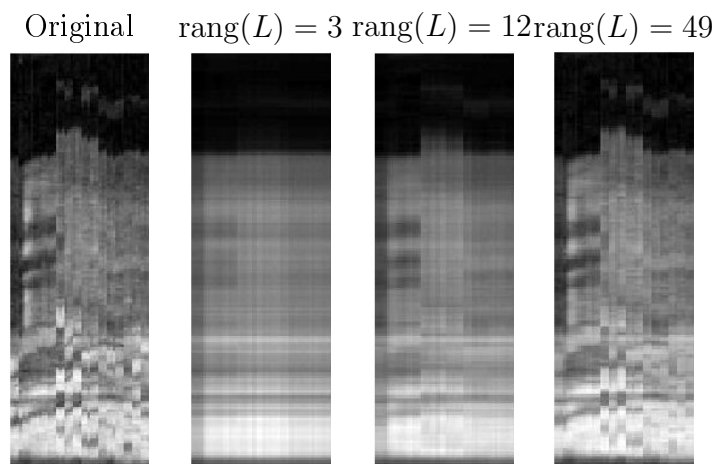


Abbildung 4.3. Zeitliches Profil der Matrizen mit geringem Rang für unterschiedliche Wahl von λ und eine fixe Spalte (28); Von links nach rechts sind die zunächst Originaldaten dann die Ergebnisse für $\lambda = 0.005$ bzw. $\lambda = 0.00071$ und $\lambda = 0.011$ zu sehen.

parameter β aus 3.12 wurde bei 8000 startend in jeder Iteration um jeweils 100 verringert. Die Idee dahinter ist, dass die Bewegung in der Low-rank Komponente, also in dem Teil der registriert werden soll, mit zunehmender Iteration schwächer wird.

Als optimales Ergebnis wird angenommen, dass horizontale Banden in der Profilansicht zu sehen sind. Ist dies für alle Profile der Fall, so ist im Zeitverlauf keine bzw. wenig Bewegung zwischen den einzelnen Zeitpunkten zu erkennen. Vergleicht man das Profil der Originaldaten mit den registrierten Daten, so fällt auf, dass eine Bewegungskorrektur stattgefunden hat, aber dennoch noch keine klaren Banden zu erkennen sind. Ein möglicher Grund könnte die noch nicht optimale Wahl der Parameter sein und könnte in weiteren Untersuchungen geprüft werden.

Als Erweiterung zu dem von Hamy u. a. [13] vorgestellten Algorithmus wurde der Ansatz auch für 3D Daten getestet. Dazu wurden nicht eine manuell ausgewählte Schicht sondern die gesamte 3D Bilder der Niere als Spaltenvektoren in der Matrix M gespeichert und als Input für den Algorithmus übergeben. Für die Registrierung wurde die in FAIR bereits implementierte Funktion

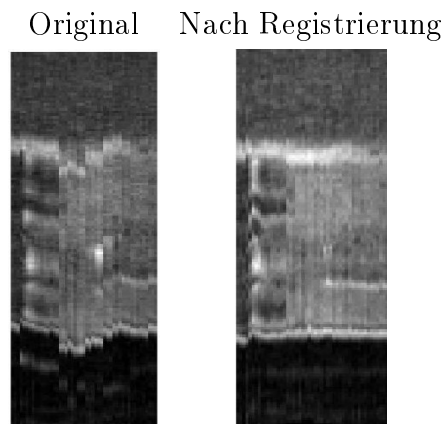


Abbildung 4.4. Ergebnisse des Algorithmus von Hamy u. a. [13] für Kontrastmittel basierte Daten der Niere. Dabei ist in beiden Bildern das Profil der Originalzeitreihe (links) und nach der Registrierung (rechts) zu sehen

MLIR [17], die 3D Daten verarbeiten kann, verwendet. Da die Laufzeit durch die erhöhte Anzahl an Pixeln bzw. Voxeln enorm steigt, konnten nur Tests von Level 3 bis 5, wobei Level 6 die feinste Auflösung ist, durchgeführt werden. In jeder Iteration müssen auf höchster Auflösung anstatt 49 2D Bildern der Größe 146×86 49 3D Bilder der Größe $146 \times 86 \times 56$ registriert werden. Das bedeutet, dass nicht 12556 sondern 703136 Pixel in jeder Registrierung miteinander verglichen werden müssen.

Des Weiteren wurde das $\lambda = \lambda_c$ konstant als das in [5] festgelegte λ gewählt und die Parameter für die Registrierung aus dem 2D Ansatz übernommen und nicht weiter angepasst. Da die Auflösung sehr gering ist, ist es schwer eine Bewertung der Ergebnisse vorzunehmen. Wird der Datensatz nach der Registrierung betrachtet lässt sich jedoch erahnen, dass eine Bewegungskorrektur erfolgt ist.

Um Vergleiche zu anderen Arbeiten ziehen zu können, wurde ein weiterer Datensatz der Niere von Jarle Rørvik einigen Tests unterzogen. Dieser besteht aus 45 3D Bildern einer rechten menschlichen Niere mit Größe $178 \times 95 \times 30$. Dieser Datensatz wurde von Brehmer u. a. [3] genutzt um die dort eingeführten Ansätze zur Registrierung von kontrastmittelbasierten MRT Daten

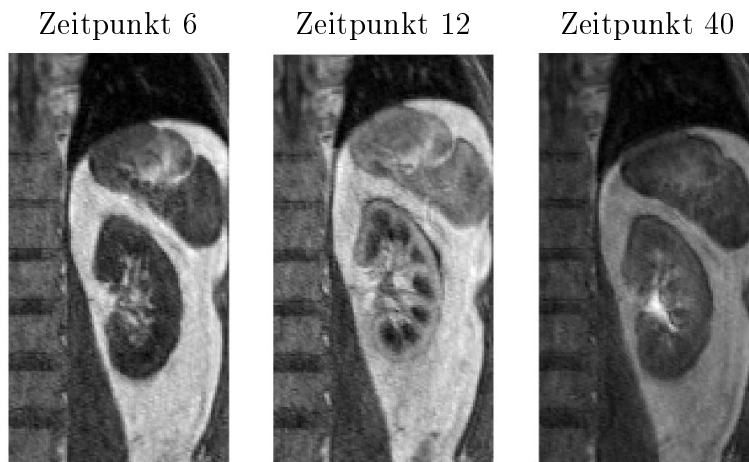


Abbildung 4.5. Drei repräsentative 2D Bilder des 4D Datensatzes der rechten menschlichen Niere. Der Datensatz hat die Größe $178 \times 95 \times 30 \times 45$ und die abgebildeten Schichten die Größe 178×95

durchzuführen. Analog zu Brehmer u. a. [3] wurden daraus dann 2D Bilder der Größe 178×95 aus der 18ten z Schicht generiert. In Abbildung 4.5 sind drei Zeitpunkte repräsentativ abgebildet.

Analog zum vorher vorgestellten Datensatz, wurden auch hier Tests für die robuste PCA durchgeführt, um einen geeigneten Startwert für λ zu finden. Die Parameter für die Registrierung wurden übernommen. In Abbildung 4.6 sind die Ergebnisse aus den verschiedenen Ansätzen zu sehen.

Den Ergebnissen für NGF liegt der sequentielle Ansatz aus [3] zu Grunde. Dabei wird eine Zeitreihe von Bildern $T = (\mathcal{T}_1, \dots, \mathcal{T}_K)$ nach folgender Zielfunktion registriert:

$$J^{seq}(Y, T) := \sum_{k=2}^K \{D^{NGF}[T_{k-1}(y_{k-1}), T(y_k)] + S(y_k)\}.$$

Dabei wird mit $Y = (y_1, \dots, y_K)$ das Array welches alle Transformationen enthält bezeichnet. Bei diesem Ansatz werden sequentiell je zwei Bilder der

Zeitreihe registriert. Dies geschieht bis eine Abbruchbedingung, z.B. eine maximale Anzahl an Iterationen erreicht ist, erfüllt ist. Ein Problem dieses Ansatzes ist, dass die y_k so gut wie nicht gekoppelt sind, so dass eine Änderung von y_k nur in jedem K -ten Schritt Auswirkungen hat. Dadurch ist eine hohe Anzahl an Iterationen und damit eine lange Rechenzeit notwendig, um ein akzeptables Ergebnis zu erzielen. Für die Ergebnisse dieses Ansatzes in 4.6 wurde nur einmal über die Zeitreihe iteriert. Es ist also noch eine Verbesserung hinsichtlich der erwarteten optimalen Lösung zu erwarten, wenn mehr Iterationen durchgeführt werden.

Für die Ergebnisse für SqN in Abbildung 4.6 wurde der globale Ansatz von Brehmer u. a. [3] genutzt. Bei diesem Ansatz wird kein explizites Referenzbild benötigt, sondern alle Bilder der Zeitreihe werden gleichzeitig registriert. Die zu minimierende Zielfunktion kann dabei wie folgt formuliert werden:

$$J^{glo} := D^{glo}[T(Y)] + S^{glo}[Y],$$

wobei $S^{glo}[Y] := \sum_{k=1}^K S(y_k)$ und S ein beliebiger Regularisierer aus Abschnitt 2.1 sei. Das globale Distanzmaß D^{glo} basiert dabei auf einem Array von Merkmalen. Die Bilder der Zeitreihe werden als registriert angesehen, wenn der Rang dieses Arrays minimal aber nicht Null ist. Mathematisch gesehen kann dies über eine Minimierung des sogenannten SqN Distanzmaßes, welches in [3] vorgestellt wird, erreicht werden. Für genauere Informationen über die Gestalt dieses Distanzmaßes wird auf Brehmer u. a. [3] verwiesen.

Beim Vergleich der Ergebnisse aus Abbildung 4.6 fällt auf, dass der im Zuge dieser Arbeit implementierte Algorithmus von Hamy u. a. [13] ähnliche Ergebnisse liefert wie der Ansatz von Brehmer u. a. [3] und damit bessere als der Ansatz mit NGF als Distanzmaß. Dabei sei nochmal erwähnt, dass bei sequentiellen Ansatz mit NGF einer Verbesserung der Ergebnisse erwartet wird, wenn die Anzahl an Iterationen erhöht wird. Vergleicht man die Ergebnisse für Position 58 miteinander, fällt auf, dass die horizontalen Kanten für den SqN Ansatz etwas klarer zu erkennen sind. Allerdings sind für unseren Ansatz deutlich mehr Banden zu erkennen. Dies könnte darauf hinweisen, dass mehr Details im Ansatz von Hamy u. a. [13] erhalten bleiben, während die

Ergebnisse von Brehmer u. a. [3] während der Registrierung geglättet wurden und so etwas Information verloren gegangen sein könnte.

Für Position 80 fällt auf, dass der dunkle Streifen am unteren Rand der Originalbilder beim Ansatz von Hamy u. a. [13] erhalten bleibt und beim Ansatz von Brehmer u. a. [3] verschwindet. Wenn davon ausgegangen wird, dass eine Deformation in die hier nicht zu sehende Dimension nicht dafür verantwortlich ist, sind die Ergebnisse für diese Position des in dieser Arbeit implementierten Ansatzes besser einzuschätzen, als die von Brehmer u. a. [3].

4.2 Tests auf künstlich erzeugten Daten

Im folgenden werden die Ergebnisse des Algorithmus von Hamy u. a. [13] bei Anwendung auf eine künstlich erzeugte Zeitreihe von Bildern präsentiert.

Für die Tests wurde ein Datensatz, der freundlicher Weise von Roland Haase zu Verfügung gestellt wurde, genutzt. Dieser Besteht aus 40 Bildern mit 200×200 Pixeln.

In Abbildung 4.7 sind zwei Bilder zur Repräsentation dargestellt. Das gefüllte Rechteck und der weiße Rahmen bleiben über die Zeit konstant. Nur die Ellipse, bei der die Orientierung des Querbalkens im Inneren in jedem Zeitschritt abwechselnd horizontal bzw. vertikal ist, bewegt sich im Verlauf entlang eines Halbkreises. In Abbildung 4.7 sind das erste und letzte Bilder der Zeitreihe zu sehen.

Als Ergebnis ist zu erwarten, dass die Zeitreihe nach der Registrierung aus zwei verschiedenen Bildern besteht. Bei beiden Bildern sollte die Bewegung der Ellipse nicht mehr zu sehen sein, während das eine Bild die Ellipse mit horizontalem Balken und das andere die Ellipse mit vertikalem Balken enthält. Dies ist damit zu Begründen, dass die Informationen in den Bildern bis auf die Richtung des Balkens in der Ellipse gleich sind.

Der Startwert für den Parameter λ der robusten PCA wurde als $\lambda_{start} = 0.0078$ gesetzt und wie bei den vorherigen Experimenten ermittelt. Für die

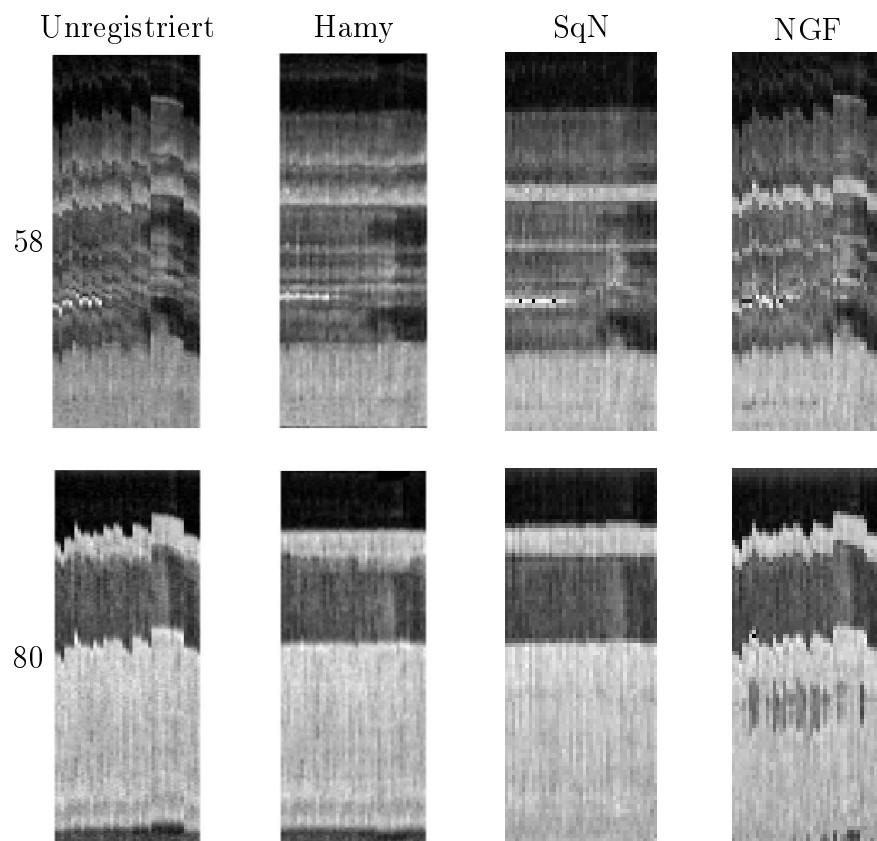


Abbildung 4.6. Ergebnisse der verschiedenen Ansätze zur Registrierung für Kontrastmittel basierte Daten der menschlichen Niere. Dabei ist in den Bildern das Profil der Originalzeitreihe nach der Registrierung nach Hamy u. a. [13], mit SqN und globalen Ansatz bzw. NGF und sequentiellen Ansatz [3] an den Positionen 58 (obere Zeile) und 80 (untere Zeile) zu sehen

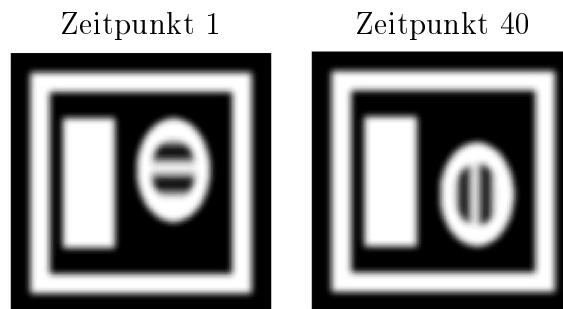


Abbildung 4.7. Zwei repräsentative Bilder aus der Zeitreihe der Künstlich erzeugten Daten. Links das erste Bild und rechts das letzte Bild der Zeitreihe. zwischen den Bildern bewegt sich die Ellipse entlang eines Halbkreises vom Startbild zum Endbild.

Parameter in der Registrierung erwiesen sich 0.05 für das α im Distanzmaß RC und 50 als Regularisierungsparameter als günstig. In Abbildung 4.8 sind die Ergebnisse für den künstlich erzeugten Datensatz mit diesen Parametern zu sehen. Dabei sei bemerkt, dass die gesamte Zeitreihe wie erwartet abwechselnd aus den beiden dort zu sehenden Bildern besteht. Es fällt auf, dass die Bewegung der Ellipse entlang des Halbkreises korrigiert wurde und auch die restlichen Elemente starr bleiben. Der Balken in der Mitte der Ellipse konnte jedoch nicht den Originaldaten entsprechend rekonstruiert werden. Es bleibt in der Mitte ein Kreis zurück. Ferner sind an den insgesamt vier Schnittpunkten von Ellipse und Balken Auswölbungen zu sehen und der Balken ist generell schmaler als in den Originaldaten. Dies liegt zum einen daran, dass in jeder Iteration auf den Durchschnitt aller Matrizen mit geringem Rang registriert wird. Jener enthält in jeder Iteration sowohl Informationen vom horizontalen, als auch vom vertikalen Balken. Außerdem ist es möglich, dass der krümmungsbasierte Regularisierer für diese Art von Daten nicht die optimale Wahl ist und dadurch innerhalb der Ellipse Rundungen an Punkten, an denen der Balken auf die Ellipse trifft, auftreten. Es liegt nahe, dass auch in diesem Setting die Parameter noch nicht optimal gewählt wurden. Hierzu müssten noch weitere Experimente durchgeführt werden.

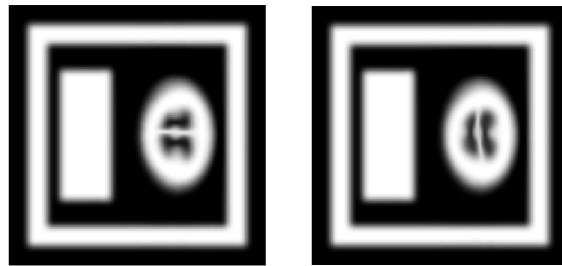


Abbildung 4.8. Ergebnisse des künstlich erzeugten Daten, nach durchlaufen des Algorithmus nach Hamy u. a. [13].

5

Fazit und Ausblick

In diesem Kapitel werden die in Abschnitt 4 vorgestellten Ergebnisse zusammengefasst und diskutiert. Zunächst werden die Ergebnisse der robusten PCA diskutiert. Anschließend wird der Algorithmus von Hamy u. a. [13] analysiert und diskutiert. Abschließend wird ein Fazit gezogen und ein Ausblick für zukünftige Arbeiten gegeben.

Die Bewegungskorrektur von Zeitreihen von MRT Bildern ist ein wichtiges Teilgebiet der Bildregistrierung. Im Zuge dieser Arbeit wurde ein Algorithmus von Hamy u. a. [13] implementiert und mit zwei anderen Ansätzen, die in [3] zu finden sind, verglichen. Dazu wurden 4D Daten aus einem kontrastmittelbasierten MRT der menschlichen Niere als Eingabebilder verwendet. Es konnte gezeigt werden, dass der Algorithmus von Hamy u. a. [13] für den betrachteten Datensatz ähnlich gute Ergebnisse liefert wie der globale Ansatz von Brehmer u. a. [3]. Des Weiteren wurde deutlich, dass der hier vorgestellte Algorithmus wesentlich bessere Ergebnisse liefert als der sequentielle Ansatz.

Weiterhin wurde durch die Anwendung auf dem künstlich erzeugtem Datensatz gezeigt, dass der Algorithmus von Hamy u. a. [13] plausible Ergebnisse für die Bewegungskorrektur liefert. Es wurden zwar die Balken in der Ellipse nicht exakt rekonstruiert, dennoch konnte die halbkreisförmige Bewegung der Ellipse komplett entfernt werden. Dabei sind die restlichen, starren Bildelemente unberührt geblieben. In weiteren Arbeiten ist zu überprüfen, ob möglicherweise die Verwendung eines anderen Regularisierungsmodells für diesen Anwendungsfall noch bessere Ergebnisse liefert.

Für zukünftige Arbeiten sind während der Anfertigung dieser Arbeit einige Fragen aufgekommen, die im Folgenden kurz zusammengefasst werden.

Beim Ansatz einer PCA stellt sich die Frage, ob eine solche Zerlegung der Daten in orthogonale Komponenten überhaupt Sinn macht oder ob eine andere Art der Repräsentation vielleicht sinnvoller wäre. Dennoch kann über den Parameter λ gut kontrolliert werden, wie stark die Daten in Low-rank und Sparse Anteil zerlegt werden.

Außerdem stellt sich die Frage, ob die robuste PCA die Daten wirklich nur in die Bewegungskomponente und den Kontrastmittel basierten Anteil aufspaltet oder ob noch weitere Komponenten verändert werden. Angenommen, es werden nur diese beiden Anteile der Daten berührt, so ist in den Ergebnissen zu erkennen, dass je nach Wahl von λ die Aufteilung von Bewegung und Kontrastmittel in die Low-rank und Sparse Komponente unterschiedlich stark ist. Auffällig ist auch, dass Bewegung und Kontrastmittel nie zu hundert Prozent separiert werden. Dabei kommt die Frage auf, ob ein Ansatz möglich ist, bei welchem dies der Fall ist. Mathematisch gesprochen könnte eine Zerlegung von

$$M = L + B + K$$

möglich sein, wobei L eine Low-rank Matrix und B und K Sparsematrizen die die Bewegungskomponente bzw. den Anteil vom Kontrastmittel enthalten. So wäre zu überprüfen, ob das Lösen des Minimierungsproblems

$$\begin{aligned} \min \quad & \|L\|_* + \alpha\|B\|_1 + \beta\|K\|_1 \\ \text{u.d.N.d} \quad & L + K + B = M \end{aligned} \tag{5.1}$$

mit zusätzlichen Anforderungen an K und B eventuell die gewünschten Ergebnisse erzielen könnte. Dann könnte man aus den Werten in B ein Deformationsfeld berechnen und sich eine Registrierung eventuell sparen.

Ein großer Punkt beim Implementieren, war das Finden geeigneter Parameter. Zukünftig könnte untersucht werden, ob die Parameter aus dem Distanzmaß und dem Regularisierer voneinander abhängig sind, oder ob sie generell weiter optimiert werden können. Ein möglicher Ansatz könnte sein, zu versuchen die Parameter durch ein Neuronales Netz zu lernen und so zu optimieren.

Des Weiteren ist die Anzahl an Iterationen N_{iter} noch nicht weiter untersucht worden. Aus diesem Grund könnte man auch hier weitere Testläufe mit weniger oder mehr als 10 Iterationen durchführen, um zu überprüfen, ob der dadurch reduzierte bzw. erhöhte Rechenaufwand die Ergebnisse nennenswert verschlechtert bzw. verbessert.

Auch für den 3D Ansatz können weitere Experimente durchgeführt werden. Zum einen kann auch hier ein variables λ in der robusten PCA eingeführt werden und zum anderen können die Parameter besser angepasst werden. Darüber hinaus könnte ein anderer Optimierungsalgorithmus die Rechenzeit verbessern und so zu interpretierbaren Ergebnissen auf höheren Auflösungen führen.

Im Zuge der Registrierung wurde Residual Komplexity als Distanzmaß verwendet. Dabei wird die Annahme getroffen, dass das Rauschen auf den Daten als möglichst unkorreliert anzunehmen ist. Hier könnten andere Rauschmodelle getestet werden, da das Rauschen von vielen Faktoren abhängig ist. So haben nicht nur die Modalität, sondern auch die Wahl des Gerätes und Störfaktoren aus der Umgebung einen Einfluss auf das Rauschen.

Beim implementierten Algorithmus wird ein Distanzmaß mit Allgemeiner Form

$$D[T, R] = r^T A r$$

genutzt (vgl. Gleichung 3.6). Dabei enthält A auf dem gesamten Bildgebiet Werte. Es werden folglich alle Pixel unterschiedlich stark gewichtet. Eine mögliche Erweiterung könnte sein, durch einfließen von a priori Wissen, die Matrix A so anzupassen, dass sich auf Teile des Bildes konzentriert wird in denen Bewegung erwartet wird. So könnte man Bildbereiche in denen z.B. die Wirbelsäule zu sehen ist durch einfügen einer weiteren Matrix W gar nicht oder weniger gewichten als z.B. den Bereich in denen die Niere zu sehen ist. Des Weiteren könnten hier Ansätze wie z.B. der von König u. a. [15] genutzt werden um die rigiden Eigenschaften von einigen Strukturen sicherzustellen.

Literaturverzeichnis

- [1] Bertsekas Dimitri P: *Constrained optimization and Lagrange multiplier methods*. Academic Press. Cambridge (Massachusetts), 2014.
- [2] Beuthien Björn: *Restringierte Medizinische Bildregistrierung*. Diss. Universität zu Lübeck. Lübeck, 2014.
- [3] Brehmer Kai, Aggrawal Hari Om, Heldmann Stefan, Modersitzki Jan: *Variational registration of multiple images with the SVD based SqN distance measure*. SSVM. Hofgeismar, Deutschland, 2019.
- [4] Broit Chaim: *Optimal Registration of Deformed Images*. Diss. University of Pennsylvania. Philadelphia, USA, 1981.
- [5] Candès Emmanuel J, Li Xiaodong, Ma Yi, Wright John: *Robust principal component analysis?* *JACM*. Bd. 58. 3. New York City: ACM, 2011.
- [6] Dantzig George Bernard: *Linear programming and extensions*. Princeton (New Jersey): Princeton University Press, 1998.
- [7] Fischer Bernd, Modersitzki Jan: *Curvature based image registration*. *JMIV*. Bd. 18. 1. New York City: Springer, 2003, S. 81–85.
- [8] Fischer Bernd, Modersitzki Jan: *Curvature based registration with applications to MR-mammography*. ICCS. New York City: Springer, 2002, S. 202–206.
- [9] Fischer Bernd, Modersitzki Jan: *Fast diffusion registration*. *Contemporary Mathematics*. Bd. 313. Singapur, 2002, S. 117–128.
- [10] Guyader Jean-Marie, Huizinga Wyke, Fortunati Valerio, Poot Dirk HJ, Kranenburg Matthijs van, Veenland Jifke F, Paulides Margarethus M, Niessen Wiro J, Klein Stefan: *Total correlation-based groupwise image registration for quantitative MRI*. CVPR. 2016, S. 186–193.
- [11] Haber Eldad, Modersitzki Jan: *Intensity gradient based registration and fusion of multi-modal images*. MICCAI. Kopenhagen, Dänemark: Springer, 2006, S. 726–733.

-
- [12] Hadamard Jacques: *Sur les problèmes aux dérivées partielles et leur signification physique. Princeton University Bulletin*. Princeton, New Jersey, 1902, S. 49–52.
- [13] Hamy Valentin, Dikaios Nikolaos, Punwani Shonit, Melbourne Andrew, Latifoltojar Arash, Makanyanga Jessica, Chouhan Manil, Helbren Emma, Menys Alex, Taylor Stuart u. a.: *Respiratory motion correction in dynamic MRI using robust data decomposition registration—Application to DCE-MRI. Medical image analysis*. Bd. 18. 2. Amsterdam: Elsevier, 2014, S. 301–313.
- [14] Horn Berthold KP, Schunck Brian G: *Determining optical flow. Artificial intelligence*. Bd. 17. 1-3. Amsterdam: Elsevier, 1981, S. 185–203.
- [15] König Lars, Derksen Alexander, Papenberg Nils, Haas Benjamin: *Deformable image registration for adaptive radiotherapy with guaranteed local rigidity constraints. Radiation Oncology*. Bd. 11. 1. London: Bio-Med Central, 2016, S. 122.
- [16] Lin Zhouchen, Chen Minming, Ma Yi: *The augmented lagrange multiplier method for exact recovery of corrupted low-rank matrices. arXiv*. 2010.
- [17] Modersitzki Jan: *FAIR: flexible algorithms for image registration*. Philadelphia, Pennsylvania: SIAM, 2009.
- [18] Modersitzki Jan: *Numerical methods for image registration*. Oxford: Oxford University Press, 2004.
- [19] Myronenko Andriy, Song Xubo: *Intensity-based image registration by minimizing residual complexity. IEEE TMI*. Bd. 29. 11. Urbana (Illinois), 2010, S. 1882–1891.
- [20] Nocedal Jorge, Wright Stephen: *Numerical optimization*. Luxemburg: Springer, 2006.
- [21] Olesch Janine: *Bildregistrierung für die navigierte Chirurgie: Spezialisierte Ansätze zur Anwendung in der navigierten Leberchirurgie*. Wiesbaden: Springer, 2014.

- [22] Seghers Dieter, D'Agostino Emiliano, Maes Frederik, Vandermeulen Dirk, Suetens Paul: *Construction of a brain template from MR images using state-of-the-art registration and segmentation techniques*. MICCAI. Saint-Malo, Brittany, 2004, S. 696–703.
- [23] Yuan Xiaoming, Yang Junfeng: *Sparse and low-rank matrix decomposition via alternating direction methods*. *Optimization Online*. 2009.