



UNIVERSITÄT ZU LÜBECK
INSTITUTE OF MATHEMATICS AND
IMAGE COMPUTING

Masterarbeit

Bildregistrierung unter Verwendung von Lie-Gruppen

Veronika Zimmer

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

Abgabe: 24.06.2011



Erklärung

Ich versichere hiermit, dass ich die vorliegende Arbeit selbstständig und lediglich unter Benutzung der angegebenen Literatur und Hilfsmittel angefertigt habe.

Ich erkläre weiterhin, dass die vorliegende Arbeit noch nicht im Rahmen eines anderen Prüfungsverfahrens eingereicht wurde.

Ort, Datum

Unterschrift

Inhaltsverzeichnis

1	Einleitung	1
2	Der Dämonen-Algorithmus von Thirion	5
2.1	Grundlagen der Bildregistrierung	6
2.2	Varianten des Dämonen-Algorithmus von Thirion	9
3	Einführung in die Differentialgeometrie	15
3.1	Differenzierbare Mannigfaltigkeiten	16
3.2	Tangentialvektoren, Vektorfelder und Differentiale	20
4	Einführung in die Theorie der Lie-Gruppen	29
4.1	Lie-Gruppen und Lie-Algebren	30
4.2	Die Exponentialabbildung	37
5	Optimierung auf Lie-Gruppen zur Bildregistrierung	43
5.1	Optimierung auf Lie-Gruppen	44
5.2	Das Gauß-Newton-Verfahren auf Lie-Gruppen zur Bildregistrierung	46
6	Optimierung auf Lie-Gruppen und der Dämonen-Algorithmus	55
6.1	Der Dämonen-Algorithmus als ein Optimierungsverfahren	56
6.2	Eine diffeomorphe Erweiterung des Dämonen-Algorithmus	61

7	Ergebnisse	65
7.1	ITK-Implementierung des Dämonen-Algorithmus	65
7.2	Ergebnisse einer Bildregistrierung mit dem Dämonen-Algorithmus	67
8	Zusammenfassung und Diskussion	71
8.1	Zusammenfassung	71
8.2	Diskussion und Ausblick	73
A	Anhang	77
A.1	Nomenklatur	77
A.2	Berechnungen	79
	Literaturverzeichnis	81

Kapitel 1

Einleitung

Die Bildregistrierung ist ein Bereich der Bildverarbeitung, der gerade in der Medizin von großer Bedeutung ist [38, 30, 7, 13]. Gegeben sind zwei Bilder, die zu verschiedenen Zeiten, von verschiedenen bildgebenden Verfahren und/oder aus verschiedenen Perspektiven aufgenommen wurden. Das Ziel der Bildregistrierung ist, eine Transformation zu finden, die eines der beiden Bilder transformiert, so dass sich die Bilder möglichst ähnlich werden [33, 15]. Oft wird das Registrierungsproblem als ein Optimierungsproblem formuliert, in dem über anwendungsspezifische Zielfunktionen bezüglich einer Transformation minimiert wird. In der parametrischen Registrierung wird eine Zielfunktion auf einer endlich dimensional Menge von Transformationen minimiert, wobei die Transformationen durch Linearkombinationen von Basisfunktionen dargestellt werden [33, 30, 13]. In der nicht-parametrischen Registrierung wird auf dem Raum aller Transformationen minimiert. Ein sogenannter Regularisierer misst deren Plausibilität [33]. Die Optimierungsverfahren, die für die parametrische und nicht-parametrische Registrierung verwendet werden, optimieren auf Vektorräumen.

In dieser Arbeit wird ein anderer Ansatz zur Registrierung verfolgt. Es wird eine Bildregistrierungsmethode unter Verwendung von Lie-Gruppen beschrieben. Lie-Gruppen sind differenzierbare Mannigfaltigkeiten, die zusätzlich die Struktur einer Gruppe aufweisen [26]. Einige Klassen von Transformationen, z.B. die rigiden oder affinen Transformationen, bilden keine Vektorräume, da sie unter der Addition nicht abgeschlossen sind. Sie bilden Lie-Gruppen, was bedeutet, dass die Verknüpfung und Invertierung derartiger Transformationen wieder zu Transformationen der gleichen Klasse führt. Die Idee des hier beschriebenen Verfahrens ist,

eine Zielfunktion auf einer Lie-Gruppe zu optimieren. Vektorräume sind triviale Lie-Gruppen, so dass die Optimierungsverfahren auf Lie-Gruppen eine Verallgemeinerung der Optimierungsverfahren auf Vektorräumen darstellen.

Die Motivation, sich mit Bildregistrierungsmethoden basierend auf Lie-Gruppen zu beschäftigen, war der Dämonen-Algorithmus von Jean-Philippe Thirion [44]. Dieser ist ein sehr effizienter Algorithmus für die Bildregistrierung, der in seiner ursprünglichen Formulierung jedoch kein Optimierungsproblem löst, sondern auf einem Diffusionsmodell beruht. Der Algorithmus ist ein Iterationsverfahren, das in jeder Iteration ein Verrückungsfeld durch sogenannte Dämonenkräfte berechnet und diese mit einer Gauß-Funktion glättet [44]. Wegen seiner Effizienz und einfachen Implementierung hat er in der Praxis viel Anwendung gefunden [35, 47, 53]. Da er jedoch auf keinem Optimierungsproblem oder sonstigem mathematischen Modell beruht, wurde dieser Algorithmus nicht vollständig verstanden.

In [49] wurde schließlich ein Registrierungsrahmen basierend auf der Optimierung auf Lie-Gruppen vorgestellt, mit dem es möglich ist, den Dämonen-Algorithmus als ein Lösungsverfahren für die Minimierung einer geeigneten Zielfunktion zu interpretieren. Der Raum der Transformationen, der im Dämonen-Algorithmus betrachtet wird, ist ein Vektorraum, also eine triviale Lie-Gruppe. Die Optimierung einer bestimmten Zielfunktion auf der trivialen Lie-Gruppe führt zur Berechnung der Dämonenkräfte.

In [49] werden die Optimierungsverfahren auf Lie-Gruppen kurz vorgestellt, es wird jedoch nicht auf die zu Grunde liegende Theorie eingegangen. Der Hauptteil dieser Arbeit besteht deswegen darin, mit ausgesuchten Begriffen aus der Differentialgeometrie und Lie-Gruppen-Theorie die Optimierung auf Lie-Gruppen zu beschreiben. Mit deren Hilfe kann anschließend die Bildregistrierung unter Verwendung von Lie-Gruppen vorgestellt werden.

Der Dämonen-Algorithmus nutzt, wie oben schon angedeutet, die Lie-Gruppen-Struktur nicht aus, da auf einem Vektorraum optimiert wird. In [51] wurde eine Erweiterung des Dämonen-Algorithmus vorgestellt, der diffeomorphe Dämonen-Algorithmus, der statt auf Vektorräumen auf der Gruppe der Diffeomorphismen optimiert.

Diffeomorphismen sind in der Bildregistrierung von großer Bedeutung, da sie invertierbar und glatt (stetig differenzierbar) sind, was zu plausibel deformierten Bildern führt [51, 3, 12].

Ein weiteres Ziel dieser Arbeit ist, die Verbindung der Bildregistrierungsmethode unter Verwendung von Lie-Gruppen zum Dämonen-Algorithmus und zu seiner diffeomorphen Erweiterung

rung herzustellen.

Die Arbeit gliedert sich folgendermaßen: In Kapitel 2 wird der Dämonen-Algorithmus von Thirion in zwei Varianten, der additiven und der kompositiven, vorgestellt. Zusätzlich werden zwei Dämonenkräfte präsentiert.

In Kapitel 3 werden differenzierbare Mannigfaltigkeiten und Strukturen auf ihnen und in Kapitel 4 ausgesuchte Begriffe der Lie-Gruppen-Theorie eingeführt. Mit diesen sehr technischen Begriffen wird in Kapitel 5 ein Optimierungsverfahren auf Lie-Gruppen, das Gauß-Newton-Verfahren auf Lie-Gruppen und seine Anwendung auf die Bildregistrierung beschrieben. Zur Verdeutlichung wird dies beispielhaft an der Lie-Gruppe der rigiden Transformationen erläutert.

In Kapitel 6 wird gezeigt, wie die Minimierung einer Zielfunktion mit dem Gauß-Newton-Verfahren auf Lie-Gruppen zu einer Berechnung der Dämonenkräfte führt. Des Weiteren wird die Idee des diffeomorphen Dämonen-Algorithmus erläutert.

In Kapitel 7 werden einige Registrierungsergebnisse des Dämonen-Algorithmus mit seinen unterschiedlichen Varianten (additiv, kompositiv und diffeomorph) kombiniert mit verschiedenen Dämonenkräften präsentiert, die mit der ITK-Implementierung [50] (ITK: Insight Segmentation and Registration Toolkit, [23]) erzielt wurden.

In Kapitel 8 folgt schließlich eine Zusammenfassung und Diskussion über noch offene Fragen und ein Ausblick.

Kapitel 2

Der Dämonen-Algorithmus von Thirion

Ein sehr effizienter Algorithmus für die nicht-parametrischen Bildregistrierung ist der Dämonen-Algorithmus von Jean-Philippe Thirion. Er ist jedoch nicht wie die meisten Registrierungsverfahren zur Lösung eines Optimierungsproblems entwickelt worden, sondern beruht auf einem Diffusionsmodell [44]. Die nicht-parametrische Registrierung wurde in [44, 43] als ein Diffusionsprozess interpretiert und sogenannte Dämonen verschieben die Bildpunkte des Templatebildes entsprechend der sogenannten Dämonenkräfte. Der Algorithmus ist ein iteratives Verfahren, das in jedem Iterationsschritt ein Verrückungsfeld durch die Dämonenkräfte berechnet und dies mit der aktuellen Iterierten verknüpft. Zur Regularisierung wird das Verrückungsfeld und/oder die Transformation durch eine Gauß-Funktion geglättet [39, 44]. Die Dämonenkräfte werden für jeden Bildpunkt separat berechnet, woraus sich ein Verrückungsvektor ergibt. Aus den Verrückungsvektoren für alle Punkte eines Bildes, bestimmt sich die Verrückung des gesamten Bildes.

In der Literatur sind verschiedene Varianten des Dämonen-Algorithmus bekannt [35, 53, 44, 43], die sich zum einen in der Berechnung der Dämonenkräfte und zum anderen in der Verknüpfung des Verrückungsfeldes mit der aktuellen Iterierten unterscheiden. In dieser Arbeit werden zwei Dämonenkräfte betrachtet, die ursprünglich von Thirion eingeführt [44], im Folgenden klassische Kräfte genannt, und eine Symmetrisierung dieser Kräfte, die symmetrischen Dämonenkräfte [35, 43]. Dabei werden zwei unterschiedliche Verknüpfungen dieser Kräfte mit

der aktuellen Iterierten betrachtet. Die additive Update-Regel verknüpft das Verrückungsfeld und die Transformation durch Addition (additiver Dämonen-Algorithmus, [49, 39]), die kompositive Update-Regel verknüpft das Verrückungsfeld und die Transformation durch eine Komposition (kompositiver Dämonen-Algorithmus, [51, 42]).

Bevor im zweiten Teil des Kapitels der Dämonen-Algorithmus mit seinen unterschiedlichen Varianten vorgestellt wird, werden einige grundlegende Konzepte der Bildregistrierung beschrieben. Wichtig ist dabei vor allem die Darstellung von Bildern und deren Ableitungen sowie die Darstellung von Transformationen.

2.1 Grundlagen der Bildregistrierung

In diesem Abschnitt werden einige Konzepte der Bildregistrierung kurz vorgestellt. Weitaus detaillierter sind sie beschrieben in [33, 15, 38, 14, 30, 13].

Das Ziel der Registrierung zweier Bilder ist es, eine plausible Transformation zu finden, die das eine Bild so transformiert, dass es dem anderen möglichst ähnlich wird. Bilder werden in der Bildregistrierung typischerweise als kontinuierliche, differenzierbare Abbildungen von einem Gebiet $\Omega \subset \mathbb{R}^d$ in die reellen Zahlen angesehen [33, 38]. Der Parameter d steht für die Bilddimension und üblicherweise ist $d \in \{2, 3\}$. Ein Bild ordnet jedem $x \in \Omega$ einen Grauwert zu. Betrachtet werden zwei Bilder $\mathcal{R}, \mathcal{T} : \Omega \rightarrow \mathbb{R}$, wobei \mathcal{R} Referenz- und \mathcal{T} Templatebild genannt wird. Zusätzlich wird angenommen, dass die Bilder einen kompakten Träger haben, also $\mathcal{R}(x) = \mathcal{T}(x) = 0$ für $x \notin \Omega$ [33]. Für die Registrierung soll das Templatebild transformiert werden, so dass es dem Referenzbild möglichst ähnlich wird.

Ein Bild wird transformiert, indem das Gebiet Ω transformiert wird. Eine Transformation y ist eine differenzierbare Abbildung $y : \mathbb{R}^d \rightarrow \mathbb{R}^d$ mit $y(x) = x + u(x)$. Hierbei ist $x \in \mathbb{R}^d$ und $u : \mathbb{R}^d \rightarrow \mathbb{R}^d$ eine sogenannte Verrückung. Die Abbildungen y und u sind Vektorfelder, d.h. sie ordnen jedem $x \in \mathbb{R}^d$ einen Vektor $y(x) \in \mathbb{R}^d$ bzw. $u(x) \in \mathbb{R}^d$ zu. Das transformierte Templatebild ist damit \mathcal{T}_y mit $\mathcal{T}_y(x) = \mathcal{T} \circ y(x) = \mathcal{T}(x + u(x))$.

Distanzmaße und Regularisierer Zwei Bilder werden als ähnlich angesehen, wenn korrespondierende Punkte in den Bildern auch räumlich nahe beieinander liegen. Korrespondenz zweier Punkte in zwei medizinischen Bildern bedeutet, dass die beiden Punkte denselben anatomischen Ort repräsentieren. Gemessen wird die Ähnlichkeit zweier Bilder über sogenannte

Distanzmaße, die einem Referenzbild \mathcal{R} und einem transformierten Templatebild \mathcal{T}_y eine reelle Zahl zuordnen. Je kleiner diese Zahl ist, desto ähnlicher sind sich die Bilder. Für detaillierte Angaben zu unterschiedlichen Distanzmaßen siehe z.B. [33, 15, 19, 52].

Das Registrierungsproblem wird als Optimierungsproblem formuliert, indem ein zuvor gewähltes Distanzmaß bezüglich einer Transformation minimiert wird. Es existieren im Allgemeinen mehrere Lösungen, die zu gleichen transformierten Templatebildern führen. Daher ist das Registrierungsproblem ein schlecht gestelltes Problem [33], die gesuchte Transformation ist nicht eindeutig.

Die Eindeutigkeit der gesuchten Transformation wird in der parametrischen Registrierung durch eine Einschränkung des Suchraumes erreicht (durch eine Parametrisierung von Transformationen) [33, 38, 30, 13] und in der nicht-parametrischen Registrierung durch die Einführung eines sogenannten Regularisierers [32], welcher die Qualität von Transformationen misst, z.B. in Bezug auf Differenzierbarkeitseigenschaften. In der Literatur finden sich unterschiedliche Regularisierer. Beispiele sind der elastische [8] oder der diffusive Regularisierer [14]. In [14] wurde gezeigt, dass die Glättung der Transformation mit einer Gauß-Funktion eine Approximation an eine diffusive Regularisierung darstellt. Dieses bietet einen ersten Zusammenhang von typischen Registrierungsverfahren zu dem Dämonen-Algorithmus von Thirion, in dem die Verrückung durch Gauß-Funktionen geglättet wird, [11, 32, 14].

Damit ist die Funktion, die in einem Registrierungsproblem mit gegebenen Bildern \mathcal{R} und \mathcal{T} optimiert werden soll, das Funktional \mathcal{J} mit

$$\mathcal{J}(y) = \mathcal{D}(\mathcal{T}_y, \mathcal{R}) + \alpha \mathcal{S}(y), \quad (2.1)$$

wobei \mathcal{D} ein Distanzmaß ist, \mathcal{S} ein Regularisierer und $\alpha \in \mathbb{R}^+$ ein Regularisierungsparameter, der den Einfluss von \mathcal{S} auf die Lösung steuert [33, 38, 32].

Interpolation und Lösung des Registrierungsproblems Das Registrierungsproblem ist kontinuierlich formuliert worden. Die Eingabe eines Registrierungsverfahrens sind jedoch diskrete Bilder, die nur an einer endlichen Menge von Punkten $x \in \Omega \subset \mathbb{R}^d$ vorliegen. Die kontinuierliche Darstellung beschreibt das Registrierungsproblem jedoch besser als eine diskrete Darstellung. Für eine genauere Erläuterung der Gründe siehe [33, 38, 32].

Kontinuierliche Bilder können durch geeignete Interpolationsmethoden, wie die lineare Interpolation, die Spline-Interpolation, Polynom-Interpolation oder Nearest-Neighbor-Interpolation

[41, 6], aus den diskreten Eingabedaten berechnet werden. Sie können an beliebigen Punkten ausgewertet werden und ihre Ableitungen können bestimmt werden.

Für die genaue Beschreibung einer Diskretisierung des Registrierungsproblems, der Distanzmaße und der Regularisierer wird auf [33, 38, 32] verwiesen.

Diskrete Bilder und Ableitungen von Bildern Durch Interpolation der gegebenen Eingangsdaten des Templatebildes erhält man eine kontinuierliche Funktion $\mathcal{T} : \Omega \subset \mathbb{R}^d \rightarrow \mathbb{R}$. Seien jetzt $x^1, \dots, x^m \in \Omega$, $m < \infty$, $x^i = (x_1^i, \dots, x_d^i) \in \mathbb{R}^d$ die Punkte, an denen \mathcal{T} ausgewertet werden soll. Ein diskretes Bild T ist dann gegeben durch

$$T = (\mathcal{T}(x^1), \dots, \mathcal{T}(x^m))^T \in \mathbb{R}^m.$$

Sei $y : \mathbb{R}^d \rightarrow \mathbb{R}^d$ eine kontinuierliche Transformationen. Eine diskrete Transformation y ist gegeben durch

$$y = \begin{pmatrix} y(x^1) \\ \vdots \\ y(x^m) \end{pmatrix} \in \mathbb{R}^{m \times d}$$

und ein diskretes, mit y transformiertes Bild $T(y)$ ist

$$T(y) = (\mathcal{T}(y(x^1)), \dots, \mathcal{T}(y(x^m)))^T \in \mathbb{R}^m.$$

Da \mathcal{T} bei geeigneter Wahl der Interpolationsmethode differenzierbar ist, können die Ableitungen $d\mathcal{T}, d\mathcal{T}_y : \mathbb{R}^d \rightarrow \mathbb{R}^d$

$$d\mathcal{T} = (\partial_1 \mathcal{T}, \dots, \partial_d \mathcal{T}),$$

$$d\mathcal{T}_y = d\mathcal{T} \circ y = d\mathcal{T} \cdot dy \quad \text{mit} \quad dy = \left(\frac{dy_i}{dx_k} \right)_{j,k=1,\dots,d}$$

und die Gradienten

$$\nabla \mathcal{T} := d^T \mathcal{T},$$

$$\nabla \mathcal{T}_y = \nabla \mathcal{T} \circ y = d^T \mathcal{T}_y = d^T y \cdot d^T \mathcal{T}$$

bestimmt werden. Die Gradienten des Referenz- und des Templatebildes werden im Dämonen-Algorithmus benutzt, um die Dämonenkräfte zu berechnen.

2.2 Varianten des Dämonen-Algorithmus von Thirion

Der Dämonen-Algorithmus von Thirion ist ein Verfahren, um das Templatebild T auf das Referenzbild R zu registrieren. Folgende Arbeiten wurden für diesen Abschnitt verwendet: [51, 47, 42, 39, 44, 43].

Der Dämonen-Algorithmus beruht auf einem Diffusionsmodell, welches von einem Gedankenexperiment des Physikers James Clerk Maxwell (veröffentlicht 1871) abgeleitet wurde. Das Experiment illustriert ein scheinbares Paradoxon in der Thermodynamik [5]. Maxwell stellte sich vor, dass ein Behälter mit einem Gas gefüllt ist, das aus zwei Arten von Molekülen besteht. Der Behälter wird durch eine semipermeable Membran getrennt auf der sogenannte Dämonen sitzen. Diese können zwischen den Molekülen unterscheiden und erlauben nur die Diffusion von Molekülen der ersten Art in die eine Hälfte des Behälters und von Molekülen der zweiten Art in die andere Hälfte. Schließlich trennt die semipermeable Membran die zwei Molekülarten (vgl. Abb. 2.2 oben links), so dass die Entropie des Systems anscheinend verringert wurde. Dieses stellt aber einen Widerspruch zum zweiten Hauptsatz der Thermodynamik dar, der besagt, dass bei einer freiwilligen Zustandsänderung eines abgeschlossenen Systems die Entropie zunimmt [5]. Die Auflösung des scheinbaren Paradoxons ist, dass jeder Dämon bei der Zuordnung der Moleküle Energie verbraucht. Dies erhöht wieder die globale Entropie [5]. Durch dieses Gedankenexperiment kam der Dämonen-Algorithmus zu seinem Namen.

Dämonen in der Bildregistrierung In der Bildregistrierung entsprechen die Moleküle den Punkten in dem Templatebild T . Das Bild T wird als deformierbar angenommen, während das Referenzbild R unverändert bleibt. Die Dämonen sind die Punkte im Bild R und jedem Dämon $x \in \Omega$ wird ein Kraftvektor $f(x) \in \mathbb{R}^d$ zugeordnet, durch den die Punkte von T verschoben werden (vgl. Abb. 2.2 oben rechts). Die Verschiebung soll kontrolliert verlaufen, so dass letztendlich die Punkte in R und T zueinander korrespondieren. Die Richtung und die Stärke der Verschiebung wird über den Kraftvektor f bestimmt (vgl. Abb. 2.2 unten).

Aus den Kräften f jedes Dämons wird ein Verrückungsfeld $u : \mathbb{R}^d \rightarrow \mathbb{R}^d$ berechnet, welches mit einer Gauß-Funktion gefaltet und dadurch geglättet wird. Die d -dimensionale Gauß-Funktion mit Mittelwert μ und einer einheitlichen Standardabweichung σ für jede Dimension ist die

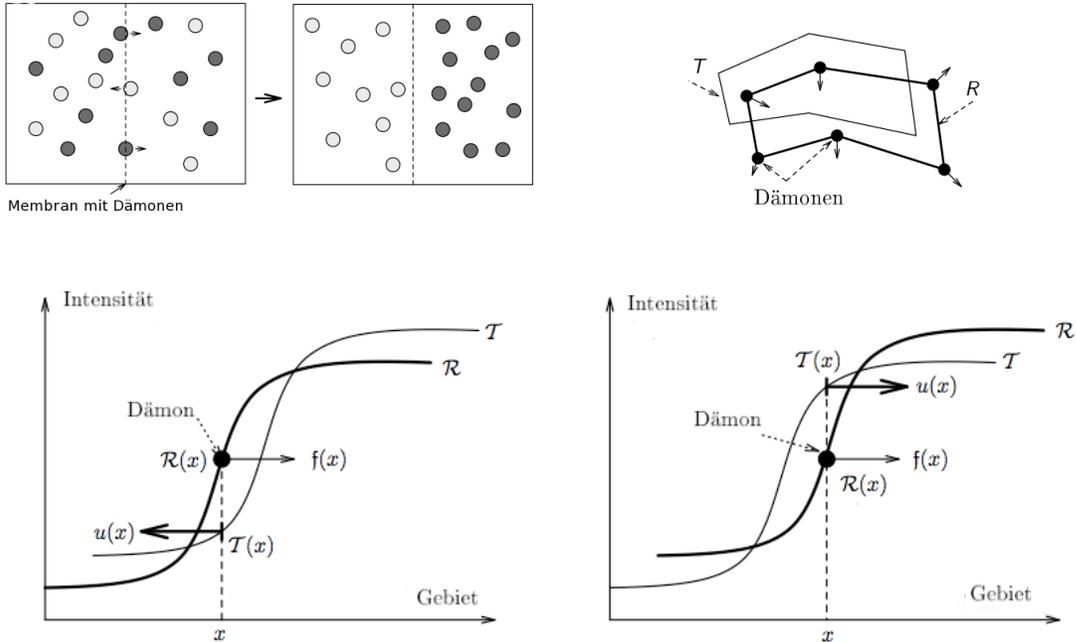


Abbildung 2.1: **Oben Links:** Illustration des Maxwellschen Gedankenexperiments; **Oben Rechts:** Jedem Dämon in \mathcal{R} ist ein Vektor zugeordnet, durch den die Punkte in \mathcal{T} verschoben werden (Abb. oben aus [44]); **Unten:** Das Template \mathcal{T} wird in Richtung $f(x)$ verschoben, wenn $\mathcal{R}(x) < \mathcal{T}(x)$ und in Richtung $-f(x)$, wenn $\mathcal{R}(x) > \mathcal{T}(x)$. Der Verschiebungsvektor $u(x)$ bestimmt die Stärke der Verschiebung (Abb. unten aus [43]).

Funktion

$$G_\sigma^\mu : \mathbb{R}^d \rightarrow \mathbb{R}^d \quad \text{mit} \quad G_\sigma^\mu(x) = (2\pi\sigma^2)^{-\frac{d}{2}} e^{-\frac{\|x-\mu\|_2^2}{2\sigma^2}}.$$

In [49, 10] wird die Glättung des Verrückungsfeldes als eine Approximation an eine fluidale [7] Regularisierung interpretiert. Wird statt des Verrückungsfeldes die sich aus diesem ergebene Transformation mit der Gauß-Funktion geglättet, wird es als eine Approximation an eine diffusive Regularisierung [14] interpretiert.

Bevor der Algorithmus für die Registrierung mit Dämonen formuliert wird, werden zwei Dämonenkräfte, die klassischen und die symmetrischen sowie die beiden Update-Regeln, die additive und die kompositive, vorgestellt.

Die Dämonenkräfte Für jeden Punkt im Bild \mathcal{T} wird ein Verrückungsvektor durch die Dämonenkräfte berechnet. In [43] wurden zwei Kräfte vorgeschlagen, von denen sich die einen von

der Gleichung des optischen Flusses herleiten lassen [22] und die anderen eine Symmetrisierung dieser Kräfte darstellen.

Die Idee des optischen Flusses ist, dass die Intensität eines sich bewegenden Objekts in einer Bildsequenz über die Zeit konstant bleibt. Die Gleichung des optischen Flusses [22] soll hier für den dreidimensionalen Fall hergeleitet werden. Sei \mathcal{R} das Referenzbild, \mathcal{T} das Templatebild und $\mathcal{I}(x(t), t)$ eine Bildsequenz für $t \in [t_1, t_2]$, so dass $\mathcal{R}(x) = \mathcal{I}(x, t_1)$ und $\mathcal{T}(x) = \mathcal{I}(x, t_2)$ mit $x = x(t)$. Es wird angenommen, dass $\mathcal{I}(x(t), t) = \text{konst.}$ Abgeleitet nach der Zeit t ergibt sich

$$\frac{\partial \mathcal{I}}{\partial x_1} \frac{\partial x_1}{\partial t} + \frac{\partial \mathcal{I}}{\partial x_2} \frac{\partial x_2}{\partial t} + \frac{\partial \mathcal{I}}{\partial x_3} \frac{\partial x_3}{\partial t} = - \frac{\partial \mathcal{I}}{\partial t}.$$

Es wird nach einer Bewegung gesucht, die zwischen den Zeitpunkten t_1 und t_2 stattgefunden hat. Der Punkt $x(t_2)$ kann als transformierte Version des Punktes $x(t_1)$ interpretiert werden: $x(t_2) = y(x(t_1))$. Wird angenommen, dass nur eine kleine Bewegung stattgefunden hat, also dass die Bilder \mathcal{R} und \mathcal{T} nur durch eine Zeiteinheit getrennt sind, ist $\frac{\partial \mathcal{I}(x(t), t)}{\partial t} = \mathcal{R}(x(t_1)) - \mathcal{T}_y(x(t_1))$. Mit der Bewegung $v(x) = (\frac{\partial x_1}{\partial t}, \frac{\partial x_2}{\partial t}, \frac{\partial x_3}{\partial t})$ und $x = x(t_1)$ ist damit

$$v(x) \cdot \nabla \mathcal{R}(x) = \mathcal{R}(x) - \mathcal{T}_y(x)$$

die Gleichung des optischen Flusses [22].

Um die Bewegung v eindeutig bestimmen zu können, müssen Bedingungen an sie gestellt werden. In diesem Fall wird gefordert, dass der Bewegungsvektor v eine minimale Länge haben soll. Dieses führt zu [44, 43] (vgl. Anhang A.2, S. 79)

$$v(x) = \frac{\mathcal{T}_y(x) - \mathcal{R}(x)}{\|\nabla \mathcal{R}(x)\|^2} \nabla \mathcal{R}(x). \quad (2.2)$$

Da Gleichung (2.2) undefiniert ist, falls der Gradient von \mathcal{R} null ist, wird in [39, 43] folgender alternativer Ausdruck für die Bewegung vorgeschlagen:

$$v(x) = \frac{\mathcal{T}_y(x) - \mathcal{R}(x)}{\|\nabla \mathcal{R}(x)\|^2 + \alpha^2 (\mathcal{R}(x) - \mathcal{T}_y(x))^2} \nabla \mathcal{R}(x). \quad (2.3)$$

Die Gleichung (2.3) modelliert die Bewegung eines Punktes x mit konstanter Intensität von \mathcal{R} nach \mathcal{T} . Die Verschiebung $u(x)$, die von \mathcal{T} nach \mathcal{R} stattgefunden hat, ist damit $u(x) = -v(x)$:

$$\begin{aligned} u^{\text{Opt}}(x) &= - \frac{\mathcal{T}_y(x) - \mathcal{R}(x)}{\|\nabla \mathcal{R}(x)\|^2 + \alpha^2 (\mathcal{R}(x) - \mathcal{T}_y(x))^2} \nabla \mathcal{R}(x) \\ &= \frac{\mathcal{R}(x) - \mathcal{T}_y(x)}{\|\nabla \mathcal{R}(x)\|^2 + \alpha^2 (\mathcal{R}(x) - \mathcal{T}_y(x))^2} \nabla \mathcal{R}(x). \end{aligned} \quad (2.4)$$

Der Vektor $u(x)$ wird im Dämonen-Algorithmus als Verrückungsvektor interpretiert, der einen Punkt x in \mathcal{T} in Richtung seines korrespondierenden Punktes in \mathcal{R} verschiebt. Der Parameter $\alpha \in \mathbb{R} \setminus \{0\}$ kontrolliert die Größe der Bewegung in jedem Schritt. Es kann für $\alpha > 0$ gezeigt werden (siehe Anhang A.2, S. 80), dass [39]

$$\|u(x)\| \leq \frac{1}{2\alpha}. \quad (2.5)$$

Im ursprünglichen Dämonen-Algorithmus von Thirion [44] wurde $\alpha = 1$ gewählt.

Die Kräfte u^{Opt} aus Gleichung (2.4) werden im Folgenden als klassische Kräfte bezeichnet und der Dämonen-Algorithmus, der diese Kräfte benutzt, als klassischer Dämonen-Algorithmus. Die klassischen Kräfte sind nicht symmetrisch, da nur der Gradient von dem Referenzbild zur Berechnung verwendet wird. Symmetrie der Transformation y heißt in der Bildregistrierung, dass, falls \mathcal{T}_y ähnlich zu \mathcal{R} ist, auch automatisch $\mathcal{R}(y^{-1})$ ähnlich zu \mathcal{T}_y ist, falls das Inverse der Transformation existiert. In [43] wurde eine modifizierte Kraft vorgeschlagen, bei der der Gradient des Templatebildes in die Berechnung der Transformation miteinbezogen wird:

$$u^{\text{Sym}}(x) = \frac{2(\mathcal{R}(x) - \mathcal{T}_y(x))(\nabla\mathcal{R}(x) + \nabla\mathcal{T}_y(x))}{\|\nabla\mathcal{R}(x) + \nabla\mathcal{T}_y(x)\|^2 + \alpha^2(\mathcal{R}(x) - \mathcal{T}_y(x))^2}. \quad (2.6)$$

Der Gradient vom Referenzbild, $\nabla\mathcal{R}(x)$, wurde durch den Mittelwert der beiden Gradienten, $\frac{1}{2}(\nabla\mathcal{R}(x) + \nabla\mathcal{T}_y(x))$, ersetzt. Diese Berechnung der Verschiebungsvektoren wird im Folgenden als symmetrische Kraft bezeichnet.

Es wurden auch andere Symmetrisierungen vorgeschlagen, die in der Praxis schneller konvergieren als der klassische Dämonen-Algorithmus [40, 53]. Diese werden in dieser Arbeit jedoch nicht weiter behandelt.

Seien $x^1, \dots, x^m, x^i \in \mathbb{R}^d$ die betrachteten Bildpunkte. Für die Berechnung der Transformation y aus der diskreten Verrückung $\mathbf{u} = (u(x^1)^T, \dots, u(x^m)^T)^T$ gibt es in der Literatur zwei Varianten [51]. Bei den sogenannten additiven Dämonen wird der Raum aller Vektorfelder von dem \mathbb{R}^d in den \mathbb{R}^d betrachtet. Dieser Raum ist ein Vektorraum, wobei der Update-Schritt aus der Addition der aktuellen Transformation mit der Verrückung \mathbf{u} besteht:

$$y \leftarrow y + \mathbf{u}.$$

Bei den kompositiven Dämonen betrachtet man die geometrischen Transformationen, deren natürliche Verknüpfung die Komposition von Funktionen ist. Im Update-Schritt des Verfahrens

wird die Verrückung \mathbf{u} mit der aktuellen Transformation \mathbf{y} verknüpft [51, 42]:

$$\mathbf{y} \leftarrow \mathbf{y} \circ (\text{id} + \mathbf{u}) \quad \text{mit } \text{id}(x) = x.$$

Beide Varianten des Update-Schritts sind mit den klassischen und symmetrischen Kräften in Algorithmus 1 dargestellt. Wie der kompositive Update-Schritt in der Praxis berechnet wird,

Algorithmus 1 Der Dämonen-Algorithmus

Eingabe: zwei diskrete Bilder R, T an Datenpunkten $x^1, \dots, x^m \in \mathbb{R}^d$, $m \in \mathbb{R}$, diskrete Transformation \mathbf{y} als Startwert, Standardabweichung $\sigma \in \mathbb{R}$

Ausgabe: Transformation \mathbf{y}

while !stop **do**

Berechne $\mathcal{R}, \mathcal{T}_y$ durch Interpolation

Berechne u für jedes x^i :
$$u(x^i) = \frac{\mathcal{R}(x^i) - \mathcal{T}_y(x^i)}{\|J(x^i)\|^2 + \alpha^2(\mathcal{R}(x^i) - \mathcal{T}_y(x^i))^2} J(x^i)$$

mit $J(x^i) = \nabla \mathcal{R}(x^i)$ **(klassisch)**

oder $J(x^i) = \frac{1}{2} (\nabla \mathcal{R}(x^i) + \nabla \mathcal{T}_y(x^i))$ **(symmetrisch)**

diskrete Verrückung $\mathbf{u} = (u(x^1), \dots, u(x^m))^T$

Fluidale Regularisierung: $\mathbf{u} \leftarrow G_\sigma^0 * \mathbf{u}$

Update-Schritt: $\mathbf{y} \leftarrow \mathbf{y} \circ (\text{id} + \mathbf{u})$ **(kompositiv)**

$\mathbf{y} \leftarrow \mathbf{y} + \mathbf{u}$ **(additiv)**

Diffusive Regularisierung: $\mathbf{y} \leftarrow G_\sigma^0 * \mathbf{y}$

end while

ist nicht trivial. Die Unterscheidung zwischen additiven und kompositiven Dämonen taucht in der Literatur erstmals in [42] auf. Für die Berechnung der kompositiven Dämonen wird in der ITK-Implementierung [50], die in dieser Arbeit verwendet wird, ein numerisches Verfahren benutzt, das eigentlich für eine diffeomorphe Erweiterung des Dämonen-Algorithmus eingesetzt wurde. Dieses Verfahren wird in Kapitel 6 kurz beschrieben.

Der Dämonen-Algorithmus ist sehr effizient in der Praxis und einfach zu implementieren, weswegen er in der Registrierung viel Anwendung gefunden hat. Er basiert jedoch auf einem Diffusionsmodell und nicht auf einem Minimierungsproblem, wie die meisten Registrierungsverfahren. Da die Formulierung als Optimierungsproblem eine gute mathematische Beschreibung für das Registrierungsproblem liefert, haben viele Arbeiten sich damit beschäftigt, dem Dämonen-Algorithmus einen Optimierungsrahmen zu geben und ihn damit in die bestehen-

den Registrierungsalgorithmen einzuordnen. In [39] wurde der Dämonen-Algorithmus mit den Kräften u^{Opt} (Gleichung (2.4)) in Beziehung gesetzt zu einer Minimierung eines speziellen Distanzmaßes, des SSD-Maßes (*Sum of Squared Differences*-Distanzmaß) mit einem Gradientenabstieg zweiter Ordnung und einer anschließenden Glättung mit Gauß-Funktionen. In [32, 14] wurde gezeigt, dass die Gauß-Funktion die Greensche Funktion der linearen Diffusionsgleichung ist und der Dämonen-Algorithmus damit als eine Approximation an eine Registrierung mit diffusiver Regularisierung interpretiert werden kann.

In [51, 49] wurde schließlich ein Registrierungsrahmen vorgestellt, mit dem der Dämonen-Algorithmus ein Minimierungsproblem löst, wobei die verschiedenen Dämonenkräfte auf die Wahl von unterschiedlichen Optimierern zurückzuführen sind. Dieser Rahmen basiert darauf, dass einige Klassen von Transformationen Lie-Gruppen bilden und die Optimierungsverfahren auf diesen Lie-Gruppen optimieren.

Lie-Gruppen sind von Bedeutung, da nicht alle Klassen von Transformationen Vektorräume bilden. Die Menge aller Abbildung von dem \mathbb{R}^d in den \mathbb{R}^d bildet einen Vektorraum. Betrachtet man jedoch Klassen bestimmter geometrischer Transformationen, wie z.B. die rigiden oder auch die diffeomorphen Transformationen, bilden diese keine Vektorräume. Sie sind bezüglich der Addition nicht abgeschlossen. Es können im Allgemeinen keine Minimierungsverfahren benutzt werden, die Vektorraumeigenschaften voraussetzen. Einige dieser Klassen von Transformationen bilden jedoch Lie-Gruppen, so dass Optimierungsverfahren auf Lie-Gruppen angewendet werden können [49].

In diesem Kapitel wurde ein Überblick über die Bildregistrierung, insbesondere über einen Algorithmus zur Bildregistrierung, den Dämonen-Algorithmus von Thirion, gegeben. Mit Hilfe von Registrierungsverfahren basierend auf der Optimierung auf Lie-Gruppen ist es gelungen, diesen Algorithmus als ein Verfahren zur Lösung eines Optimierungsproblems zu interpretieren [49]. Um diese Registrierungsverfahren beschreiben zu können, werden in den nachfolgenden Kapiteln einige ausgesuchte Begriffe aus der Differentialgeometrie und der Theorie der Lie-Gruppen eingeführt.

Kapitel 3

Einführung in die Differentialgeometrie

Um Bildregistrierungsmethoden unter der Verwendung von Lie-Gruppen verstehen und beschreiben zu können, werden einige technische Begriffe aus der Lie-Gruppen-Theorie benötigt. In [49] wird auf diese nicht näher eingegangen. In dieser Arbeit werden sie im Besonderen thematisiert.

Eine Lie-Gruppe ist gleichzeitig eine Gruppe und eine differenzierbare Mannigfaltigkeit, so dass die Gruppenverknüpfung und die Inversenbildung differenzierbar sind. Dieses Kapitel und das folgende sind inhaltlich eng miteinander verknüpft. Während in diesem differenzierbare Mannigfaltigkeiten und Begrifflichkeiten wie Tangentialvektoren und Vektorfelder eingeführt werden, beschreibt das nächste Kapitel mit deren Hilfe spezielle Strukturen auf Lie-Gruppen.

In diesem Kapitel werden differenzierbare Mannigfaltigkeiten näher betrachtet. Es werden einige theoretische Grundbegriffe, wie der des topologischen Raumes und des Hausdorffraumes und spezielle Klassen von Abbildungen, wie Diffeomorphismen und Isomorphismen, definiert. Zusammen mit diesen und den sogenannten Karten und Atlanten können differenzierbare Mannigfaltigkeiten beschrieben werden.

Es werden zwei Definitionen für Tangentialvektoren an Punkten der Mannigfaltigkeit und für Vektorfelder auf der Mannigfaltigkeit präsentiert. Es wird der Vektorfeld-Kommutator vorgestellt, der eine abgeschlossene Verknüpfung für Vektorfelder liefert. Schließlich werden

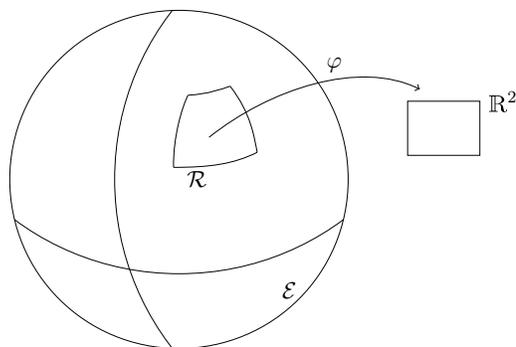


Abbildung 3.1: Die Sphäre im \mathbb{R}^3 oder auch die Erdoberfläche \mathcal{E} ist eine Mannigfaltigkeit. Eine Teilmenge $\mathcal{R} \subset \mathcal{E}$ kann durch eine Karte φ lokal auf eine Ebene im \mathbb{R}^2 abgebildet werden.

Funktionen auf und zwischen differenzierbaren Mannigfaltigkeiten betrachtet und deren Ableitungen beschrieben. Für die Funktionen zwischen Mannigfaltigkeiten führt dies zum Begriff des Differential, das vom Tangentialbündel der einen ins Tangentialbündel der anderen Mannigfaltigkeit abbildet.

Für dieses Kapitel wurde überwiegend folgende Literatur benutzt: [16, 27, 54, 25, 1, 26, 21, 4, 9, 45].

3.1 Differenzierbare Mannigfaltigkeiten

Eine Mannigfaltigkeit ist ein topologischer Raum (siehe Definition 3.1), der lokal dem euklidischen Raum entspricht. Ein anschauliches Beispiel ist die Sphäre im \mathbb{R}^3 oder die Erdoberfläche. Für jede Region der Erde existiert eine Karte (eine Landkarte oder ein Stadtplan), so dass diese Region auf eine Ebene im \mathbb{R}^2 abgebildet werden kann (vgl. Abb. 3.1). An jede Karte schließt eine andere an. Eine Sammlung von Karten, die jede Region der Erde abbildet, heißt Atlas. Um abstrakte Mannigfaltigkeiten definieren zu können, müssen die Begriffe der Karte und des Atlas auf sie übertragen werden. Dafür sind vorher einige theoretische Begriffe von Nöten.

Definition 3.1 (Topologie, Topologischer Raum, Hausdorffraum, siehe [27, 16, 4])

Ein Mengensystem Y , bestehend aus offenen Teilmengen einer Grundmenge X , ist eine Topologie, wenn gilt

(i) \emptyset, X offen,

(ii) $U_i \in Y$ offen und $n < \infty \Rightarrow \bigcap_{i=1}^n U_i$ offen,

(iii) $U_i \in Y$ offen und $n < \infty \Rightarrow \bigcup_{i=1}^n U_i$ offen.

Ein topologischer Raum (X, Y) ist eine Grundmenge X mit einer Topologie Y auf X .

Ein Hausdorffraum M ist ein topologischer Raum, in dem alle paarweise verschiedenen Punkte durch Umgebungen (offene Teilmengen) voneinander getrennt werden, d.h. es gilt

$$x, y \in M, x \neq y \Rightarrow \exists U(x), V(y) : U(x) \cap V(y) = \emptyset.$$

Beispiele für Hausdorffräume sind alle metrischen und diskreten Räume [27]. Ein Gegenbeispiel ist z.B. der topologische Raum (X, \emptyset) , wenn die Grundmenge X mehr als ein Element enthält. Die Topologie $Y = \emptyset$ heißt auch triviale Topologie [27].

Für die theoretischen Betrachtungen in dieser Arbeit werden einige Klassen von Abbildungen benötigt. Die einen sind die Diffeomorphismen, die anderen die Isomorphismen.

Definition 3.2 (Homöomorphismus, Diffeomorphismus) *Ein Homöomorphismus ist eine Abbildung zwischen zwei topologischen Räumen, die bijektiv und stetig ist und deren Umkehrabbildung auch stetig ist.*

Ein C^k -Diffeomorphismus ist ein k -mal stetig differenzierbarer Homöomorphismus, dessen Umkehrabbildung auch k -mal stetig differenzierbar ist.

Definition 3.3 (Homomorphismus, Isomorphismus) *Ein Homomorphismus ist eine strukturerhaltene Abbildung zwischen algebraischen Strukturen. Ein Gruppenhomomorphismus h , der von der Gruppe $(G_1, *)$ in die Gruppe (G_2, \circ) abbildet, erfüllt folgende Eigenschaft $\forall u, v \in G_1$:*

$$h(u * v) = h(u) \circ h(v).$$

Ein Isomorphismus ist eine bijektive Abbildung f auf algebraischen Strukturen, so dass f und das Inverse von f Homomorphismen sind.

Differenzierbare Mannigfaltigkeiten Mit den eben definierten Begriffen kann jetzt die Definition für die abstrakten Äquivalente zu den Karten und Atlanten der Erdoberfläche gegeben werden.

Definition 3.4 (Karte und Atlas, siehe [54, 26]) Sei \mathcal{M} ein Hausdorffraum und $\mathcal{U} \subset \mathcal{M}$ offen. Das Paar (\mathcal{U}, φ) heißt d -dimensionale Karte oder Koordinatensystem auf \mathcal{M} , wenn φ ein Homöomorphismus ist und

$$\varphi : \mathcal{U} \subset \mathcal{M} \rightarrow \varphi(\mathcal{U}) \subset \mathbb{R}^n.$$

Für $p \in \mathcal{U}$ heißen $\varphi(p) \in \mathbb{R}^n$ die Koordinaten von p der Karte φ .

Zwei Karten $(\mathcal{U}_1, \varphi_1)$ und $(\mathcal{U}_2, \varphi_2)$ heißen kompatibel, wenn $\mathcal{U}_1 \cap \mathcal{U}_2 = \emptyset$ oder wenn die Funktion

$$\varphi_1 \circ \varphi_2^{-1} : \varphi_2(\mathcal{U}_1 \cap \mathcal{U}_2) \subset \mathbb{R}^n \rightarrow \varphi_1(\mathcal{U}_1 \cap \mathcal{U}_2) \subset \mathbb{R}^n,$$

die den Wechsel von der Karte φ_1 zu der Karte φ_2 realisiert, ein C^k -Diffeomorphismus ist.

Die Funktion $\varphi_1 \circ \varphi_2^{-1}$ wird auch Koordinatenwechsel genannt.

Ein n -dimensionaler Atlas von \mathcal{M} ist eine Familie $\{(\mathcal{U}_i, \varphi_i), i \in I \subset \mathbb{R}\}$ von paarweisen kompatiblen Karten (alle Koordinatenwechsel sind C^k -Diffeomorphismen) mit $\bigcup_I \mathcal{U}_i = \mathcal{M}$.

Mit diesen Begriffen können differenzierbare Mannigfaltigkeiten eingeführt werden:

Definition 3.5 (Mannigfaltigkeit) Ein Hausdorffraum mit einem n -dimensionalen Atlas, deren Koordinatenwechsel C^k -Diffeomorphismen sind, ist eine n -dimensionale differenzierbare Mannigfaltigkeit.

Eine Mannigfaltigkeit entspricht durch den Atlas lokal dem euklidischen n -dimensionalen Raum. Das bedeutet, dass eine offene Umgebung $\mathcal{U} \subset \mathcal{M}$ durch eine Karte φ im \mathbb{R}^n dargestellt werden kann (siehe Abb. 3.2). Die Karten werden auch lokale Karten genannt, weil sie nur lokal auf einer Teilmenge auf \mathcal{M} definiert sind. Betrachtet man eine beliebige Karte (\mathcal{U}, φ) , $\varphi : \mathcal{U} \subset \mathcal{M} \rightarrow \tilde{\mathcal{U}} \subset \mathbb{R}^n$, so kann \mathcal{U} mit $\tilde{\mathcal{U}}$ identifiziert und ein Punkt $p \in \mathcal{U} \subset \mathcal{M}$ durch seine Koordinaten $(u_1, \dots, u_n) = \varphi(p)$ in $\tilde{\mathcal{U}}$ repräsentiert werden [26].

Beispiele für differenzierbare Mannigfaltigkeiten sind:

1. Der \mathbb{R}^n ist eine differenzierbare Mannigfaltigkeit mit einem Atlas, der eine einzige Karte enthält: $(\mathbb{R}^n, \text{id}_{\mathbb{R}^n})$. Hierbei ist $\text{id}_{\mathbb{R}^n}$ die Identitätsabbildung auf \mathbb{R}^n [26].

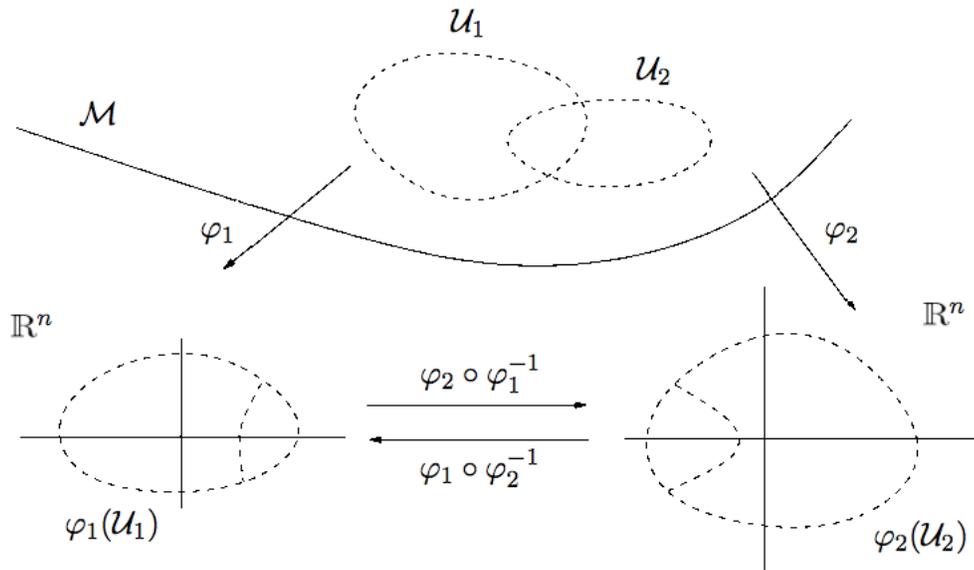


Abbildung 3.2: Eine Mannigfaltigkeit \mathcal{M} mit den Karten $(\mathcal{U}_1, \varphi_1), (\mathcal{U}_2, \varphi_2)$ und den Koordinatenwechseln $\varphi_2 \circ \varphi_1^{-1}, \varphi_1 \circ \varphi_2^{-1}$ (Abb. aus [1]).

2. Die 2-dimensionale Sphäre $\mathbb{S}^2 = \{x \in \mathbb{R}^3 \mid \|x\| = 1\}$ ist eine differenzierbare Mannigfaltigkeit mit folgendem Atlas: Seien \mathcal{U}_i^+ und \mathcal{U}_i^- Teilmengen von \mathbb{S}^2 mit

$$\begin{aligned} \mathcal{U}_i^+ &= \{x = (x_1, x_2, x_3) \in \mathbb{S}^2 \mid x_i > 0\} \quad \text{und} \\ \mathcal{U}_i^- &= \{x = (x_1, x_2, x_3) \in \mathbb{S}^2 \mid x_i < 0\}. \end{aligned}$$

Dies sind die Mengen, in denen die i -te Koordinate jeweils positiv bzw. negativ ist. Sei $\mathbb{B}^2 = \{x \in \mathbb{R}^2 \mid \|x\| < 1\}$ die offene Einheitskreisscheibe im \mathbb{R}^2 . Jede Karte des \mathbb{S}^2 bildet eine der Mengen \mathcal{U}_i^\pm auf die Kreisscheibe ab, indem die i -te Koordinate weggelassen wird, z.B.

$$\varphi_1^\pm : \mathcal{U}_1^\pm \rightarrow \mathbb{B}^2 \quad \text{mit} \quad \varphi_1^\pm(x_1, x_2, x_3) = (x_2, x_3)$$

und analog φ_2^\pm und φ_3^\pm [26].

3. Die Menge $M(m \times n, \mathbb{R})$ der reellen $m \times n$ Matrizen ist ein Vektorraum der Dimension mn mit der Matrizenaddition und Skalarmultiplikation. Damit ist sie eine differenzierbare mn -dimensionale Mannigfaltigkeit [26].

4. Die *general linear group* $GL(n, \mathbb{R})$ ist die Menge der invertierbaren reellen $n \times n$ Matrizen und eine offene Teilmenge von $M(n \times n, \mathbb{R})$. Damit ist $GL(n, \mathbb{R})$ eine n^2 -dimensionale differenzierbare Mannigfaltigkeit [26].

Sei \mathcal{M} im Folgenden eine n -dimensionale differenzierbare Mannigfaltigkeit und (\mathcal{U}, φ) eine Karte auf \mathcal{M} . Auf einer Mannigfaltigkeit können Kurven und Funktionen definiert werden [54, 25, 26, 9]. Eine Kurve bildet von einem Intervall $I \subset \mathbb{R}$ in die Mannigfaltigkeit \mathcal{M} ab. Dieselbe Kurve kann durch unterschiedliche Parametrisierungen dargestellt werden. Dabei kann ohne Einschränkung angenommen werden, dass $0 \in I$.

Eine Funktion f auf \mathcal{M} ist eine Abbildung von \mathcal{M} in die reellen Zahlen, auf der Erdoberfläche z.B. die Temperaturangabe an einem Ort. Durch den Kompatibilitätsbegriff der Karten können Aussagen über die Differenzierbarkeitseigenschaften von Funktionen getroffen werden. Eine Funktion f heißt unendlich oft stetig differenzierbar auf \mathcal{M} , wenn für jede Karte (\mathcal{U}, φ) die Funktion

$$f \circ \varphi^{-1} : \varphi(\mathcal{U}) \subset \mathbb{R}^n \rightarrow \mathbb{R}$$

unendlich oft stetig differenzierbar ist. Die Funktion $f \circ \varphi^{-1}$ heißt Interpretation von f in φ . Im Folgenden sei der Raum der unendlich oft stetig differenzierbaren Funktionen auf \mathcal{M} gekennzeichnet durch

$$C^\infty(\mathcal{M}) := \{f : \mathcal{M} \rightarrow \mathbb{R} \mid f \text{ unendlich oft stetig differenzierbar}\}.$$

Die partiellen Ableitungen von f in einem Punkt p werden über die partiellen Ableitungen der Interpretation von f bezüglich der Koordinaten $\varphi(p) = (u_1, \dots, u_n)$ bestimmt [26]:

$$\left. \frac{\partial}{\partial u_i} f \right|_p = \left. \frac{\partial}{\partial u_i} (f \circ \varphi^{-1}) \right|_{\varphi(p)}. \quad (3.1)$$

3.2 Tangentialvektoren, Vektorfelder und Differentiale

Für die Optimierung auf Lie-Gruppen werden neben den Mannigfaltigkeiten noch weitere theoretische Begriffe benötigt. In diesem Abschnitt werden Tangentialvektoren, Vektorfelder auf Mannigfaltigkeiten und Differentiale von Abbildungen zwischen Mannigfaltigkeiten eingeführt.

Tangentialvektoren Um Tangentialvektoren auf einer abstrakten Mannigfaltigkeit anschaulich beschreiben zu können, wird wieder die Erdoberfläche, hier mit \mathcal{E} bezeichnet, betrachtet.

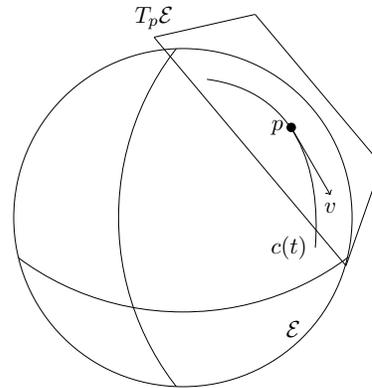


Abbildung 3.3: Der Tangentialvektor v von \mathcal{E} in p wird bestimmt über die Ableitung der Kurve c an der Stelle 0. Die Menge aller Tangentialvektoren in p bildet den Tangentialraum $T_p\mathcal{E}$ von p .

Eine Kurve $c : I \subset \mathbb{R} \rightarrow \mathcal{E}$ auf der Erdoberfläche kann z.B. eine Straße oder einen Weg darstellen. Sei $p \in \mathcal{E}$ ein Ort auf der Erde, durch den die Kurve für $t = 0$ führt, also $c(0) = p$. Die Ableitung der Kurve c an der Stelle 0 ist der Tangentialvektor v in p (siehe Abb. 3.3):

$$\left. \frac{dc(t)}{dt} \right|_{t=0} = c'(0) = v(p).$$

Anschaulich sind das die Geschwindigkeitsvektoren. Durch einen Punkt der Erdoberfläche können unendlich viele Kurven mit $c(0) = p$ und $c'(0) = v$ verlaufen. Ein Punkt besitzt damit unendliche viele Tangentialvektoren. Diese bilden eine Ebene, die die Erde im Punkt p berührt. Diese Ebene heißt Tangentialraum $T_p\mathcal{E}$.

Die Erde ist eingebettet in den \mathbb{R}^3 , so dass die Ableitung der Kurve c existiert. Auf abstrakten Mannigfaltigkeiten existieren die Ableitungen

$$c'(0) = \lim_{t \rightarrow 0} \frac{c(t) - c(0)}{t}$$

nicht, da die Differenz der Punkte $c(0), c(t) \in \mathcal{M}$ auf einer Mannigfaltigkeit \mathcal{M} im Allgemeinen nicht definiert ist. Nun gibt es mehrere Möglichkeiten Tangentialvektoren an abstrakten Mannigfaltigkeiten zu beschreiben. Im Folgenden werden zwei Definitionen präsentiert. Die erste nutzt die Karten der Mannigfaltigkeit, um Ableitungen von Kurven zu bestimmen und die zweite Richtungsableitungen von Funktionen auf der Mannigfaltigkeit. Es wird gezeigt, dass die beiden Definitionen äquivalent sind, so dass auch für beide der Begriff Tangentialvektor gewählt wird.

Bei der ersten Definition des Tangentialvektors werden die Karten der Mannigfaltigkeit genutzt, die eine Kurve c an einem Punkt $c(0) = p \in \mathcal{M}$ von der Mannigfaltigkeit in den \mathbb{R}^d überführen. Die dortige Ableitung wird als Tangentialvektor interpretiert.

Definition 3.6 Sei \mathcal{M} eine Mannigfaltigkeit, $p \in \mathcal{M}$, (U, φ) eine Karte auf \mathcal{M} und c_1 und c_2 zwei C^∞ -Kurven auf \mathcal{M} mit $c_1(0) = c_2(0) = p$. Die Verknüpfungen $\varphi \circ c_1, \varphi \circ c_2 : \mathbb{R} \rightarrow \mathbb{R}^n$ sind differenzierbar und heißen tangential zueinander in p , wenn

$$(\varphi \circ c_1)'(0) = (\varphi \circ c_2)'(0).$$

Alle Kurven c auf \mathcal{M} , die zueinander tangential in p sind, bilden eine Äquivalenzklasse. Diese ist eindeutig durch $(\varphi \circ c)'(0)$ bestimmt.

Definition 3.7 (Tangentialvektor I, siehe [34, 54, 26]) Die Äquivalenzklasse bezüglich zueinander tangentialen Kurven in p heißt Tangentialvektor \bar{c} von \mathcal{M} in p .

Die Äquivalenzklasse \bar{c} ist zwar definiert über eine Karte, ist jedoch unabhängig von der speziellen Wahl der Karte [34]. Diese Definition von Tangentialvektoren als Ableitungen von Kurven kommt der Vorstellung eines Tangentialvektors auf der Erdoberfläche sehr nahe. Für die weiteren Betrachtungen ist sie jedoch etwas unpraktisch, so dass noch eine andere Definition eingeführt wird, die äquivalent zu der obigen ist. Die Tangentialvektoren von p werden jetzt nicht über die Karten bestimmt, sondern über die Richtungsableitungen von Funktionen auf \mathcal{M} entlang der Kurven durch p :

Definition 3.8 (Tangentialvektor II, siehe [34, 54, 1, 26]) Sei \mathcal{M} eine Mannigfaltigkeit, $p \in \mathcal{M}$ und c eine C^∞ -Kurve auf \mathcal{M} . Für ein beliebiges $f \in C^\infty(\mathcal{M})$, $f : \mathcal{M} \rightarrow \mathbb{R}$, ist

$$\xi_p(f) = \dot{c}(0)f := \left. \frac{d(f(c(t)))}{dt} \right|_{t=0}.$$

Die Abbildung $\xi_p : C^\infty(\mathcal{M}) \rightarrow \mathbb{R}$ heißt Tangentialvektor von \mathcal{M} in p .

Der Tangentialvektor ist nach der zweiten Definition eine Abbildung, die jeder Funktion auf \mathcal{M} ihre Richtungsableitung zuordnet. Man kann zeigen, dass es einen Isomorphismus zwischen den beiden Definitionen des Tangentialvektors gibt und zwar die Abbildung

$$\bar{c}_p \mapsto \left(f \mapsto \left. \frac{df(c(t))}{dt} \right|_{t=0} \right),$$

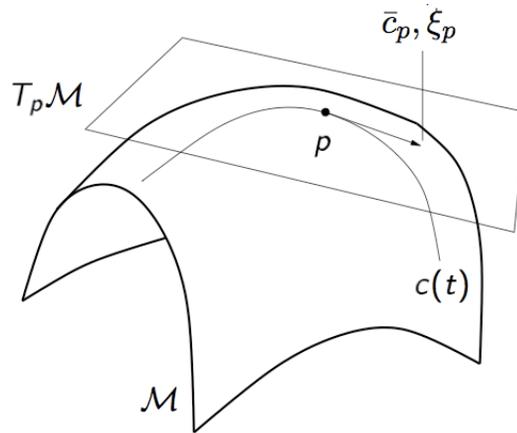


Abbildung 3.4: Der Tangentialvektor einer Mannigfaltigkeit \mathcal{M} im Punkt $p = c(0)$ kann durch \bar{c} (Def. 3.7) oder $\xi_p(f)$ (Def. 3.8) angegeben werden. Der Vektor \bar{c}_p ist hier ein Repräsentant der Äquivalenzklasse \bar{c} . Der Tangentialraum ist für \bar{c}_p und ξ_p derselbe (Abb. aus [1]).

wobei \bar{c}_p ein Repräsentant der Äquivalenzklasse \bar{c} ist. Es ist somit egal, welche Definition für Tangentialvektoren verwendet wird [34, 26].

Die Tangentialvektoren aus Def. 3.8 sind sogenannte Derivationen in p , da sie die Gleichung $\xi_p(fg) = f(p)\xi_p(g) + g(p)\xi_p(f)$ für $f, g \in C^\infty(\mathcal{M})$ erfüllen [54, 26].

Auch auf abstrakten Mannigfaltigkeiten existieren unendlich viele Tangentialvektoren eines Punktes p und jeder Punkt besitzt einen Tangentialraum (siehe Abb. 3.4). Alle Tangentialräume einer Mannigfaltigkeit zusammengefasst bilden das Tangentialbündel.

Definition 3.9 (Tangentialraum, Tangentialbündel) Sei \mathcal{M} eine Mannigfaltigkeit und $p \in \mathcal{M}$. Die Menge aller Tangentialvektoren von \mathcal{M} in p heißt Tangentialraum von \mathcal{M} in p und wird mit $T_p \mathcal{M}$ bezeichnet.

Die Vereinigung aller Tangentialräume von \mathcal{M} bildet das Tangentialbündel $T\mathcal{M}$ von \mathcal{M} :

$$T\mathcal{M} = \bigcup_{p \in \mathcal{M}} T_p \mathcal{M}.$$

Der Tangentialraum eines beliebigen Punktes $p \in \mathcal{M}$ ist ein Vektorraum [26].

Neben den Tangentialvektoren und Tangentialräumen ist für die Theorie der Lie-Gruppen auch der Begriff des Vektorfeldes sehr wichtig.

Vektorfelder Analog zu den Tangentialvektoren werden für Vektorfelder zwei unterschiedliche, aber äquivalente Definitionen vorgestellt.

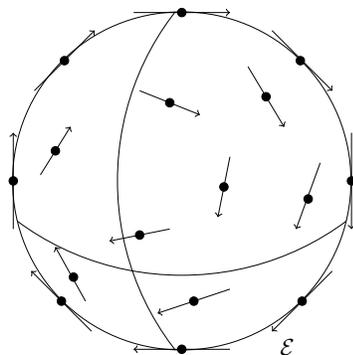


Abbildung 3.5: Ein Vektorfeld auf der Erdoberfläche \mathcal{E} (Sphäre im \mathbb{R}^3).

Die erste Definition ist intuitiv:

Definition 3.10 (Vektorfeld I) Ein Vektorfeld X auf einer Mannigfaltigkeit \mathcal{M} ist eine Abbildung, die jedem Punkt $p \in \mathcal{M}$ einen Tangentialvektor aus seinem Tangentialraum zuordnet:

$$X : \mathcal{M} \rightarrow T\mathcal{M}$$

$$X(p) \in T_p\mathcal{M}.$$

Ein Vektorfeld X heißt glatt, wenn es als Abbildung $X : \mathcal{M} \rightarrow T\mathcal{M}$ unendlich oft stetig differenzierbar ist. Die Menge aller glatten Vektorfelder auf \mathcal{M} wird mit $\mathfrak{X}(\mathcal{M})$ bezeichnet. Auf der Erdoberfläche, die eine 2-dimensionale Mannigfaltigkeit ist, kann ein Vektorfeld als eine Menge von Pfeilen (Tangentialvektoren) beschrieben werden. An jedem Punkt der Mannigfaltigkeit ist ein Pfeil angeheftet (vgl. Abb. 3.5).

Genau wie Tangentialvektoren (vgl. Def. 3.8) können Vektorfelder auch auf Funktionen wirken, die auf der Mannigfaltigkeit definiert sind. Sei $X \in \mathfrak{X}(\mathcal{M})$ und $f \in C^\infty(\mathcal{M})$. Da durch Def. 3.8 Tangentialvektoren auch als Abbildung von C^∞ nach \mathbb{R} interpretiert werden können, kann eine Funktion definiert werden durch $p \mapsto (X(p))(f)$, die von \mathcal{M} nach $C^\infty(\mathcal{M})$ abbildet. Der Ausdruck $(X(p))(f)$ ist ein Tangentialvektor nach Definition 3.8. Durch die Funktion

$$f \mapsto (p \mapsto (X(p))(f))$$

ist ein Isomorphismus von $C^\infty(\mathcal{M})$ nach $C^\infty(\mathcal{M})$ gegeben [34].

Definition 3.11 (Vektorfeld II, siehe [54, 26, 21]) Sei $f \in C^\infty(\mathcal{M})$, $p \in \mathcal{M}$ und $\xi_p(f) \in C^\infty(\mathcal{M})$. Die Abbildung

$$\begin{aligned} X : C^\infty(\mathcal{M}) &\rightarrow C^\infty(\mathcal{M}), \\ X(f)(p) &= \xi_p(f) \end{aligned}$$

heißt Vektorfeld und weist jeder Funktion f die Tangentialvektoren im Punkte p zu.

Das Vektorfeld $X(f)(p)$ ist also die Richtungsableitung von f in Richtung X .

Für jedes Vektorfeld $X : C^\infty(\mathcal{M}) \rightarrow C^\infty(\mathcal{M})$ und für alle $\alpha, \beta \in \mathbb{R}$, $f, g \in C^\infty(\mathcal{M})$ gilt

$$\begin{aligned} X(\alpha f + \beta g) &= \alpha X(f) + \beta X(g), \\ X(fg) &= X(f)g + fX(g), \end{aligned} \tag{3.2}$$

wobei die Multiplikation eines Vektorfeldes mit einer Funktion auf \mathcal{M} definiert ist durch: $(gX(f))(p) = g(p)X(f)(p)$ [54, 1, 21]. Mit diesen Eigenschaften bilden die Vektorfelder einen Vektorraum [26].

Seien $X, Y \in \mathfrak{X}(\mathcal{M})$ zwei glatte Vektorfelder und damit stetig differenzierbare Funktionen $X, Y : C^\infty(\mathcal{M}) \rightarrow C^\infty(\mathcal{M})$. Die Vektorfelder können miteinander verknüpft werden, so dass $(X \circ Y)(f) = X(Y(f))$. Eine derartige Verknüpfung erfüllt jedoch nicht die Eigenschaften (3.2), hauptsächlich, weil sie zweite Ableitungen enthält [54, 26]. Durch den sogenannten Vektorfeld-Kommutator als Verknüpfung bleiben die Eigenschaften (3.2) erhalten:

Satz 3.1 (Vektorfeld-Kommutator) Seien $X, Y \in \mathfrak{X}(\mathcal{M})$ und sei eine Verknüpfung der Vektorfelder definiert durch

$$\begin{aligned} [.,.] : \mathfrak{X}(\mathcal{M}) \times \mathfrak{X}(\mathcal{M}) &\rightarrow \mathfrak{X}(\mathcal{M}), \\ [X, Y](f) &= X(Y(f)) - Y(X(f)), \quad \forall f \in C^\infty(\mathcal{M}). \end{aligned}$$

Dann ist $[X, Y](f) \in \mathfrak{X}(\mathcal{M})$ und erfüllt damit die Gleichungen (3.2). Die Verknüpfung heißt Vektorfeld-Kommutator.

Beweis: (siehe [26]) Zu zeigen ist, dass für alle $\alpha, \beta \in \mathbb{R}$ und $f, g \in C^\infty(\mathcal{M})$ die Gleichungen (3.2) erfüllt sind, also dass $[X, Y](\alpha f + \beta g) = \alpha[X, Y](f) + \beta[X, Y](g)$ und $[X, Y](fg) = [X, Y](f)g + f[X, Y](g)$ gilt.

Für $X, Y \in \mathfrak{X}(\mathcal{M})$ gilt:

$$\begin{aligned}
 [X, Y](\alpha f + \beta g) &= X(Y(\alpha f + \beta g)) - Y(X(\alpha f + \beta g)) \\
 &= X(\alpha Y(f) + \beta Y(g)) - Y(\alpha X(f) + \beta X(g)) \\
 &= \alpha X(Y(f)) + \beta X(Y(g)) - \alpha Y(X(f)) - \beta Y(X(g)) \\
 &= \alpha [X, Y](f) + \beta [X, Y](g). \\
 [X, Y](fg) &= X(Y(fg)) - Y(X(fg)) \\
 &= X(Y(f)g + f(Y(g))) - Y(X(f)g + fX(g)) \\
 &= X(Y(f))g + Y(f)X(g) + X(f)Y(g) + fX(Y(g)) \\
 &\quad - Y(X(f))g - X(f)Y(g) - Y(f)X(g) - fY(X(g)) \\
 &= [X, Y](f)g + f[X, Y](g).
 \end{aligned}$$

□

Ein weiterer Begriff ist für die Betrachtungen in den nächsten Kapiteln von Bedeutung, und zwar das Differential einer stetig differenzierbaren Abbildung zwischen Mannigfaltigkeiten.

Differential Die Abbildung $F : \mathcal{M} \rightarrow \mathcal{N}$ ist eine Abbildung zwischen zwei Mannigfaltigkeiten \mathcal{M} und \mathcal{N} . Sie ist genau dann unendlich oft stetig differenzierbar, wenn für alle $f \in C^\infty(\mathcal{N})$ die Funktion $f \circ F : \mathcal{M} \rightarrow \mathbb{R}$ aus dem $C^\infty(\mathcal{M})$ ist.

Das Differential einer Abbildung zwischen zwei Mannigfaltigkeiten ist ihre Richtungsableitung in Richtung eines Tangentialvektors [54, 1, 21].

Definition 3.12 (Differential) Für eine unendlich oft stetig differenzierbare Funktion $F : \mathcal{M} \rightarrow \mathcal{N}$ zwischen zwei Mannigfaltigkeiten \mathcal{M}, \mathcal{N} ist das Differential $DF : T\mathcal{M} \rightarrow T\mathcal{N}$ eine Abbildung zwischen den Tangentialbündeln von \mathcal{M} und \mathcal{N} und das Differential von F im Punkte $p \in \mathcal{M}$ eine Abbildung zwischen den Tangentialräumen von \mathcal{M} und \mathcal{N} im Punkt p bzw. $F(p)$ und es gilt für alle $\xi_p \in T_p\mathcal{M}$, $f \in C^\infty(\mathcal{N})$:

$$\begin{aligned}
 DF &: T\mathcal{M} \rightarrow T\mathcal{N}, \\
 (DF)_p &: T_p\mathcal{M} \rightarrow T_{F(p)}\mathcal{N}, \\
 (DF)_p(\xi_p(f)) &= \xi_p(f \circ F).
 \end{aligned}$$

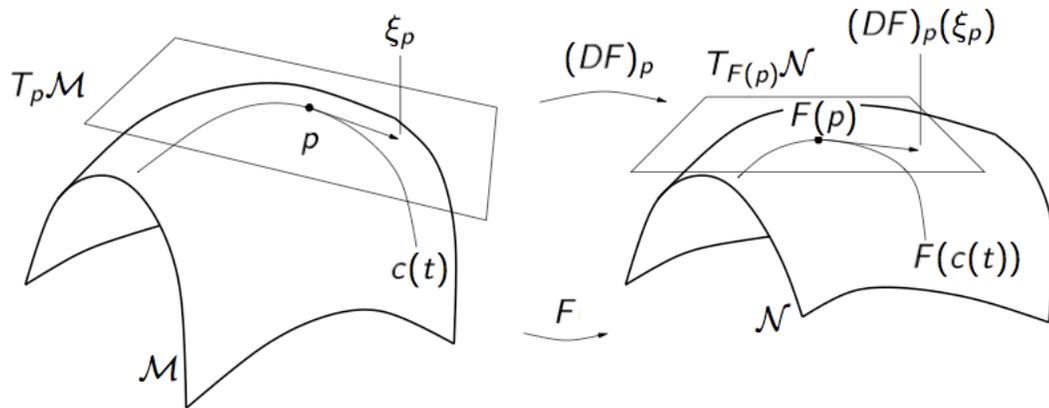


Abbildung 3.6: Das Differential $(DF)_p$ einer Funktion $F : \mathcal{M} \rightarrow \mathcal{N}$ bildet einen Tangentialvektor $\xi_p \in T_p \mathcal{M}$ auf den Tangentialvektor $(DF)_p(\xi_p) \in T_{F(p)} \mathcal{N}$ ab (Abb. aus [1]).

Wird ein Punkt $p \in \mathcal{M}$ durch eine Abbildung $F : \mathcal{M} \rightarrow \mathcal{N}$ auf den Punkt $F(p) \in \mathcal{N}$ abgebildet, so bildet das Differential von F , $(DF)_p$, einen Tangentialvektor $\xi_p \in T_p \mathcal{M}$ auf einen Tangentialvektor $(DF)_p(\xi_p) \in T_{F(p)} \mathcal{N}$ ab (vgl. Abb. 3.6).

In der Literatur findet man das Differential auch unter dem Namen Pushforward [26].

Mit Hilfe des Differentials kann eine weitere Eigenschaft des Vektorfeld-Kommutators gegeben werden, welche im nächsten Kapitel Verwendung finden wird. Angenommen, die Funktion $F : \mathcal{M} \rightarrow \mathcal{N}$ ist ein Diffeomorphismus. Dann gilt für $X, Y \in \mathfrak{X}(\mathcal{M})$:

$$DF([X, Y]) = [DF(X), DF(Y)]. \quad (3.3)$$

Der Beweis kann nachgelesen werden in [26].

Zum Abschluss dieses Kapitels wird der Satz der lokalen Invertierbarkeit, allerdings ohne Beweis, angegeben. Dieser Satz wird für die Betrachtungen im nächsten Kapitel benötigt. Der Beweis ist z.B. nachzulesen in [25, 21, 4].

Satz 3.2 (Satz der lokalen Invertierbarkeit) *Seien \mathcal{M}, \mathcal{N} zwei Mannigfaltigkeiten und $F : \mathcal{M} \rightarrow \mathcal{N}$ eine Abbildung. Ist das Differential $(DF)_p$ von F in einem Punkt $p \in \mathcal{M}$ ein Isomorphismus, so gibt es eine offene Umgebung $U \subset \mathcal{M}$ von p derart, dass $V := F(U)$ eine offene Umgebung von $F(p)$ ist und die auf U eingeschränkte Abbildung $F_U : U \rightarrow V$ ein Diffeomorphismus und damit invertierbar ist.*

In diesem Kapitel wurde die differenzierbare Mannigfaltigkeit und einige Begriffe, wie der Tangentialraum und die Vektorfelder, eingeführt, die für die Theorie der Lie-Gruppen, die im nächsten Kapitel näher erläutert wird, von großer Bedeutung sind. Es wurden jeweils zwei Definitionen für Tangentialvektoren an Mannigfaltigkeiten und für Vektorfelder auf Mannigfaltigkeiten präsentiert. Definiert wurden der Tangentialraum an einem Punkt der Mannigfaltigkeit, der ein Vektorraum ist, und das Differential einer Abbildung zwischen Mannigfaltigkeiten.

Kapitel 4

Einführung in die Theorie der Lie-Gruppen

In diesem Kapitel werden ausgesuchte Begriffe aus der Theorie der Lie-Gruppen präsentiert, die für die Einführung von Optimierungsverfahren auf Lie-Gruppen benötigt werden. Das im nächsten Kapitel betrachtete Optimierungsverfahren auf Lie-Gruppen bestimmt, ausgehend von einem Startwert, in jedem Iterationsschritt eine Suchrichtung und berechnet aus dieser und der aktuellen Iterierten einen neuen Kandidaten für einen Optimierer. Bei der Optimierung auf Lie-Gruppen ist nicht sichergestellt, dass die neue Iterierte wieder in der Lie-Gruppe liegt, da die Suchrichtung im Allgemeinen nicht Element der Lie-Gruppe ist. Die Suchrichtung ist ein Element der sogenannten Lie-Algebra der Lie-Gruppe.

Eine bestimmte Menge von Vektorfeldern, die linksinvarianten Vektorfelder, bildet die Lie-Algebra eine Lie-Gruppe. Es wird gezeigt, dass der Tangentialraum am neutralen Element der Lie-Gruppe isomorph zu der Menge der linksinvarianten Vektorfelder ist und somit mit der Lie-Algebra der Lie-Gruppe identifiziert werden kann.

Schließlich wird die Exponentialabbildung definiert, die einen Zusammenhang herstellt zwischen der Lie-Gruppe und ihrer Lie-Algebra und damit eine Verknüpfung liefert für die aktuelle Iterierte und der Suchrichtung im Optimierungsverfahren.

Die Begriffe aus diesem Kapitel stammen überwiegend aus den folgenden Werken: [34, 54, 20, 26, 29, 21, 45].

4.1 Lie-Gruppen und Lie-Algebren

Definition 4.1 (Lie-Gruppe, siehe [54, 26, 21, 45]) Eine Lie-Gruppe \mathcal{G} ist eine differenzierbare Mannigfaltigkeit und gleichzeitig eine Gruppe, so dass die Gruppenverknüpfung

$$(\sigma, \tau) \mapsto \sigma * \tau : \mathcal{G} \times \mathcal{G} \rightarrow \mathcal{G}$$

und die Inversenbildung

$$\tau \mapsto \tau^{-1} : \mathcal{G} \rightarrow \mathcal{G}$$

differenzierbare Diffeomorphismen sind.

Da eine Lie-Gruppe eine differenzierbare Mannigfaltigkeit ist, können die Begriffe des Tangentialvektors, des Tangentialraums, des Vektorfeldes und des Differential auf die Lie-Gruppe übertragen werden. Zusätzlich hat die Lie-Gruppe eine Gruppenstruktur:

Definition 4.2 (Gruppe) Eine Gruppe $(G, *)$ ist eine Menge G mit einer Verknüpfung $* : G \times G \rightarrow G$, $(\sigma, \tau) \mapsto \sigma * \tau$, die folgende Bedingungen erfüllt:

- (i) *Assoziativität:* $\sigma, \tau, \rho \in G : (\sigma * \tau) * \rho = \sigma * (\tau * \rho)$,
- (ii) *Existenz eines neutralen Elements:* $\exists \varepsilon \in G$, so dass $\forall \sigma \in G : \sigma * \varepsilon = \varepsilon * \sigma = \sigma$,
- (iii) *Existenz eines inversen Elements:* $\forall \sigma \in G \exists \sigma^{-1} \in G$, so dass $\sigma * \sigma^{-1} = \sigma^{-1} * \sigma = \varepsilon$.

Da die Lie-Gruppe \mathcal{G} eine Gruppe ist, können Elemente von \mathcal{G} verknüpft werden, es existiert ein neutrales Element $\varepsilon \in \mathcal{G}$ und für jedes Element aus \mathcal{G} ein Inverses.

Zur Veranschaulichung folgen hier zwei Beispiele für Lie-Gruppen:

- Die reellen Zahlen \mathbb{R} und der euklidische Raum \mathbb{R}^n bilden Lie-Gruppen unter der Addition.
- Die *general linear group* $GL(n, \mathbb{R})$ ist eine n^2 -dimensionale differenzierbare Mannigfaltigkeit (vgl. Kapitel 3) und bildet mit der Matrizenmultiplikation eine Gruppe. Da die Multiplikation und die Invertierung von Matrizen rationale Funktionen in deren Einträgen sind, sind diese differenzierbar [26].

In den folgenden Kapiteln wird ein Optimierungsverfahren auf Lie-Gruppen betrachtet, das zu der Klasse der Newton-Methoden gehört. Die Suchrichtung dieses Verfahrens liegt bei der Optimierung auf Lie-Gruppen in der sogenannten Lie-Algebra der Lie-Gruppe.

Definition 4.3 (Lie-Algebra, Lie-Klammer, siehe [20, 26, 45]) Eine Lie-Algebra \mathfrak{g} ist ein \mathbb{R} -Vektorraum mit einer Abbildung

$$[\cdot, \cdot] : \mathfrak{g} \times \mathfrak{g} \rightarrow \mathfrak{g},$$

die folgende Eigenschaften für alle $X, Y, Z \in \mathfrak{g}$ und für alle $a, b \in \mathbb{R}$ erfüllt:

- (i) $[aX + bY, Z] = a[X, Z] + b[Y, Z]$,
 $[Z, aX + bY] = a[Z, X] + b[Z, Y]$ (Bilinearität),
- (ii) $[X, Y] = -[Y, X]$ (Antisymmetrie),
- (iii) $[[X, Y], Z] + [[Y, Z], X] + [[Z, X], Y] = 0$ (Jacobi-Identität).

Die Abbildung $[\cdot, \cdot]$ heißt Lie-Klammer.

Beispiele für Lie-Algebren mit Lie-Klammern sind:

1. Der euklidische \mathbb{R}^3 mit dem Kreuzprodukt $[x, y] = x \times y$ als Lie-Klammer ist eine 3-dimensionale Lie-Algebra [20].
2. Der Vektorraum $M(n, \mathbb{R})$ der reellen $n \times n$ Matrizen ist eine Lie-Algebra mit folgender Lie-Klammer [26, 45]:

$$[A, B] = AB - BA.$$

3. Der Tangentialraum am neutralen Element einer differenzierbaren Mannigfaltigkeit ist eine Lie-Algebra mit dem Vektorfeld-Kommutator als Lie-Klammer [20, 26, 45].
4. Ist \mathcal{G} eine Lie-Gruppe, dann ist die Menge der linksinvarianten Vektorfelder auf \mathcal{G} (s.u.) eine Lie-Algebra mit dem Vektorfeld-Kommutator als Lie-Klammer [20, 26, 45].

Die beiden letztgenannten Beispiele sind die wichtigsten Lie-Algebren für die Betrachtungen in dieser Arbeit. Sie werden beide als die *Lie-Algebra der Lie-Gruppe* bezeichnet. Im Folgenden

wird die Menge der linksinvarianten Vektorfelder auf Lie-Gruppen eingeführt. Ferner wird gezeigt, dass diese mit dem Vektorfeld-Kommutator als Lie-Klammer eine Lie-Algebra bildet und dass der Tangentialraum $T_\varepsilon\mathcal{G}$ am neutralen Element $\varepsilon \in \mathcal{G}$ ebenfalls eine Lie-Algebra ist. Diese beiden Lie-Algebren sind isomorph zueinander und können somit miteinander identifiziert werden.

Linksinvariante Vektorfelder Die Funktion L_σ bezeichne für ein $\sigma \in \mathcal{G}$ die Linksverknüpfung zweier Elemente aus der Lie-Gruppe:

$$L_\sigma : \mathcal{G} \rightarrow \mathcal{G} \quad \text{mit} \quad L_\sigma(\tau) = \sigma * \tau.$$

Das Differential DL_σ (für das Differential vgl. Def. 3.12) der Linksverknüpfung L_σ kann als Linkstranslation interpretiert werden und

$$(DL_\sigma)_\tau : T_\tau\mathcal{G} \rightarrow T_{\sigma*\tau}\mathcal{G}.$$

Mit der Linksverknüpfung kann man eine besondere Menge von Vektorfeldern definieren, die linksinvarianten Vektorfelder:

Definition 4.4 (linksinvariante Vektorfelder, siehe [54, 26, 45]) *Ein glattes Vektorfeld $X \in \mathfrak{X}(\mathcal{G})$ ($X : \mathcal{G} \rightarrow T\mathcal{G}$) heißt linksinvariant, falls $\forall \sigma, \tau \in \mathcal{G}$ gilt*

$$X(\sigma * \tau) = (DL_\sigma)_\tau(X(\tau)),$$

wobei $X(\tau) \in T_\tau\mathcal{G}$. Die Menge der linksinvarianten glatten Vektorfelder wird mit $\mathfrak{X}^*(\mathcal{G})$ bezeichnet.

Die Interpretation dieser sehr formalen Definition ist folgende: Erzeugt man einen Tangentialvektor $X(\tau) \in T_\tau\mathcal{G}$ durch ein linksinvariantes Vektorfeld X , transliert ihn durch das Differential der Linksverknüpfung in den Tangentialraum von $T_{\sigma*\tau}\mathcal{G}$ (das Differential $(DL_\sigma)_\tau$ bildet von $T_\tau\mathcal{G}$ nach $T_{\sigma*\tau}\mathcal{G}$ ab), dann ist $X(\sigma * \tau)$ der so erzeugte Tangentialvektor, d.h. der Vektor, auf den X das Element $\sigma * \tau$ abbildet (vgl. dazu Abb. 4.1).

Jedes linksinvariante Vektorfeld ist eindeutig bestimmt durch seinen Wert am neutralen Element $\varepsilon \in \mathcal{G}$. Dies bedeutet, dass für $X, Y \in \mathfrak{X}^*(\mathcal{G})$ mit $X(\varepsilon) = Y(\varepsilon)$ gilt $X = Y$. Die Gleichheit

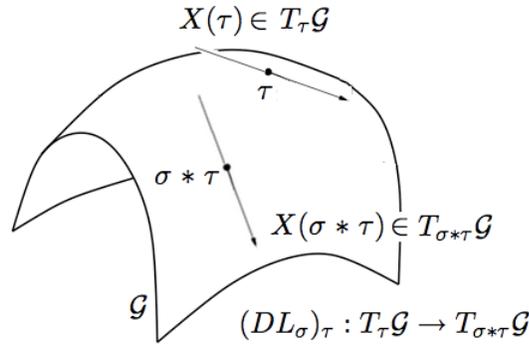


Abbildung 4.1: Betrachtet man ein linksinvariantes Vektorfeld X , so ist $X(\sigma * \tau)$ derselbe Tangentialvektor, den man erhält, wenn man $X(\tau) \in T_\tau \mathcal{G}$ durch das Differential der Linksverknüpfung $(DL_\sigma)_\tau$ auf den Tangentialraum $T_{\sigma * \tau} \mathcal{G}$ abbildet (Abb. aus [1]).

$X = Y$ folgt aus [45]:

$$\begin{aligned}
 X(\sigma) &= X(\sigma * \varepsilon) = (DL_\sigma)_\varepsilon(X(\varepsilon)), & \text{da } X \in \mathfrak{X}^*(\mathcal{G}) \\
 &= (DL_\sigma)_\varepsilon(Y(\varepsilon)), & \text{da } X(\varepsilon) = Y(\varepsilon) \\
 &= Y(\sigma * \varepsilon), & \text{da } Y \in \mathfrak{X}^*(\mathcal{G}) \\
 &= Y(\sigma).
 \end{aligned}$$

Diese Eigenschaft von linksinvarianten Vektorfeldern wird später noch benutzt, um die Menge $\mathfrak{X}^*(\mathcal{G})$ mit dem Tangentialraum $T_\varepsilon \mathcal{G}$ am neutralen Element zu identifizieren.

Die Lie-Algebra der LieGruppe Die linksinvarianten glatten Vektorfelder auf \mathcal{G} bilden einen Vektorraum [26]. Damit sie auch eine Lie-Algebra bilden, wird eine Abbildung benötigt, die die Eigenschaften der Lie-Klammer (vgl. Def. 4.3) erfüllt. Diese Abbildung ist der Vektorfeld-Kommutator (vgl. Def. 3.1).

Satz 4.1 Die Menge der linksinvarianten Vektorfelder bildet eine Lie-Algebra mit dem Vektorfeld-Kommutator als Lie-Klammer.

Beweis: Es ist bereits bekannt, dass $\mathfrak{X}^*(\mathcal{G})$ einen Vektorraum bildet. Zu zeigen ist noch, dass der Vektorfeld-Kommutator eine Abbildung der Form $[\cdot, \cdot] : \mathfrak{X}^*(\mathcal{G}) \times \mathfrak{X}^*(\mathcal{G}) \rightarrow \mathfrak{X}^*(\mathcal{G})$ ist (Teil (a)) und die Eigenschaften einer Lie-Klammer (Def. 4.3) erfüllt (Teil (b)).

Teil (a): Da nach der Definition für linksinvariante Vektorfelder $X, Y \in \mathfrak{X}^*(\mathcal{G})$ (Def. 4.4) gilt,

dass $DL_\sigma(X) = X$ und $DL_\sigma(Y) = Y$ und da die Funktion $L_\sigma : \mathcal{G} \rightarrow \mathcal{G}$ ein Diffeomorphismus ist, gilt (vgl. Gleichung (3.3)):

$$DL_\sigma([X, Y]) = [DL_\sigma(X), DL_\sigma(Y)] = [X, Y].$$

Damit ist auch die Verknüpfung zweier linksinvarianter Vektorfelder mit dem Vektorfeld-Kommutator ein linksinvariantes Vektorfeld: $[X, Y] \in \mathfrak{X}^*(\mathcal{G})$.

Teil (b): Der Vektorfeld-Kommutator erfüllt die Eigenschaften der Lie-Klammer (Bilinearität, Antisymmetrie, Jacobi-Identität):

Seien $a, b \in \mathbb{R}$ und $X, Y, Z \in \mathfrak{X}^*(\mathcal{G})$. Es gilt:

(i) Bilinearität:

$$\begin{aligned} [aX + bY, Z] &= (aX + bY)(Z) - Z(aX + bY) \\ &= aX(Z) + bY(Z) - Z(aX) - Z(bY) \\ &= aX(Z) - aZ(X) + bY(Z) - bZ(Y) \\ &= a[X, Z] + b[Y, Z]. \end{aligned}$$

Analog für $[Z, aX + bY] = a[Z, X] + b[Z, Y]$.

(ii) Antisymmetrie:

$$\begin{aligned} [X, Y] &= X(Y) - Y(X) \\ &= -(Y(X) - X(Y)) \\ &= -[Y, X]. \end{aligned}$$

(iii) Jacobi-Identität:

$$\begin{aligned} [[X, Y], Z] &= [X(Y) - Y(X), Z] \\ &= [X(Y), Z] - [Y(X), Z] \\ &= X(Y(Z)) - Z(X(Y)) - Y(X(Z)) + Z(Y(X)). \\ [[Y, Z], X] &= [Y(Z) - Z(Y), X] \\ &= [Y(Z), X] - [Z(Y), X] \\ &= Y(Z(X)) - X(Y(Z)) - Z(Y(X)) + X(Z(Y)). \\ [[Z, X], Y] &= [Z(X) - X(Z), Y] \\ &= [Z(X), Y] - [X(Z), Y] \\ &= Z(X(Y)) - Y(Z(X)) - X(Z(Y)) + Y(X(Z)). \end{aligned}$$

Da alle Summanden sich wegheben, ist

$$[[X, Y], Z] + [[Y, Z], X] + [[Z, X], Y] = 0.$$

□

Vergleiche für den Beweis auch [26, 45]. Mit dem Vektorfeld-Kommutator werden zwei linksinvariante Vektorfelder miteinander verknüpft. Da linksinvariante Vektorfelder eindeutig bestimmt sind durch ihren Wert am neutralen Element der Lie-Gruppe \mathcal{G} , kann eine Abbildung definiert werden, die zwei Elemente des Tangentialraums $T_\varepsilon\mathcal{G}$ miteinander verknüpft. Seien $X, Y \in \mathfrak{X}^*(\mathcal{G})$ mit $X(\varepsilon) = \xi_\varepsilon \in T_\varepsilon\mathcal{G}$ und $Y(\varepsilon) = \eta_\varepsilon \in T_\varepsilon\mathcal{G}$. Die Verknüpfung von ξ_ε und η_ε wird definiert durch:

$$[\xi_\varepsilon, \eta_\varepsilon] = [X(\varepsilon), Y(\varepsilon)] := [X, Y](\varepsilon), \quad (4.1)$$

wobei die Klammer auf der rechten Seite der Kommutator für Vektorfelder ist.

Da die Verknüpfung (4.1) über den Vektorfeld-Kommutator definiert wurde und dieser die Eigenschaften einer Lie-Klammer erfüllt, erfüllt auch die Verknüpfung (4.1) die Eigenschaften der Lie-Klammer. Da der Tangentialraum $T_\varepsilon\mathcal{G}$ ein Vektorraum ist (vgl. Kapitel 3), bildet $T_\varepsilon\mathcal{G}$ zusammen mit der Verknüpfung (4.1) eine Lie-Algebra mit Lie-Klammer.

Der folgende Satz zeigt, dass die Lie-Algebren $\mathfrak{X}^*(\mathcal{G})$ und $T_\varepsilon\mathcal{G}$ isomorph zueinander sind. Zwei Mengen heißen isomorph zueinander, wenn ein Isomorphismus zwischen den Mengen existiert. Damit ist es egal, ob der Raum der linksinvarianten Vektorfelder oder der Tangentialraum am neutralen Element betrachtet wird.

Satz 4.2 *Die Menge der linksinvarianten Vektorfelder $\mathfrak{X}^*(\mathcal{G})$ ist isomorph zum Tangentialraum $T_\varepsilon\mathcal{G}$ am neutralen Element $\varepsilon \in \mathcal{G}$ der Lie-Gruppe \mathcal{G} .*

Beweis: Wenn $\mathfrak{X}^*(\mathcal{G})$ und $T_\varepsilon\mathcal{G}$ isomorph zueinander sind, gibt es einen Isomorphismus (einen bijektiven Homomorphismus, vgl. Def. 3.3) zwischen ihnen. Sei F eine Abbildung mit

$$\begin{aligned} F : \mathfrak{X}^*(\mathcal{G}) &\rightarrow T_\varepsilon\mathcal{G} \\ X &\mapsto X(\varepsilon). \end{aligned}$$

Nun ist zu zeigen, dass F ein Homomorphismus (Teil (a)) und bijektiv (Teil (b)) ist.

Teil (a): Es gilt $\forall X, Y \in \mathfrak{X}^*(\mathcal{G}), \forall \alpha \in \mathbb{R}$:

$$F(X + Y) = (X + Y)(\varepsilon) = X(\varepsilon) + Y(\varepsilon) = F(X) + F(Y),$$

$$F(\alpha X) = (\alpha X)(\varepsilon) = \alpha X(\varepsilon) = \alpha F(X).$$

$\Rightarrow F$ ist ein Homomorphismus.

Teil (b): Injektivität: Angenommen $F(X) = F(Y)$. Daraus folgt, dass $X(\varepsilon) = Y(\varepsilon)$. Dieses gilt nur, wenn $X = Y$, da linksinvariante Vektorfelder eindeutig durch ihren Wert am neutralen Element bestimmt sind. Also ist F injektiv.

Surjektivität: Sei $v \in T_\varepsilon \mathcal{G}$ und $X(\sigma) = (DL_\sigma)_\varepsilon(v)$ ein Vektorfeld mit $\sigma \in \mathcal{G}$. Zu zeigen ist, dass $X \in \mathfrak{X}^*(\mathcal{G})$, also dass $X(\tau * \sigma) = (DL_\tau)_\sigma(X(\sigma))$. Dann kann jedes Element von $T_\varepsilon \mathcal{G}$ durch ein linksinvariantes Vektorfeld dargestellt werden. Sei $\tau \in \mathcal{G}$, dann gilt $L_{\tau * \sigma} = L_\tau \circ L_\sigma$. Damit gilt

$$\begin{aligned} (DL_\tau)_\sigma(X(\sigma)) &= (DL_\tau)_\sigma((DL_\sigma)_\varepsilon(v)) \\ &= (DL_{\tau * \sigma})_\varepsilon(v) \\ &= X(\tau * \sigma). \end{aligned}$$

Der Beweis ist auch nachzulesen in [26, 45].

$\Rightarrow X \in \mathfrak{X}^*(\mathcal{G})$ und die Umkehrung von F existiert.

$\Rightarrow F$ ist ein Isomorphismus und $\mathfrak{X}^*(\mathcal{G})$ und $T_\varepsilon \mathcal{G}$ sind isomorph zueinander. \square

Da $\mathfrak{X}^*(\mathcal{G})$ und $T_\varepsilon \mathcal{G}$ isomorph zueinander sind, werden sie im Folgenden beide mit \mathfrak{g} bezeichnet und heißen *die Lie-Algebra \mathfrak{g} der Lie-Gruppe \mathcal{G}* .

Damit gibt es zwei Definitionen der Lie-Algebra einer Lie-Gruppe:

1. \mathfrak{g} als Raum der linksinvarianten Vektorfelder $\mathfrak{X}^*(\mathcal{G})$ mit dem Vektorfeld-Kommutator (Def. 3.1) als Lie-Klammer,
2. \mathfrak{g} als Tangentialraum $T_\varepsilon \mathcal{G}$ von \mathcal{G} im neutralen Element $\varepsilon \in \mathcal{G}$ mit der Verknüpfung (4.1) als Lie-Klammer.

Da die beiden Lie-Algebren isomorph zueinander sind, ist es egal, welche Definition verwendet wird.

4.2 Die Exponentialabbildung

Im vorherigen Abschnitt wurde die Lie-Algebra einer Lie-Gruppe beschrieben. Sie ist wichtig für das Optimierungsverfahren auf Lie-Gruppen, das in den weiteren Kapiteln behandelt wird, da die Suchrichtung dieses Verfahrens in der Lie-Algebra liegt. Bei Optimierungsverfahren auf Vektorräumen wird im jeweiligen Iterationsschritt der neue Kandidat für einen Minimierer durch eine Suchrichtung bestimmt. Diese wird mit der aktuellen Iterierten durch Addition verknüpft. Auf Lie-Gruppen führt dieses Vorgehen zu Problemen. Zum einen kann man im Allgemeinen die Elemente der Lie-Gruppe und der Lie-Algebra nicht addieren. Zum anderen ist nicht sichergestellt, dass die Verknüpfung eines Elementes der Lie-Algebra mit einem Element der Lie-Gruppe wieder in der Lie-Gruppe liegt.

Ein gängiges Hilfsmittel, um eine Beziehung zwischen den Elementen der Lie-Gruppe und der Lie-Algebra herzustellen, ist die Exponentialabbildung [1]. Sie bildet von der Lie-Algebra in die Lie-Gruppe ab und liefert damit neben anderen Eigenschaften eine Verknüpfung für Elemente der Lie-Algebra mit den Elementen der Lie-Gruppe.

Dieser Abschnitt gibt einen kurzen Einblick in die Exponentialabbildung. Es werden die Eigenschaften beschrieben, die für die Optimierung auf Lie-Gruppen von Bedeutung sind. Für eine detailliertere Einführung in die Exponentialabbildung siehe [1, 26, 21, 45].

Die Exponentialabbildung wird über eine Integralkurve und als Lösung eines Anfangswertproblems beschrieben. Eine Integralkurve eines Vektorfeldes ist eine stetig differenzierbare Kurve, deren Tangentialvektoren in allen Punkten mit dem jeweiligen Wert des Vektorfeldes übereinstimmen. Die Menge aller Integralkurven eines Vektorfeldes bildet den Fluss dieses Vektorfeldes. Damit kann man sich die Bewegung eines Teilchens auf einer Lie-Gruppe (oder allgemein auf einer Mannigfaltigkeit) vorstellen. Ist ein Vektorfeld $X \in \mathfrak{X}^*(\mathcal{G})$ gegeben, dann bewegt sich das Teilchen in einem Punkt $\sigma \in \mathcal{G}$ mit der Geschwindigkeit $X(\sigma)$ (vgl. Abb. 4.2).

Definition 4.5 (Integralkurve, siehe [26]) Sei X ein Vektorfeld auf \mathcal{G} und $\sigma \in \mathcal{G}$. Eine stetig differenzierbare Kurve $\gamma : I \subset \mathbb{R} \rightarrow \mathcal{G}$ heißt Integralkurve von X durch σ , wenn

$$\begin{aligned}\gamma(t_0) &= \sigma, \quad t_0 \in I, \\ \dot{\gamma}(t) &= X(\gamma(t)), \quad \forall t \in I.\end{aligned}$$

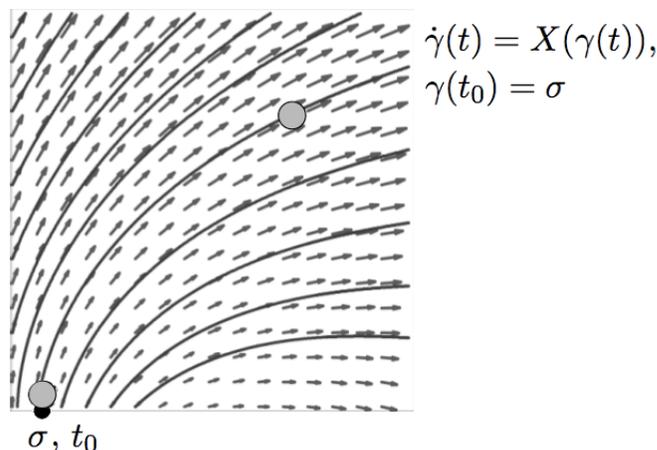


Abbildung 4.2: Fluss eines Teilchens (dargestellt durch die grauen Kreise), gestartet im Punkt σ . Der Fluss ist die Integralkurve γ vom Vektorfeld X (dargestellt durch die Pfeile) durch den Punkt σ . Startet das Teilchen in einem anderen Punkt, ist die Integralkurve eine andere (Abb. aus [48]).

Die Integralkurve heißt maximale Integralkurve, wenn der Definitionsbereich I maximal ist. Der Punkt $\sigma \in \mathcal{G}$ heißt Startpunkt von γ , falls $t_0 = 0$.

Mit anderen Worten, der Tangentialvektor von γ entspricht an jedem Punkt dem von X gegebenen Tangentialvektor an diesem Punkt. Es kann gezeigt werden, dass jedes glatte Vektorfeld in jedem Punkt eine eindeutige maximale Integralkurve bestimmt [26].

Die Integralkurven von linksinvarianten Vektorfeldern bilden sogenannte Ein-Parameter-Untergruppen [26, 21], d.h. Lie-Gruppen-Homomorphismen, die von den reellen Zahlen in die Lie-Gruppe \mathcal{G} abbilden [26]:

Definition 4.6 (Ein-Parameter-Untergruppe, siehe [26, 21]) Sei $(\mathcal{G}, *)$ eine Lie-Gruppe mit neutralem Element $\varepsilon \in \mathcal{G}$. Eine Familie von Elementen $(\gamma(t))_{t \in \mathbb{R}}$ von \mathcal{G} heißt Ein-Parameter-Untergruppe von \mathcal{G} genau dann, wenn

1. $\gamma(0) = \varepsilon$ und
2. $\gamma(t_1) * \gamma(t_2) = \gamma(t_1 + t_2)$.

Mit Hilfe der Integralkurven kann nun die Exponentialabbildung definiert werden, die eine stetig differenzierbare Abbildung von der Lie-Algebra in die Lie-Gruppe ist.

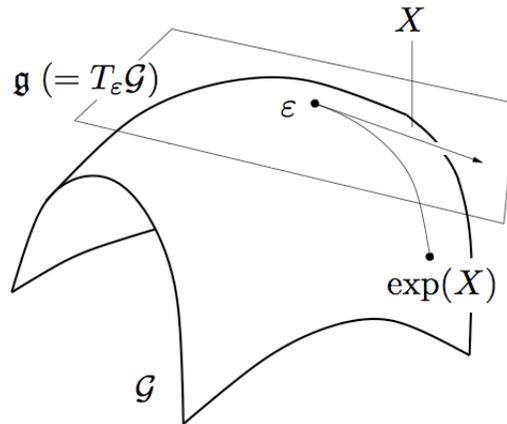


Abbildung 4.3: Die Exponentialabbildung $\exp : \mathfrak{g} \rightarrow \mathcal{G}$ bildet ein Element der Lie-Algebra auf ein Element der Lie-Gruppe ab. Das Element $\exp(X)$ ist das Element, das man erhält, wenn man dem Fluss des Vektorfeldes X für eine Zeiteinheit lang folgt (Abb. aus [1]).

Definition 4.7 (Exponentialabbildung, siehe [26, 21]) Sei $X \in \mathfrak{g}$ (X ist ein linksinvariantes Vektorfeld), $\varepsilon \in \mathcal{G}$ das neutrale Element der Lie-Gruppe \mathcal{G} und $\gamma(t)$ die Integralkurve von X mit dem Startpunkt ε , d.h.

$$\begin{aligned}\dot{\gamma}(t) &= X(\gamma(t)), \\ \gamma(0) &= \varepsilon.\end{aligned}$$

Es wird $\exp(X) := \gamma(1)$ gesetzt. Die Abbildung

$$\begin{aligned}\exp : \mathfrak{g} &\rightarrow \mathcal{G} \\ X &\mapsto \exp(X)\end{aligned}$$

heißt Exponentialabbildung.

Aus der Eindeutigkeit der Integralkurven folgt auch die Eindeutigkeit der Exponentialabbildung [26].

Für ein $X \in \mathfrak{g}$ ist $\exp(X) \in \mathcal{G}$ das Element, das man erhält, wenn man im neutralen Element von \mathcal{G} dem Fluss des Feldes X eine Zeiteinheit lang folgt (vgl. Abb. 4.3).

Der Satz 4.3 liefert einige wichtige Eigenschaften der Exponentialabbildung:

Satz 4.3 Sei \mathcal{G} eine Lie-Gruppe und \mathfrak{g} die zugehörige Lie-Algebra mit den neutralen Elementen $\varepsilon \in \mathcal{G}$ und $\mathbf{0} \in \mathfrak{g}$. Dann gilt:

- (i) Die Exponentialabbildung $\exp : \mathfrak{g} \rightarrow \mathcal{G}$ ist stetig differenzierbar.
- (ii) Für jedes $X \in \mathfrak{g}$ ist $\gamma(t) = \exp(tX)$ die Ein-Parameter-Untergruppe, die von X erzeugt wird.
- (iii) Für $X \in \mathfrak{g}$ gilt $\exp((s+t)X) = \exp(sX) * \exp(tX)$.
- (iv) Das Differential der Exponentialabbildung $D \exp : T_0 \mathfrak{g} \rightarrow T_\varepsilon \mathcal{G}$ ist die Identitätsabbildung.
- (v) Die Exponentialabbildung ist in einer offenen Umgebung von $0 \in \mathfrak{g}$ ein Diffeomorphismus in eine offene Umgebung von $\varepsilon \in \mathcal{G}$.

Beweis: (Vgl. dazu [26])

(i): Sei γ_X^ε die Integralkurve eines Vektorfeldes $X \in \mathfrak{X}^*(\mathcal{G})$ mit dem Startpunkt ε . Um (i) zu beweisen, muss gezeigt werden, dass $\gamma_X^\varepsilon(1)$ in X stetig differenzierbar ist. Dieses kann mit einem Umweg über Produktmannigfaltigkeiten gezeigt werden, was in [26] nachzulesen ist.

(ii): Die Ein-Parameter-Untergruppe, die von X erzeugt wird, ist die Integralkurve von X mit dem Startpunkt ε . Um (ii) zu beweisen, muss gezeigt werden, dass $\exp(tX) = \gamma_X^\varepsilon(t)$, also dass

$$\gamma_{tX}^\varepsilon(1) = \gamma_X^\varepsilon(t). \quad (4.2)$$

Es wird sogar gezeigt, dass für alle $s, t \in \mathbb{R}$ gilt

$$\gamma_{tX}^\varepsilon(s) = \gamma_X^\varepsilon(st),$$

woraus sich natürlich (4.2) ergibt.

Sei $t \in \mathbb{R}$ beliebig, aber fest gewählt und sei $c : \mathbb{R} \rightarrow \mathcal{G}$ eine stetig differenzierbare Kurve mit

$$c(s) = \gamma_X^\varepsilon(st).$$

Mit der Kettenregel gilt:

$$\dot{c}(s) = t \dot{\gamma}_X^\varepsilon(st) = tX(\gamma_X^\varepsilon(st)) = tX(c(s)).$$

Deswegen und weil $c(0) = \gamma_X^\varepsilon(0) = \varepsilon$, ist c eine Integralkurve des Vektorfeldes tX mit dem Startpunkt ε . Wegen der Eindeutigkeit von Integralkurven ist damit $c(s) = \gamma_{tX}^\varepsilon(s)$ und (ii) ist bewiesen.

(iii): Dies folgt direkt aus den Eigenschaften der Ein-Parameter-Untergruppen (Def. 4.6) und insbesondere daraus, dass Ein-Parameter-Untergruppen Gruppen-Homomorphismen sind.

(iv): Das Differential bildet von dem Tangentialraum des Definitionsbereiches in den Tangentialraum des Zielbereiches ab. Der Definitionsbereich von der Exponentialabbildung ist \mathfrak{g} und da

\mathfrak{g} ein Vektorraum ist, gilt $T_{\mathbf{0}}\mathfrak{g} = \mathfrak{g}$. Also ist $D \exp : \mathfrak{g} \rightarrow \mathfrak{g}$ eine Abbildung von der Lie-Algebra in die Lie-Algebra.

Sei jetzt $X \in \mathfrak{g}$ beliebig und $\eta : \mathbb{R} \rightarrow \mathfrak{g}$ eine Kurve mit $\eta(t) = tX$. Dann ist $\dot{\eta}(0) = X$ und mit (ii) folgt:

$$D \exp(X) = D \exp(\dot{\eta}(0)) = (\exp \circ \eta)'(0) = \left. \frac{d}{dt} \exp(tX) \right|_{t=0} = X.$$

(v): Nach dem Satz der lokalen Umkehrbarkeit (Satz 3.2) ist eine differenzierbare Abbildung mit invertierbarem Differential lokal ein Diffeomorphismus. Das Differential $D \exp(X) = X$ ist die Identitätsabbildung und damit offensichtlich ein Isomorphismus. Damit gilt die Behauptung. \square

Die wichtigsten Aussagen des Satzes sind, dass die Exponentialabbildung lokal ein Diffeomorphismus von einer offenen Umgebung von $\mathbf{0} \in \mathfrak{g}$ in eine offene Umgebung $\varepsilon \in \mathcal{G}$ ist und dass $D \exp(X) = X$. Damit kann nun ein Element der Lie-Algebra auf sinnvolle Weise auf ein Element der Lie-Gruppe abgebildet werden. Anschaulich wird sozusagen ein Lot von der Lie-Algebra auf die Lie-Gruppe gefällt.

Betrachtet man Matrix-Lie-Gruppen (z.B. $GL(n, \mathbb{R})$, siehe oben), ist die Exponentialabbildung für $t \in \mathbb{R}$ und $X \in \mathfrak{g} \subset \mathbb{R}^{n \times n}$ gegeben durch das Matrix-Exponential [20, 26]:

$$\exp(tX) = \sum_{k=0}^{\infty} \frac{t^k}{k!} X^k. \quad (4.3)$$

Ableitungen von Funktionen auf differenzierbaren Mannigfaltigkeiten Sei $f \in C^\infty(\mathcal{G})$ eine Funktion auf der Lie-Gruppe. Für die Optimierungsverfahren auf Lie-Gruppen, die im nächsten Kapitel beschrieben werden, wird eine Taylorentwicklung von f benötigt.

Die Ableitungen von f sind abhängig von der speziellen Wahl der Karte (vgl. (3.1)). In [29] wurde dafür eine Karte vorgestellt. Sei \mathcal{G} eine Lie-Gruppe, \mathfrak{g} die zugehörige Lie-Algebra und $\{e_1, \dots, e_n\}$ eine Basis von \mathfrak{g} . Jedes Element $u \in \mathfrak{g}$ kann dargestellt werden durch $u = \sum_{i=1}^n u_i e_i$. Auf die spezielle Wahl der Basis und wie diese die Ableitungen von f beeinflusst, wird in dieser Arbeit nicht näher eingegangen.

Sei (\mathcal{U}, φ) eine Karte von \mathcal{G} mit [29]

$$\varphi(\sigma * \exp(u)) = (u_1, \dots, u_n)^T = u.$$

Damit ist $\varphi^{-1}(u) = \sigma * \exp(u)$. Für die partiellen Ableitungen der Interpretation von f bezüglich dieser Karte gilt

$$\frac{\partial}{\partial u_i} (f \circ \varphi^{-1}) \Big|_{u=\varphi(\sigma)} = \frac{\partial}{\partial u_i} f(\sigma * \exp(u)) \Big|_{u=\varphi(p)} .$$

Die Koordinaten u_1, \dots, u_n können benutzt werden, um eine Taylorentwicklung zweiter Ordnung anzugeben [47, 49, 29, 45]:

$$f(\sigma * \exp(u)) = f(\sigma) + df_\sigma(0)u + \frac{1}{2}u^T H_\sigma^f(0)u + \mathcal{O}(\|u\|^3) \quad (4.4)$$

mit $df_\sigma(0) \in \mathbb{R}^{1 \times n}$, $H_\sigma^f(0) \in \mathbb{R}^{n \times n}$ und

$$(df_\sigma(0))_i = \frac{\partial}{\partial u_i} f(\sigma * \exp(u)) \Big|_{u=0} \quad \text{und} \\ (H_\sigma^f(0))_{i,j} = \frac{\partial}{\partial u_i \partial u_j} f(\sigma * \exp(u)) \Big|_{u=0} .$$

Die Ableitung der Funktion f ist abhängig von der Wahl der Karte φ . Inwiefern die Wahl der Karte φ die Ableitungen und damit die Taylorformel beeinflusst, wurde nicht weiter untersucht.

In diesem Kapitel wurde der Begriff der Lie-Gruppe und der Lie-Algebra einer Lie-Gruppe eingeführt. Die Exponentialabbildung liefert eine Beziehung zwischen der Lie-Algebra und der Lie-Gruppe, da sie von der einen in die andere abbildet. Mit Hilfe dieser Begriffe können im nächsten Kapitel Optimierungsmethoden auf Lie-Gruppen vorgestellt werden.

Kapitel 5

Optimierung auf Lie-Gruppen zur Bildregistrierung

Die Bildregistrierungsmethode, die in dieser Arbeit betrachtet wird [49], basiert auf einer Optimierung auf Lie-Gruppen. Mit den in Kapitel 3 und Kapitel 4 eingeführten Begriffen der Differentialgeometrie und der Lie-Gruppen-Theorie können nun Optimierungsverfahren auf Lie-Gruppen beschrieben werden. Für weiterführende Informationen zu Lie-Gruppen-Methoden und Optimierung auf Lie-Gruppen siehe [24, 37].

Für die Optimierung werden in dem Registrierungsrahmen aus [49] Newton-Methoden verwendet. Diese Verfahren minimieren eine Zielfunktion, indem sie ausgehend von einem Startwert und einer Schrittweite iterativ eine Suchrichtung bestimmen und diese mit der aktuellen iterierten verknüpfen.

In diesem Kapitel wird im ersten Teil ein kurzer Überblick über Newton-Methoden auf Vektorräumen gegeben. Im Anschluß daran wird ein spezielles Verfahren, das Gauß-Newton-Verfahren, für die Optimierung auf Lie-Gruppen angepasst [49]. Der wichtigste Unterschied zwischen den hier betrachteten Optimierungsverfahren auf Vektorräumen und auf Lie-Gruppen besteht in der Verknüpfung der Suchrichtung mit der aktuellen Iterierten und der Berechnung der Ableitung. Auf Vektorräumen liegen sowohl die Iterierte als auch die Suchrichtung im Vektorraum, so dass diese durch Addition verknüpft werden können. Auf Lie-Gruppen liegt die Iterierte in der Lie-Gruppe, während die Suchrichtung in der Lie-Algebra der Lie-Gruppe liegt [29]. Die Verknüpfung wird in dem Fall durch die Exponentialabbildung realisiert [1, 29]. Die

Ableitungen von Funktionen werden über Einbettungen der Lie-Gruppe in einen euklidischen Raum berechnet [49, 26].

Im zweiten Teil wird die Bildregistrierung unter Verwendung von Lie-Gruppen vorgestellt. Das Gauß-Newton-Verfahren auf Lie-Gruppen wird dazu benutzt, eine Zielfunktion in der Registrierung zu minimieren. Dieses Verfahren benutzt erste Ableitungen der Zielfunktion, welche mit einer sogenannten Einbettung einer Lie-Gruppe in einen euklidischen Raum berechnet werden können. Anschließend wird die Bildregistrierung unter Verwendung von Lie-Gruppen am Beispiel der rigiden Transformationen erläutert.

5.1 Optimierung auf Lie-Gruppen

Das Gauß-Newton-Verfahren gehört zu den sogenannten Newton-Methoden, mit denen iterativ ein Minimierer einer Zielfunktion bestimmt wird. Ein Minimierer einer Funktion ist der Punkt, der den kleinsten Funktionswert hat (globaler Minimierer) oder der in einer Umgebung den kleinsten Funktionswert hat (lokaler Minimierer). Das wohl bekannteste Verfahren dieser Art ist das Newton-Verfahren auf Vektorräumen, das die ersten und zweiten Ableitungen der Zielfunktion für die Berechnung der Suchrichtung verwendet [36]. Das Gauß-Newton-Verfahren benutzt hingegen Approximationen an die zweiten Ableitungen, um die Suchrichtung zu bestimmen.

Für Newton-Verfahren auf Vektorräumen gilt, dass sie eine Sequenz von Iterierten $\{x_k\}_{k=0}^{\infty}$ bestimmen und dass in jedem Iterationsschritt, ausgehend von einem Startwert $x_0 \in \mathbb{R}^d$, eine Suchrichtung $s_k \in \mathbb{R}^d$, $\nabla f(x_k)^T s_k < 0$ und eine Schrittweite $t_k \in \mathbb{R}$ bestimmt werden. Mit ihnen wird die neue Iterierte $x_{k+1} = x_k + t_k s_k$ berechnet. Der Fokus liegt in dieser Arbeit auf der Berechnung einer geeigneten Suchrichtung. Die Berechnung der Schrittweite t_k wird hier nicht weiter thematisiert. In [49] wird eine Standardwahl $t_k = 1$ benutzt. Verfahren für eine Bestimmung einer effizienten Schrittweite sind z.B. in [36] zu finden. Mit der Wahl $t_k = 1$ wird die neue Iterierte berechnet durch $x_{k+1} = x_k + s_k$.

Das Iterationsverfahren soll terminieren, wenn ein Minimum erreicht wurde oder wenn die aktuelle Iterierte den Minimierer hinreichend approximiert. Für geeignete Abbruchkriterien siehe [17].

Bei der Optimierung auf Lie-Gruppen liegt die aktuelle Iterierte in der Lie-Gruppe und die

Suchrichtung in der Lie-Algebra [1, 29]. Sei \mathcal{G} eine Lie-Gruppe, \mathfrak{g} die Lie-Algebra der Lie-Gruppe \mathcal{G} , $\sigma \in \mathcal{G}$ die aktuelle Iterierte und $u \in \mathfrak{g}$ ($= T_\sigma \mathcal{G}$) die Suchrichtung. Die Addition $\sigma + u$ ist im Allgemeinen nicht definiert bzw. liegt nicht in der Lie-Gruppe. Die Exponentialabbildung stellt jedoch einen Zusammenhang her zwischen der Lie-Algebra und der Lie-Gruppe (vgl. Kapitel 4.2) und der Ausdruck $\sigma * \exp(u) \in \mathcal{G}$ liegt in der Lie-Gruppe. Durch die besonderen Eigenschaften der Exponentialabbildung (vgl. Kapitel 4.2) kann somit

$$\sigma \leftarrow \sigma * \exp(\lambda u)$$

als Update-Schritt der Iterierten verwendet werden [29]. Standardmäßig wird für die Schrittweite $\lambda = 1$ gewählt [49, 47], es können jedoch auch Verfahren zur Schrittweitenbestimmung verwendet werden.

In [29] wurde das Newton-Verfahren auf Lie-Gruppen mit dem eben genannten Update-Schritt beschrieben und in [49] wurde das Gauß-Newton-Verfahren auf die Optimierung auf Lie-Gruppen angepasst:

Sei die Zielfunktion $f \in C^\infty(\mathcal{G})$ die Summe von quadrierten Funktionen

$$f(\sigma) = \frac{1}{2} \|r(\sigma)\|^2 = \frac{1}{2} \sum_{j=1}^m r^j(\sigma)^2.$$

mit $r^j \in C^\infty(\mathcal{G})$ für $j = 1, \dots, m$, der Residuumsfunktion $r : \mathcal{G} \rightarrow \mathbb{R}^n$ und $\sigma \in \mathcal{G}$. Das Gauß-Newton-Verfahren basiert auf einer linearen Approximation der Zielfunktion um die aktuelle Iterierte. Wegen Gleichung (4.4) gilt

$$r(\sigma * \exp(u)) = r(\sigma) + dr_\sigma(0)u + \mathcal{O}(\|u\|_2^2),$$

wobei $(dr_\sigma(0))_i = \left. \frac{\partial}{\partial u_i} r(\sigma * \exp(u)) \right|_{u=0}$. Damit ist

$$f(\sigma * \exp(u)) = \frac{1}{2} \|r(\sigma * \exp(u))\|^2 \approx \frac{1}{2} \|r(\sigma) + dr_\sigma(0)u\|^2. \quad (5.1)$$

Das so entstandene lineare Ausgleichsproblem (least-squares Problem) [36, 18] wird typischerweise mit einer QR-Zerlegung [18] gelöst. In [49] wird die Lösung mit den Normalgleichungen [36] bestimmt: Der Minimierer u^* von (5.1) erfüllt die sogenannte Normalgleichung [36]

$$dr_\sigma(0)^T dr_\sigma(0) \cdot u^* = -dr_\sigma(0)^T \cdot r(\sigma).$$

In jedem Iterationsschritt wird dieser z.B. durch eine QR- oder Cholesky-Zerlegung berechnet [36]. Das Gauß-Newton-Verfahren auf Lie-Gruppen ist in Algorithmus 2 skizziert. Die Berechnung der Jacobi-Matrizen der Residuumsfunktion wird im Folgenden näher erläutert. Da bei

Algorithmus 2 Das Gauß-Newton-Verfahren auf Lie-Gruppen

Eingabe: Funktion $f = \frac{1}{2} \|r\|^2$, Startwert σ_0 **Ausgabe:** Minimierer σ **while** !stop **do** Berechnung von $dr_\sigma(0)$ Suchrichtung u durch Lösung von $dr_\sigma(0)^T dr_\sigma(0) \cdot u = -dr_\sigma(0)^T \cdot r(\sigma)$ $\sigma \leftarrow \sigma * \exp(u)$ **end while**

der Approximation der Zielfunktion eine Taylorentwicklung erster Ordnung verwendet wurde, ist nur eine lineare Konvergenz des Algorithmus zu erwarten. Für mehr Informationen zu Konvergenzaussagen siehe [36]. Für die partiellen Ableitungen wird f nach den Koordinaten u_i abgeleitet. Diese werden bezüglich einer Basis $\{e_1, \dots, e_n\}$ der Lie-Algebra \mathfrak{g} bestimmt (vgl. Kapitel 4). In [29] wird zumindest für das Newton-Verfahren auf Lie-Gruppen gezeigt, dass dieses unabhängig von der Wahl der Basis ist.

5.2 Das Gauß-Newton-Verfahren auf Lie-Gruppen zur Bildregistrierung

In der Bildregistrierung wird ein gegebenes Distanzmaß bezüglich einer Transformation minimiert. Die Eindeutigkeit der Lösung wird entweder durch die Einführung von Regularisierern (nicht-parametrisch) oder durch eine Parametrisierung der Transformationen und einer Optimierung auf dem Parameterraum (parametrisch) erreicht [33, 38] (vgl. Kapitel 2.1). Die in [49, 47] beschriebene Methode folgt einem anderen Ansatz. Einige Klassen von Transformationen, wie z.B. die rigiden, affinen oder die diffeomorphen Transformationen, bilden mit der Komposition \circ von Funktionen als Gruppenverknüpfung eine Lie-Gruppe [49]. Die Strategie der Registrierungsmethode besteht darin, ein Distanzmaß auf einer gegebenen Lie-Gruppe von Transformationen zu minimieren.

Eine Zielfunktion zur Optimierung auf Lie-Gruppen Seien R, T zwei diskrete Bilder, gegeben auf den Punkten $x^1, \dots, x^m \in \mathbb{R}^d$ und y eine Transformation, die ebenfalls nur an den Punkten x^i bekannt ist. Der Suchraum, in dem die optimale Transformation für die Registrie-

ung gesucht wird, wird eingeschränkt, in dem nur in einer bestimmten Klasse von Transformationen gesucht wird. Es wird vorausgesetzt, dass diese Klasse eine Lie-Gruppe bildet, die Lie-Gruppe \mathcal{G} . Beispiele für derartige Transformationsklassen wären die rigiden, affinen oder diffeomorphen Transformationen [49]. Abbildungsmatrizen der affinen Transformationen können z.B. die Gruppen $SE(2)$ (s.u.), $SO(2)$ (Rotationsmatrizen), $GL(2, \mathbb{R})$ (s.o.) und $SL(2, \mathbb{R})$ (invertierbare Matrizen mit einer Determinante von eins) sein [55].

Nun kann angenommen werden, dass y die diskrete Form eines $y \in \mathcal{G}$ ist. Das bedeutet, dass y gleichzeitig eine Abbildung, die einen Vektor aus dem \mathbb{R}^d in den \mathbb{R}^d transformiert, und eine Element der Lie-Gruppe ist.

Ein mögliches Distanzmaß, mit dem die Ähnlichkeit zweier Bilder gemessen werden kann, ist das *sum of squared differences*-Distanzmaß (SSD-Maß). Es bestimmt die Ähnlichkeit von Bildern über deren Grauwertdifferenzen:

$$\mathcal{D}(y) = \frac{1}{2} \|R - T(y)\|^2 = \frac{1}{2m} \sum_{j=1}^m |\mathcal{R}(x^j) - \mathcal{T}(y(x^j))|^2. \quad (5.2)$$

Es gibt in der Bildregistrierung die unterschiedlichsten Distanzmaße [33, 38, 19, 52], in [49] wurde nur das SSD-Maß betrachtet.

Die Optimierung mit dem Gauß-Newton-Verfahren Für die Minimierung von (5.2) kann das Gauß-Newton-Verfahren auf Lie-Gruppen verwendet werden, wobei die Funktionen r, r_j des nicht-linearen least-squares Problems gegeben sind durch

$$\begin{aligned} r : \mathcal{G} &\rightarrow \mathbb{R}^m & \text{mit } r(y) &= R - T(y), m < \infty, \\ r^j : \mathcal{G} &\rightarrow \mathbb{R} & \text{mit } r^j(y) &= \mathcal{R}(x^j) - \mathcal{T}(y(x^j)) \end{aligned}$$

mit $y = (y(x^1)^T, \dots, y(x^m)^T)^T$. Im Verfahren wird eine lineare Approximation der Zielfunktion an der Stelle $y \circ \exp(u)$ vorgenommen und

$$r^j(y \circ \exp(u)) = \mathcal{R}(x^j) - \mathcal{T}((y \circ \exp(u))(x^j)), \quad (5.3)$$

so dass hier die Ableitung $dr_y^j(0) = \frac{\partial}{\partial u} r^j(y \circ \exp(u))|_{u=0}$ berechnet werden muss. Dies wird über eine Einbettung der Lie-Gruppe in einen euklidischen Raum durchgeführt.

Einbettung von Lie-Gruppen zur Berechnung der Ableitungen Die Ableitung $dr_y^j(0)$ muss für praktische Anwendungen berechnet werden. Dies wird über eine sogenannte Einbettung der Lie-Gruppe in einen euklidischen Raum durchgeführt. Nach dem Satz von Whitney

[26] existiert eine Einbettung $\Theta : \mathcal{G} \rightarrow \mathbb{R}^N$, $y \mapsto \Theta(y)$ der Lie-Gruppe \mathcal{G} in den euklidischen Raum \mathbb{R}^N .

Wie oben erwähnt ist die Transformation y gleichzeitig ein Element der Lie-Gruppe \mathcal{G} und eine Abbildung, die einen Punkt $x \in \mathbb{R}^d$ transformiert. Um einen Punkt $x \in \mathbb{R}^d$ in der Einbettung durch y zu transformieren, wird eine Funktion benötigt, die die Struktur der Transformation in der Einbettung beschreibt. Sei $w(\Theta(y), x)$ diese Funktion, also die Darstellung der Transformation eines Punktes x durch die Transformation y . Es gilt, dass $w(\Theta(y), x) \in \mathbb{R}^d$. Mit der Repräsentierung der Elemente der Lie-Gruppe in einem euklidischen Raum kann die Ableitung $dr_y^j(0)$ bestimmt werden. Es gilt mit $r^j(y) = \mathcal{R}(x^j) - \mathcal{T}(y(x^j))$

$$dr_y^j(0) = \left. \frac{\partial}{\partial u} r^j(y \circ \exp(u)) \right|_{u=0} = - \left. \frac{\partial}{\partial u} \mathcal{T}(y \circ \exp(u)(x^j)) \right|_{u=0}.$$

Die Darstellung von $\exp(u)(x^j)$ im Einbettungsraum ist $\exp(u)(x^j) = w(\Theta(\exp(u)), x^j)$. Dies bezeichnet die Transformation von x^j durch $\exp(u)$ im euklidischen Einbettungsraum. Es gilt

$$\mathcal{T}(y \circ \exp(u)(x^j)) = \mathcal{T}(y(w(\Theta(\exp(u)), x^j))).$$

Sei $\{e_1, \dots, e_n\}$ eine Basis der Lie-Algebra \mathfrak{g} , wobei n die Dimension der Lie-Gruppe als differenzierbare Mannigfaltigkeit ist (vgl. Kapitel 3). Jedes $u \in \mathfrak{g}$ lässt sich darstellen, als eine Linearkombination der Basisvektoren $\{e_1, \dots, e_n\}$. Wie in [47] gezeigt, gilt mit der Kettenregel

$$\begin{aligned} (dr_y^j(0))_i &= - \left. \frac{\partial}{\partial u_i} \mathcal{T}(y(w(\Theta(\exp(u)), x^j)) \right|_{u_i=0} \\ &= - \underbrace{\left. \frac{\partial \mathcal{T}(y(z))}{\partial z} \right|_{z=\exp(0)(x^j)=x^j}}_{=d\mathcal{T}_y(x^j)} \cdot \underbrace{\left. \frac{\partial w(\xi, x^j)}{\partial \xi} \right|_{\xi=\Theta(\exp(0))=\Theta(id)}}_{=:dw^j} \cdot \underbrace{\left. \frac{\partial \Theta(\exp(\sum_i u_i e_i))}{\partial u_i} \right|_{u_i=0}}_{=\Theta(e_i)} \\ &= -d\mathcal{T}_y(x^j) \cdot dw^j \cdot \Theta(e_i), \end{aligned}$$

wobei $\Theta(e_i)$ die Darstellung des i -ten Basisvektors e_i der Lie-Algebra \mathfrak{g} in der Einbettung ist. Ungeklärt bleibt, warum $\left. \frac{\partial \Theta(\exp(\sum_i u_i e_i))}{\partial u_i} \right|_{u_i=0} = \Theta(e_i)$ ist. In [47] wird argumentiert, dass das Differential der Exponentialabbildung am neutralen Element die Identität ist, aber offen bleibt, was die Ableitung der Einbettung Θ ist oder wie diese berechnet werden kann.

Es gilt, dass $d\mathcal{T}_y(x^j) \in \mathbb{R}^{1 \times d}$, $dw^j \in \mathbb{R}^{d \times N}$ und $\Theta(e_i) \in \mathbb{R}^{N \times 1}$, wobei N die Dimension des Einbettungsraumes ist. Somit ist $(dr_y^j(0))_i \in \mathbb{R}$ und

$$dr_y^j(0) = -d\mathcal{T}_y(x^j) \cdot dw^j \cdot e_\Theta \in \mathbb{R}^{1 \times n}, \quad (5.4)$$

wobei $e_\Theta = (\Theta(e_1), \dots, \Theta(e_n)) \in \mathbb{R}^{N \times n}$ die in der Einbettung dargestellten Basisvektoren von \mathfrak{g} enthält.

Es wurden die Ableitungen dr_y^j bestimmt. Für das Gauß-Newton-Verfahren wird die Ableitung dr_y benötigt (siehe Algorithmus 2). Es gilt

$$dr_y = \begin{pmatrix} dr_y^1 \\ \vdots \\ dr_y^m \end{pmatrix} \in \mathbb{R}^{m \times n}.$$

Um diese Methode zur Bildregistrierung und die Einbettung einer Lie-Gruppe in einen euklidischen Raum etwas greifbarer zu machen, wird im Folgenden die Einbettung am Beispiel der Lie-Gruppe der rigiden Transformationen vorgestellt.

Einbettung der rigiden Transformationen in den \mathbb{R}^9 Die rigiden Transformationen bilden eine Lie-Gruppe und es wird beispielhaft gezeigt, wie ein Gauß-Newton-Schritt durchgeführt werden kann. Dazu wird im Folgenden die Einbettung dieser Lie-Gruppe beschrieben, die Lie-Algebra der eingebetteten Lie-Gruppe, dw^j und e_Θ werden hergeleitet. Zusammen mit der Bildableitung kann damit die Ableitung dr_y^j für jeden Punkt und die Ableitung dr_y für alle Punkte bestimmt werden. Mit der Ableitung dr_y kann die Gauß-Newton-Richtung u durch $dr_y^T dr_y \cdot u = dr_y^T r(y)$ berechnet werden. Die Komponenten von u sind die Koeffizienten für die Basiselemente der Lie-Algebra der Einbettung. Das so entstehende Element der Lie-Algebra wird durch das Matrix Exponential wieder auf die Lie-Gruppe abgebildet.

Die Menge $SE(2)$ der 2-dimensionalen rigiden Transformationen

$$SE(2) = \left\{ y \mid y(x) = \begin{pmatrix} \cos(\alpha) & -\sin(\alpha) \\ \sin(\alpha) & \cos(\alpha) \end{pmatrix} \cdot \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} + \begin{pmatrix} t_1 \\ t_2 \end{pmatrix}, \alpha, t_1, t_2 \in \mathbb{R} \right\}$$

enthält die Transformationen, die aus einer Rotation mit dem Rotationswinkel α und einer Translation mit dem Translationsvektor $t = (t_1, t_2)^T$ bestehen.

Die Menge $SE(2)$ bildet eine Lie-Gruppe: Sei $y \in SE(2)$ die durch $y(x) = \begin{pmatrix} \cos(\alpha) & -\sin(\alpha) \\ \sin(\alpha) & \cos(\alpha) \end{pmatrix} \cdot \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} + \begin{pmatrix} t_1 \\ t_2 \end{pmatrix}$

definierte rigide Transformation. Durch

$$\Theta(y) = \begin{pmatrix} \cos(\alpha) & -\sin(\alpha) & t_1 \\ \sin(\alpha) & \cos(\alpha) & t_2 \\ 0 & 0 & 1 \end{pmatrix}$$

kann sie auf die Menge

$$\left\{ \begin{pmatrix} \cos(\alpha) & -\sin(\alpha) & t_1 \\ \sin(\alpha) & \cos(\alpha) & t_2 \\ 0 & 0 & 1 \end{pmatrix}, \alpha, t_1, t_2 \in \mathbb{R} \right\} \quad (5.5)$$

abgebildet werden. Die Abbildung Θ ist bijektiv [34] und $SE(2)$ kann somit mit der Menge (5.5) identifiziert werden. Diese ist eine abgeschlossene Teilmenge der Lie-Gruppe $GL(3, \mathbb{R})$ (vgl. Kapitel 3) und selbst eine Lie-Gruppe [20, 26]. Daher ist auch $SE(2)$ eine Lie-Gruppe.

Eine Karte, die von der Lie-Gruppe der rigiden Transformationen in einen euklidischen Raum abbildet, ist die Karte $\varphi : SE(2) \rightarrow \mathbb{R}^3$ mit $\varphi(y) = (\alpha, t_1, t_2)^T$. Diese Karte ist sogar global und $SE(2)$ ist eine 3-dimensionale Mannigfaltigkeit.

Die Abbildung Θ , die die rigide Transformation y mit ihrer Abbildungsmatrix identifiziert, ist die Einbettung der Lie-Gruppe $SE(2)$ in den Einbettungsraum \mathbb{R}^9 .

Um das Gauß-Newton-Verfahren beispielhaft durchzurechnen, müssen die Ausdrücke dw^j und e_Θ bestimmt werden. Für e_Θ wird die Lie-Algebra der eingebetteten Lie-Gruppe berechnet. Die Lie-Algebra einer Lie-Gruppe ist der Tangentialraum am neutralen Element. Das neutrale Element ε der Lie-Gruppe $SE(2)$ wird in der Einbettung repräsentiert durch

$$\Theta(\varepsilon) = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}.$$

Um die Tangentialvektoren der Einbettung $\Theta(SE(2))$ zu berechnen, wird eine Kurve c in der Einbettung betrachtet mit $c(0) = \Theta(\varepsilon)$. Die Repräsentierung von $c'(0)$ in der Einbettung soll berechnet werden. Für $c(t)$ gilt:

$$c(t) = \begin{pmatrix} c_1(t) & c_2(t) & c_3(t) \\ c_4(t) & c_5(t) & c_6(t) \\ c_7(t) & c_8(t) & c_9(t) \end{pmatrix} \stackrel{(*)}{=} \begin{pmatrix} \cos(f_1(t)) & \sin(f_1(t)) & f_2(t) \\ -\sin(f_1(t)) & \cos(f_1(t)) & f_3(t) \\ 0 & 0 & 1 \end{pmatrix}.$$

Die Gleichheit (*) gilt wegen der Identifizierung einer rigiden Transformation mit ihrer Abbildungsmatrix. Da $c(0) = \Theta(\varepsilon)$ ist, gilt

$$c(0) = \begin{pmatrix} \cos(f_1(0)) & \sin(f_1(0)) & f_2(0) \\ -\sin(f_1(0)) & \cos(f_1(0)) & f_3(0) \\ 0 & 0 & 1 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} = \Theta(\varepsilon).$$

Um $c'(0)$ zu bestimmen, werden die $c'_i(t)$'s abgeleitet:

$$\begin{aligned} c'_1(t) &= -f'_1(t) \sin(f_1(t)) \quad \Rightarrow \quad c'_1(0) = -f'_1(0) \underbrace{\sin(f_1(0))}_{=0} = 0 \\ c'_2(t) &= f'_1(t) \cos(f_1(t)) \quad \Rightarrow \quad c'_2(0) = f'_1(0) \underbrace{\cos(f_1(0))}_{=1} = f'_1(0) =: a. \end{aligned}$$

Dies wird analog für die anderen $c_i(t)$'s durchgeführt. Damit ergibt sich für $c'(0)$:

$$c'(0) = \begin{pmatrix} 0 & a & b \\ -a & 0 & c \\ 0 & 0 & 0 \end{pmatrix}, \quad \text{mit } a, b, c \in \mathbb{R}.$$

Die Lie-Algebra der Lie-Gruppe $SE(2)$ in der Einbettung ist demnach [34, 47]

$$\mathfrak{se}(2) := \left\{ \left(\begin{pmatrix} 0 & a & b \\ -a & 0 & c \\ 0 & 0 & 0 \end{pmatrix} \middle| a, b, c \in \mathbb{R} \right\}.$$

Folgenden Matrizen bilden eine Basis von $\mathfrak{se}(2)$:

$$\Theta(e_1) = \begin{pmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} \quad \Theta(e_2) = \begin{pmatrix} 0 & 0 & 1 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} \quad \Theta(e_3) = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{pmatrix}.$$

Es werden drei Basiselemente gewählt, da die Dimension der Lie-Gruppe $SE(2)$ drei ist. Der Einbettungsraum ist der \mathbb{R}^9 , so dass die Basiselemente hier als Vektoren geschrieben werden:

$$\begin{aligned} \Theta(e_1) &= (0, 1, 0, -1, 0, 0, 0, 0, 0)^T \\ \Theta(e_2) &= (0, 0, 0, 0, 0, 0, 1, 0, 0)^T \\ \Theta(e_3) &= (0, 0, 0, 0, 0, 0, 0, 1, 0)^T. \end{aligned}$$

Um die Ableitungen berechnen zu können, müssen dw^j und e_Θ (vgl. Gleichung (5.4)) bestimmt werden. Für die Dimensionen der Matrizen gilt: $dw^j \in \mathbb{R}^{d \times N}$ und $e_\Theta \in \mathbb{R}^{N \times n}$. Hier ist $d = 2$ (man betrachtet die 2-dimensionalen rigiden Transformationen), $N = 9$ (der Einbettungsraum ist der \mathbb{R}^9) und $n = 3$ (die Dimension der Lie-Gruppe $SE(2)$ ist 3).

Für einen Punkt $x^j \in \mathbb{R}^2$ und eine rigide Transformation $y \in SE(2)$ beschreibt $w(\Theta(y), x^j)$ die Darstellung von $y(x^j)$ im Einbettungsraum:

$$w(\Theta(y), x^j) = \begin{pmatrix} \cos(\alpha)x_1^j + \sin(\alpha)x_2^j + t_1 \\ -\sin(\alpha)x_1^j + \cos(\alpha)x_2^j + t_2 \end{pmatrix}.$$

Damit ist

$$dw^j = \frac{\partial w(\xi, x^j)}{\partial \xi^T} \Big|_{\xi=\Theta(id)} = \begin{pmatrix} x_1^j & 0 & 0 & x_2^j & 0 & 0 & 1 & 0 & 0 \\ 0 & x_1^j & 0 & 0 & x_2^j & 0 & 0 & 1 & 0 \end{pmatrix} \in \mathbb{R}^{2 \times 9}.$$

Mit $e_\Theta = (\Theta(e_1), \Theta(e_2), \Theta(e_3)) \in \mathbb{R}^{9 \times 3}$ ist

$$dw^j \cdot e_\Theta = \begin{pmatrix} x_1^j & 0 & 0 & x_2^j & 0 & 0 & 1 & 0 & 0 \\ 0 & x_1^j & 0 & 0 & x_2^j & 0 & 0 & 1 & 0 \end{pmatrix} \begin{pmatrix} 0 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \\ -1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{pmatrix} = \begin{pmatrix} -x_2^j & 1 & 0 \\ x_1^j & 0 & 1 \end{pmatrix}.$$

Damit kann jetzt die Ableitung für einen Punkt x^j

$$dr_y^j = -dT_y(x^j) \cdot dw^j \cdot e_\Theta = -dT_y(x^j) \begin{pmatrix} -x_2^j & 1 & 0 \\ x_1^j & 0 & 1 \end{pmatrix} \in \mathbb{R}^{1 \times 3}$$

und die Ableitung $dr_y = ((dr_y^1)^T, \dots, (dr_y^m)^T) \in \mathbb{R}^{m \times 3}$ für alle Punkte x^1, \dots, x^m bestimmt werden.

Für die Gauß-Newton-Richtung u gilt nun $\underbrace{dr_y^T dr_y}_{\in \mathbb{R}^{3 \times 3}} \cdot u = -\underbrace{dr_y^T r(y)}_{\in \mathbb{R}^{3 \times 1}}$. Dies ist ein 3×3 lineares Gleichungssystem. Die Komponenten u_i der Lösung $u \in \mathbb{R}^3$ sind die Koeffizienten der

Basisvektoren $\Theta(e_i)$. Es gilt

$$X = \sum_{i=1}^3 u_i \cdot \Theta(e_i) = \begin{pmatrix} 0 & -u_1 & u_2 \\ u_1 & 0 & u_3 \\ 0 & 0 & 0 \end{pmatrix} \in \mathfrak{se}(2).$$

Im Update-Schritt wird $X \in \mathfrak{se}(2)$ durch die Exponentialabbildung wieder auf die Lie-Gruppe abgebildet und mit der aktuellen Iterierten, hier y , verknüpft. Die Exponentialabbildung für Matrizen-Lie-Gruppen ist das Matrix Exponential (vgl. Gleichung 4.3):

$$\exp(tX) = \sum_{k=0}^{\infty} \frac{t^k}{k!} X^k.$$

Man kann zeigen, dass $\exp : \mathfrak{se}(2) \rightarrow \text{SE}(2)$ [28], wobei mit $\text{SE}(2)$ die Menge (5.5) gemeint ist. Die Verknüpfung $z = y \circ \exp(X) \in \mathcal{G}$ ist damit wieder eine rigide Transformation.

In diesem Kapitel wurde eine Bildregistrierungsmethode unter Verwendung von Lie-Gruppen beschrieben [47, 49]. Die Optimierung eines bekannten Distanzmaßes der Registrierung, des SSD-Maßes, wurde mit dem Gauß-Newton-Verfahren auf Lie-Gruppen [49] durchgeführt, wobei die Ableitungen mit Hilfe einer Einbettung der Lie-Gruppe in einen euklidischen Raum berechnet werden und die Suchrichtung durch die Exponentialabbildung auf die Lie-Gruppe abgebildet wird. Durch das Beispiel der rigiden Transformationen konnte diese Methode veranschaulicht werden.

Kapitel 6

Optimierung auf Lie-Gruppen und der Dämonen-Algorithmus

In [44] wurde die nicht-parametrische Registrierung als ein Diffusionsprozess interpretiert. Sogenannte Dämonen verschieben die Bildpunkte entsprechend den Dämonenkräften und der Dämonen-Algorithmus verläuft so, dass er abwechselnd die Dämonenkräfte berechnet und eine so entstandene Verrückung mit einer Gauß-Funktion glättet.

In [49] wurde ein Registrierungsrahmen basierend auf der Optimierung auf Lie-Gruppen vorgestellt, der im vorherigen Kapitel beschrieben wurde. Es wurde in [49] zusätzlich gezeigt, dass der Dämonen-Algorithmus in diesen Rahmen passt: Im Falle der additiven Dämonen werden alle Transformationen, die vom \mathbb{R}^d in den \mathbb{R}^d abbilden, betrachtet. Es handelt sich um die Menge aller Vektorfelder auf dem \mathbb{R}^d , die einen Vektorraum und damit eine triviale Lie-Gruppe bilden. Die Exponentialabbildung ist die Identität und die Gruppenverknüpfung die Addition. Damit gilt für die additiven Dämonen $y \circ \exp(u) = y + u$.

Im Fall der kompositiven Dämonen wird die Menge der Transformationen betrachtet, die vom \mathbb{R}^d in den \mathbb{R}^d abbilden und deren natürliche Verknüpfung die Komposition \circ von Funktionen ist. Dann gilt für die kompositiven Dämonen $y \circ \exp(u) = y \circ (id + u)$.

Dieses Kapitel gliedert sich in zwei Teile, im ersten wird eine Zielfunktion vorgestellt und es wird gezeigt, dass deren Minimierung mit den im vorherigen Kapitel beschriebenen Methoden zu einer Berechnung der Dämonenkräfte in jedem Iterationsschritt führt. Im zweiten Teil wird eine Erweiterung des Dämonen-Algorithmus auf Diffeomorphismen kurz vorgestellt, der

diffeomorphe Dämonen-Algorithmus. Er beruht auf einer Optimierung über der Gruppe der Diffeomorphismen und unterscheidet sich vom Dämonen-Algorithmus nur in der Update-Regel. Es wird nicht genauer auf die diffeomorphe Bildregistrierung eingegangen. Hierzu siehe z.B. [3, 12].

6.1 Der Dämonen-Algorithmus als ein Optimierungsverfahren

Mit dem Dämonen-Algorithmus soll das Templatebild T auf das Referenzbild R registriert werden. Die Zielfunktionen, deren Minimierungen durch den Dämonen-Algorithmus erreicht werden sollen, sind

$$E_a(y) = \|R - T(y + u)\|^2 + \frac{\alpha_1^2}{\alpha_2} \|u\|^2 \quad (6.1)$$

im additiven und

$$E_c(y) = \|R - T(y \circ (id + u))\|^2 + \frac{\alpha_1^2}{\alpha_2} \|u\|^2 \quad (6.2)$$

im kompositiven Fall. Der erste Summand der Zielfunktionen entspricht dem SSD-Maß. Der zweite Summand kann als Nebenbedingung interpretiert werden. Für eine detaillierte Motivation dieser Wahl der Zielfunktion siehe [49, 47, 10]. Auf die Bedeutung des Parameters $\frac{\alpha_1^2}{\alpha_2} \in \mathbb{R}$ wird im weiteren Verlauf des Kapitels noch eingegangen.

Die Interpretation des so entstehenden Registrierungsproblems ist, dass die Bilder R und T mit dem SSD-Maß als Ähnlichkeitsmaß aufeinander registriert werden sollen. Zusätzlich wird gefordert, dass die Verrückung u möglichst klein sein soll. Die Regularisierung wird durch die Faltung mit einer Gauß-Funktion realisiert [10].

Diese Zielfunktionen sollen bezüglich u minimiert werden. Da es sich um nicht-lineare least-squares Probleme handelt, kann das Gauß-Newton-Verfahren [49, 47] (vgl. Kapitel 5) angewendet werden. Dementsprechend wird eine Linearisierung der inneren Funktion der ersten Terme von (6.1) und (6.2) durchgeführt: Sei $r^j(y) = \mathcal{R}(x^j) - \mathcal{T}(y(x^j))$ die Intensitätsdifferenz des Referenz- und des Templatebildes an einem Punkt x^j und $r_y^j(u) := \mathcal{R}(x^j) - \mathcal{T}((y + u)(x^j))$ im additiven und $r_y^j(u) := \mathcal{R}(x^j) - \mathcal{T}(y \circ (id + u)(x^j))$ im kompositiven Fall. Anzumerken ist, dass der Ausdruck $y \circ \exp(u)$ aus Gleichung (5.3) ersetzt wurde durch $y + u$ bzw. durch $y \circ (id + u)$. Für jeden Punkt x^j wird die Intensitätsdifferenz linearisiert:

$$r_y^j(u) \approx r_y^j(0) + dr_y^j(0)u(x^j) = \mathcal{R}(x^j) - \mathcal{T}(y(x^j)) + dr_y^j(0)u(x^j). \quad (6.3)$$

Berechnung der Ableitung Die Ableitung $dr_y^j(0) \in \mathbb{R}^{1 \times d}$ ist abhängig von der Update-Regel, additiv oder kompositiv. Der Registrierungsrahmen aus [49] hängt stark von der Einbettung der Lie-Gruppe in einen euklidischen Raum ab. Im Falle der Dämonen ist dies der Raum der Vektorfelder, so dass sich die Berechnungen der Ableitungen vereinfachen, da $\Theta(y) = y$, $w(\Theta(y), x^j) = y(x^j)$, $y \circ \exp(u) = y + u$ im additiven Fall und $y \circ \exp(u) = y \circ (id + u)$ im kompositiven Fall. [47].

Sei $r_y^j(u) := \mathcal{R}(x^j) - \mathcal{T}(y \circ (id + u)(x^j))$ die Intensitätsdifferenz im kompositiven Fall. Durch eine Taylorentwicklung erhält man eine Linearisierung $r_y^j(u) = r_y^j(0) + dr_y^j(0)u + \mathcal{O}(\|u\|^2)$, wobei $dr_y^j(0) \in \mathbb{R}^{1 \times d}$ mit $dr_y^j(0) = \left. \frac{\partial}{\partial u^i} r_y^j(u) \right|_{u=0}$ und $u^i = u(x^i)$. Anzumerken ist, dass nun nicht nach einer Komponente u_i abgeleitet wird, sondern nach $u(x^i) \in \mathbb{R}^d$.

Für die kompositive Update-Regel gilt

$$\begin{aligned} dr_y^j(0) &= \left. \frac{\partial}{\partial u^i} r_y^j(u) \right|_{u=0} \\ &= - \left. \frac{\partial}{\partial u^i} \mathcal{T}(y(id + u)(x^j)) \right|_{u=0} \\ &= - \left. \frac{\partial}{\partial u^i} \mathcal{T}(y(x^j + u(x^j))) \right|_{u=0} \\ &= -d\mathcal{T}_y(x^j) \underbrace{\left. \frac{\partial}{\partial u^i} (x^j + u^j) \right|_{u=0}}_{=\delta_{i,j}} \\ &= -d\mathcal{T}_y(x^j) \cdot \delta_{i,j}, \end{aligned}$$

was eine Taylorentwicklung erster Ordnung für r^j liefert:

$$r_y^j(u) = r_y^j(0) - d\mathcal{T}_y(x^j)u + \mathcal{O}(\|u\|^2).$$

Für die additive Update-Regel gilt dagegen:

$$\begin{aligned} dr_y^j(0) &= \left. \frac{\partial}{\partial u^j} r_y^j(u) \right|_{u=0} \\ &= - \left. \frac{\partial}{\partial u^j} \mathcal{T}((y + u)(x^j)) \right|_{u=0} \\ &= - \left. \frac{\partial}{\partial u^j} \mathcal{T}(y(x^j) + u(x^j)) \right|_{u=0} \\ &= -d\mathcal{T}(y(x^j)) \left. \frac{\partial}{\partial u^j} (x^j + u^j) \right|_{u=0} \\ &= -d\mathcal{T}(y(x^j)). \end{aligned}$$

Damit ist die Taylorentwicklung erster Ordnung von r_j im additiven Fall

$$r_y^j(u) = r_y^j(0) - d\mathcal{T}(y(x^j))u + \mathcal{O}(\|u\|^2).$$

Anzumerken ist der Unterschied zwischen den Ableitungen $d\mathcal{T}_y$ und $d\mathcal{T}$ des Templatebildes \mathcal{T} . Es gilt der Zusammenhang $d\mathcal{T}_y = d\mathcal{T} \cdot dy$.

Die Berechnungen der Ableitungen wurden in [47] beschrieben. Warum im additiven Fall nach u^j und im kompositiven Fall nach u^i abgeleitet wird, konnte noch nicht geklärt werden.

Übersichtshalber wird in den folgenden Berechnungen $dr_j := dr_y^j(0)$ gesetzt. Werden diese Taylorentwicklungen für jeden Punkt x^i als Linearisierung der inneren Funktion des ersten Terms in (6.1) und (6.2) verwendet, erhält man eine Approximation an die Zielfunktionen folgendermaßen:

$$\begin{aligned} E_a(u) &\approx \frac{1}{2m} \sum_{j=1}^m |\mathcal{R}(x^j) - \mathcal{T}(y(x^j)) + dr_j \cdot u(x^j)|^2 + \frac{\alpha_1^2}{\alpha_2^2} \|u\|^2 \\ &= \frac{1}{2m} \sum_{j=1}^m |\mathcal{R}(x^j) - \mathcal{T}(y(x^j)) + dr_j \cdot u(x^j)|^2 + \frac{\alpha_1(x^j)^2}{\alpha_2^2} \|u(x^j)\|^2 \\ &= \frac{1}{2m} \sum_{j=1}^m \left\| \begin{pmatrix} \mathcal{R}(x^j) - \mathcal{T}(y(x^j)) \\ \mathbf{0}_{d \times 1} \end{pmatrix} + \begin{pmatrix} dr_j \\ \frac{\alpha_1(x^j)}{\alpha_2} I \end{pmatrix} u(x^j) \right\|^2, \end{aligned} \quad (6.4)$$

wobei $\mathbf{0}_{d \times 1} = (0, \dots, 0)^T \in \mathbb{R}^{d \times 1}$ und I die $d \times d$ Einheitsmatrix ist. Dasselbe gilt auch für die lineare Approximation von $E_c(u)$. Anzumerken ist, dass der Zähler α_1 des Regularisierungsparameters abhängig ist vom Punkt x^j . Hierauf wird später noch eingegangen.

Für jeden Punkt x^j sind die Approximationen unabhängig voneinander. Die Minimierung von $E_a(u)$ bzw. $E_c(u)$ vereinfacht sich durch diese Approximation, da sie in einfache $(d+1) \times d$ Systeme für jeden Punkt aufgeteilt werden kann. Die Unabhängigkeit der Approximationen für jeden Pixel ist im Falle von rigiden Transformationen (siehe vorherigen Abschnitt) nicht gegeben.

Für jeden Datenpunkt x^j muss ein in $u(x^j)$ lineares Problem der Form $\|b + A \cdot u\|^2$ gelöst werden (least-squares Problem), wobei $b \in \mathbb{R}^{(d+1) \times 1}$, $A \in \mathbb{R}^{(d+1) \times d}$ und $u \in \mathbb{R}^{d \times 1}$. Typischerweise wird ein derartiges Problem mit Hilfe einer QR-Zerlegung gelöst [18]. In [49, 47] wurde die Normalengleichung [36, 18] zur Lösung verwendet. Voraussetzung dafür ist, dass die Matrix A vollen Rank hat, was in diesem Falle gegeben ist. Mit der Abkürzung $a_j = \frac{\alpha_1(x^j)}{\alpha_2}$ muss

folgendes System für jeden Punkt x^j gelöst werden:

$$\begin{aligned} \begin{pmatrix} dr_j^T & a_j \cdot I \end{pmatrix} \begin{pmatrix} dr_j \\ a_j \cdot I \end{pmatrix} \cdot u(x^j) &= - \begin{pmatrix} dr_j^T & a_j \cdot I \end{pmatrix} \begin{pmatrix} \mathcal{R}(x^j) - \mathcal{T}(y(x^j)) \\ \mathbf{0}_{d \times 1} \end{pmatrix} \\ (dr_j^T dr_j + a_j^2 \cdot I) \cdot u(x^j) &= - (\mathcal{R}(x^j) - \mathcal{T}(y(x^j))) dr_j^T \\ u(x^j) &= - (dr_j^T dr_j + a_j^2 \cdot I)^{-1} (\mathcal{R}(x^j) - \mathcal{T}(y(x^j))) dr_j^T. \end{aligned} \quad (6.5)$$

Voraussetzung für (6.5) ist natürlich die Existenz der Inversen. Die Existenz wird in diesem speziellen Fall gewährleistet durch die Sherman-Morrison-Woodbury-Formel (das Matrix-Inversions-Lemma), welche einen geeigneten Ausdruck für die Inverse $(dr_j^T dr_j + a_j^2 I)^{-1}$ liefert [18]. Das Lemma besagt, dass das Inverse der Summe einer invertierbaren Matrix A mit dem Produkt eines Spaltenvektors u und eines Zeilenvektors v^T gegeben ist durch

$$(A + uv^T)^{-1} = A^{-1} - \frac{A^{-1}uv^T A^{-1}}{1 + v^T A^{-1}u}. \quad (6.6)$$

Mit dieser Formel soll das Inverse von $(dr_j^T dr_j + a_j^2 \cdot I)$ bestimmt werden, wobei $a_j^2 \cdot I = A \in \mathbb{R}^{d \times d}$ mit $a_j^2 \cdot I = A^{-1}$ und $u = v = dr_j^T \in \mathbb{R}^{d \times 1}$. Nach der Formel (6.6) gilt

$$(dr_j^T dr_j + a_j^2 I)^{-1} = \frac{1}{a_j^2} I - \frac{\frac{1}{a_j^2} I dr_j^T dr_j \frac{1}{a_j^2} I}{1 + dr_j^T \frac{1}{a_j^2} I dr_j^T} = \frac{1}{a_j^2} I - \frac{1}{a_j^2} \frac{dr_j^T dr_j}{a_j^2 + dr_j^T dr_j}.$$

Dieses wird in (6.5) eingesetzt und damit ergibt sich für $u(x^j)$:

$$\begin{aligned} u(x^j) &= - \left(\frac{1}{a_j^2} I - \frac{1}{a_j^2} \frac{dr_j^T dr_j}{a_j^2 + dr_j^T dr_j} \right) (\mathcal{R}(x^j) - \mathcal{T}(y(x^j))) dr_j^T \\ &= - \frac{(\mathcal{R}(x^j) - \mathcal{T}(y(x^j))) (dr_j^T a_j^2 + dr_j^T dr_j dr_j^T - dr_j^T dr_j dr_j^T)}{(a_j^2 + dr_j^T dr_j) a_j^2} \\ &= - \frac{\mathcal{R}(x^j) - \mathcal{T}(y(x^j))}{\|dr_j\|^2 + a_j^2} dr_j^T = - \frac{\mathcal{R}(x^j) - \mathcal{T}(y(x^j))}{\|dr_j\|^2 + \frac{\alpha_1^2(x^j)}{\alpha_2^2}} dr_j^T. \end{aligned} \quad (6.7)$$

Die Suchrichtung u hat demnach für jeden Punkt x^j einen geschlossenen Ausdruck. Wird dieser Ausdruck mit den klassischen und symmetrischen Dämonenkräften

$$\begin{aligned} u(x^j) &= \frac{\mathcal{R}(x^j) - \mathcal{T}_y(x^j)}{\|\nabla \mathcal{R}(x^j)\|^2 + \alpha^2 (\mathcal{R}(x^j) - \mathcal{T}_y(x^j))^2} \nabla \mathcal{R}(x^j) \quad (\text{klassisch}) \\ u(x^j) &= \frac{2(\mathcal{R}(x^j) - \mathcal{T}_y(x^j))(\nabla \mathcal{R}(x^j) + \nabla \mathcal{T}_y(x^j))}{\|\nabla \mathcal{R}(x^j) + \nabla \mathcal{T}_y(x^j)\|^2 + \alpha^2 (\mathcal{R}(x^j) - \mathcal{T}_y(x^j))^2} \quad (\text{symmetrisch}) \end{aligned}$$

verglichen, fällt sofort die Ähnlichkeit der Ausdrücke auf. In [49] wird argumentiert, dass der Parameter α_1 für das Rauschen in den Bildern steht [10] und dass dieser gewählt werden kann durch $\alpha_1(x^j) = |\mathcal{R}(x^j) - \mathcal{T}_y(x^j)|$. Der Parameter $\frac{1}{\alpha_2}$ ist im Vergleich mit den Dämonenkräften gleichzusetzen mit α^2 . Damit unterscheiden sich die Ausdrücke nur noch in dr_j^T . Würde $dr_j^T = -\nabla\mathcal{R}(x^j)$ gewählt werden, wäre die Suchrichtung im Punkt x^j der Verrückungsvektor, berechnet durch die klassischen Kräfte und mit $dr_j^T = -\frac{1}{2}(\nabla\mathcal{R}(x^j) + \nabla\mathcal{T}_y(x^j))$ berechnet durch die symmetrischen Kräfte. Durch die Optimierung mit dem Gauß-Newton-Verfahren ist im Falle der additiven Dämonen $dr_j = -d\mathcal{T}(y(x^j))$, also $dr_j^T = -\nabla\mathcal{T}(y(x^j))$ und im Falle der kompositiven $dr_j = -d\mathcal{T}_y(x^j)$, also $dr_j^T = -\nabla\mathcal{T}_y(x^j)$.

In [49, 47] wurde zusätzlich zum Gauß-Newton-Verfahren ein weiteres Optimierungsverfahren vorgestellt und auf die Optimierung auf Lie-Gruppen erweitert, die *Efficient Second-Order Minimization* [31]. Das Verfahren basiert auf der Idee des BFGS-Verfahrens (nach Broyden, Fletcher, Goldfarb und Shanno) zur Approximation der zweiten Ableitungen [36]. Es konnte gezeigt werden, dass mit diesem Verfahren mit der kompositiven Update-Regel $dr_j = -\frac{1}{2}(d\mathcal{R}(x^j) + d\mathcal{T}_y(x^j))$ und mit der additiven Update-Regel $dr_j = -\frac{1}{2}(d\mathcal{R}(x^j) + d\mathcal{T}(y(x^j)))$ gewählt werden kann. Damit gilt für die Gradienten $dr_j^T = -\frac{1}{2}(\nabla\mathcal{R}(x^j) + \nabla\mathcal{T}_y(x^j))$ bzw. $dr_j^T = -\frac{1}{2}(\nabla\mathcal{R}(x^j) + \nabla\mathcal{T}(y(x^j)))$.

Somit wurde argumentiert, dass die Minimierung der Zielfunktionen (6.1) und (6.2) zu einer Berechnung der verschiedenen Varianten der Dämonenkräfte führt. Durch die Optimierung mit dem Gauß-Newton-Verfahren konnten zwei weitere Kräfte entwickelt werden:

$$u(x^j) = \frac{\mathcal{R}(x^j) - \mathcal{T}_y(x^j)}{\|\nabla\mathcal{T}_y(x^j)\|^2 + \alpha^2(\mathcal{R}(x^j) - \mathcal{T}_y(x^j))^2} \nabla\mathcal{T}_y(x^j) \quad (\text{Gauß-Newton, kompositiv})$$

$$u(x^j) = \frac{\mathcal{R}(x^j) - \mathcal{T}_y(x^j)}{\|\nabla\mathcal{T}(y(x^j))\|^2 + \alpha^2(\mathcal{R}(x^j) - \mathcal{T}_y(x^j))^2} \nabla\mathcal{T}(y(x^j)) \quad (\text{Gauß-Newton, additiv}).$$

Somit kann der Dämonen-Algorithmus mit dieser neuen Interpretation angegeben werden als Algorithmus 3. Die Minimierung von $E_c(u)$ bzw. $E_a(u)$ führt zu einer Berechnung der verschiedenen Varianten der Dämonenkräfte, je nach Wahl des Optimierers. In Tabelle 6.1 sind die verschiedenen Möglichkeiten für die Wahl von dr_j in (6.7) zusammengefasst, die vorgestellt wurden.

Der in [49] vorgestellte Registrierungsrahmen kann, wie oben beschrieben, benutzt werden, um den Dämonen-Algorithmus als ein Verfahren zur Lösung eines Optimierungsproblems zu

Algorithmus 3 Der Dämonen-Algorithmus

Eingabe: zwei diskrete Bilder R, T an Datenpunkten $x^1, \dots, x^m \in \mathbb{R}^d$, $m \in \mathbb{R}$, Startwert y , Standardabweichung $\sigma \in \mathbb{R}$

Ausgabe: Transformation y

while !stop **do**

Berechne $\mathcal{R}, \mathcal{T}_y$ durch Interpolation

Berechne Update u durch Minimierung von:

$$E_c(u) = \|\mathcal{R} - \mathcal{T}(y \circ (id + u))\|^2 + \frac{\alpha_1^2}{\alpha_2^2} \|u\|^2 \quad (\mathbf{kompositiv})$$

$$E_a(u) = \|\mathcal{R} - \mathcal{T}(y + u)\|^2 + \frac{\alpha_1^2}{\alpha_2^2} \|u\|^2 \quad (\mathbf{additiv})$$

Fluidale Regularisierung: $u \leftarrow G_\sigma^0 * u$

Update-Schritt: $y \leftarrow y \circ (id + u) \quad (\mathbf{kompositiv})$

$y \leftarrow y + u \quad (\mathbf{additiv})$

Diffusive Regularisierung: $y \leftarrow G_\sigma^0 * y$

end while

interpretieren. Die besondere Struktur der Lie-Gruppen wurde hierbei jedoch nicht genutzt, da der Raum aller Transformationen vom \mathbb{R}^d in den \mathbb{R}^d betrachtet wird. Dieser bildet eine triviale Lie-Gruppe, so dass die Exponentialabbildung die Identitätsabbildung ist. Die besonderen Eigenschaften der Lie-Algebra und der Exponentialabbildung wurden überhaupt nicht verwendet.

In [51] wurde eine Erweiterung des Dämonen-Algorithmus vorgestellt, der diffeomorphe Dämonen-Algorithmus. Dieser beruht darauf, dass Diffeomorphismen eine Lie-Gruppe bilden und eine Zielfunktion mittels Optimierungsverfahren auf Lie-Gruppen [49, 29] minimiert wird. Im Folgenden wird diese Erweiterung der Dämonen kurz beschrieben. Für eine ausführlichere Beschreibung siehe [51, 47].

6.2 Eine diffeomorphe Erweiterung des Dämonen-Algorithmus

In der Bildregistrierung ist man bestrebt, diffeomorphe Transformationen zu bestimmen, vor allem da diese invertierbar sind. Zusätzlich sind sie glatt (differenzierbar) und führen so zu plausibel transformierten Bildern [3, 12].

	additiv	kompositiv
ESM	$-\frac{1}{2}(\nabla\mathcal{R}(x) + \nabla\mathcal{T}(y(x)))$	$-\frac{1}{2}(\nabla\mathcal{R}(x) + \nabla\mathcal{T}_y(x))$
Gauß-Newton	$-\nabla\mathcal{T}(y(x))$	$-\nabla\mathcal{T}_y(x)$
klassisch	$-\nabla\mathcal{R}(x)$	$-\nabla\mathcal{R}(x)$

Tabelle 6.1: Varianten für dr in (6.7), je nach Wahl der Update-Regel und des Optimierungsverfahren. „ESM“ steht für die Efficient Second-Order Minimization [49, 31], die für die kompositive Update-Regel zu den symmetrischen Dämonenkräften (vgl. Kapitel 2, Gleichung (2.6)) führt.

Die Idee des diffeomorphen Dämonen-Algorithmus ist, die Optimierung auf den Raum der Diffeomorphismen zu beschränken. Die C^∞ -Diffeomorphismen bilden eine Lie-Gruppe, so dass die Optimierungsverfahren auf Lie-Gruppen verwendet werden können.

Im Dämonen-Algorithmus wird für seine diffeomorphe Erweiterung nur der Update-Schritt der Optimierungsverfahren ersetzt. Statt einer Addition (im Fall der additiven Dämonen) und einer einfachen Komposition (im Fall der kompositiven Dämonen) wird die Exponentialabbildung berechnet. Um in den Optimierungsverfahren die Transformation mit der Suchrichtung aus der Lie-Algebra zu verknüpfen, wird

$$y \leftarrow y \circ \exp(u)$$

berechnet. In [2] wurde ein numerisches Verfahren vorgestellt, welches die Exponentialabbildung berechnet. Sei \mathcal{G} eine Lie-Gruppe und \mathfrak{g} die Lie-Algebra der Lie-Gruppe \mathcal{G} . Das numerische Verfahren beruht darauf, dass die Exponentialabbildung $\exp : \mathfrak{g} \rightarrow \mathcal{G}$ Ein-Parameter-Untergruppen $(\exp(tX))_{t \in \mathbb{R}}$ bildet, wobei $X \in \mathfrak{g}$ (vgl. Kapitel 4, Satz 4.3 (ii)). Wegen Satz 4.3 (iii) gilt $\exp(X) = \exp\left(\frac{X}{2}\right) \circ \exp\left(\frac{X}{2}\right) = \exp\left(\frac{X}{2}\right)^2$ und für jedes beliebige $N \in \mathbb{N}$:

$$\exp(X) = \exp\left(\frac{X}{N}\right)^N.$$

Das Potenzieren ist hierbei bezüglich der Komposition \circ von Funktionen gemeint. In dem numerischen Verfahren wird statt $\exp(X)$ der Ausdruck $\exp\left(\frac{X}{N}\right)^N$ berechnet. Auf die technischen Details des Verfahrens wird nicht weiter eingegangen, siehe dafür [2].

Im Folgenden wird davon ausgegangen, dass die Exponentialabbildung eines Elementes der Lie-Algebra berechnet werden kann.

Analog zu den additiven und kompositiven Dämonen (vgl. (6.1) und (6.2)) wird im diffeomorphen Dämonen-Algorithmus eine Zielfunktion der Form

$$E_d(y) = \|R - T(y \circ \exp(u))\|^2 + \frac{\alpha_1^2}{\alpha_2^2} \|u\|^2 \quad (6.8)$$

minimiert. Mit der linearen Approximation

$$\mathcal{R}(x^i) - \mathcal{T}(y \circ \exp(u)(x^i)) = \mathcal{R}(x^i) - \mathcal{T}(y(x^i)) + dr_y^j \cdot u(x^i) + \mathcal{O}(\|u(x^i)\|^2)$$

für jeden Punkt x^i ergeben sich analog zu den additiven und kompositiven Dämonen unabhängige Teilprobleme für jeden Punkt, die in einer Berechnung des Ausdrucks (6.7) resultieren.

Nun muss noch geklärt werden, wie für das Gauß-Newton-Verfahren die Ableitung dr_j im diffeomorphen Fall berechnet wird. Anders als bei der Berechnung für die kompositiven und additiven Dämonen, befindet man sich im Falle der diffeomorphen Dämonen auf einer Lie-Gruppe. Das bedeutet, dass die Ableitung über eine Einbettung bestimmt wird, denn $dr_y^j = -d\mathcal{T}_y(x^j) \cdot dw^j \cdot e_\Theta$ (vgl. Gleichung (5.4)).

Für die Berechnung von dr_y^j wird in [47] folgendermaßen argumentiert: Der Einbettungsraum der Lie-Gruppe der Diffeomorphismen ist der Raum aller Verrückungsfelder. Dies ermöglicht die Darstellung jeder diffeomorphen Transformation $y = id + u$ durch die Verrückung u . Die Einbettung Θ ist damit $\Theta(y) = u$ und die Darstellung der Transformation eines Punktes x^j durch den Diffeomorphismus y ist gegeben durch

$$w(\Theta(y), x^j) = w(u, x^j) = y(x^j) = x^j + u(x^j).$$

Damit gilt für die i -te Zeile von $dw^j = \frac{\partial w(u, x^j)}{\partial u} \Big|_{u=\Theta(id)=0}$ mit $u^i = u(x^i)$:

$$dw_i^j = \frac{\partial w(u, x^j)}{\partial u^i} \Big|_{u=\Theta(id)=0} = \frac{\partial (x^j + u(x^j))}{\partial u^i} \Big|_{u=0} = \delta_{i,j}.$$

Somit ist $dw^j = I$, wobei I die $d \times d$ Einheitsmatrix ist.

Die Lie-Algebra der Lie-Gruppe der Diffeomorphismen enthält alle Verrückungsfelder, was bedeutet, dass die gleiche Basis für die Lie-Algebra und den Einbettungsraum verwendet werden kann und damit ist $e_\Theta = I$. Also ergibt sich für die Ableitung

$$dr_j = -\nabla \mathcal{T}_y(x^j).$$

Der diffeomorphe Dämonen-Algorithmus ist in Algorithmus 4 dargestellt.

Algorithmus 4 Der diffeomorphe Dämonen-Algorithmus

Eingabe: zwei diskrete Bilder R, T an Datenpunkten $x^1, \dots, x^m \in \mathbb{R}^d$, $m \in \mathbb{R}$, Startwert y , Standardabweichung $\sigma \in \mathbb{R}$

Ausgabe: Transformation y

while !stop **do**

 Berechne $\mathcal{R}, \mathcal{T}_y$ durch Interpolation

 Berechne Update u durch Minimierung von:

$$E_d(u) = \|R - T(y \circ \exp(u))\|^2 + \frac{\alpha_1^2}{\alpha_2^2} \|u\|^2$$

 Fluidale Regularisierung: $u \leftarrow G_\sigma^0 * u$

 Update-Schritt: $y \leftarrow y \circ \exp(u)$

 Diffusive Regularisierung: $y \leftarrow G_\sigma^0 * y$

end while

In diesem Kapitel wurde beschrieben, wie der Dämonen-Algorithmus mit der Bildregistrierungsmethode basierend auf Lie-Gruppen als ein Verfahren zur Lösung eines Optimierungsproblems interpretiert werden kann. Durch die Wahl des Gauß-Newton-Verfahrens zur Optimierung konnte jeweils für die additive und die kompositive Update-Regel eine weitere Variante der Dämonenkräfte angegeben werden [51, 49, 47]. Zusätzlich wurde eine diffeomorphe Erweiterung des Dämonen-Algorithmus [51] beschrieben, die die Lie-Gruppen-Struktur der Diffeomorphismen ausnutzt. Im nächsten Kapitel werden die verschiedenen Varianten des Dämonen-Algorithmus an einem synthetischen Beispiel getestet.

Kapitel 7

Ergebnisse

In diesem Kapitel wird der Dämonen-Algorithmus mit seinen verschiedenen Varianten an einem (synthetischen) Beispiel getestet. Verwendet wurde das *Insight Segmentation and Registration Toolkit* (ITK), welches eine Open-Source C++-Klassenbibliothek zur Segmentierung und Registrierung von Bildern bereitstellt [23]. In [50] wurde eine Implementierung des Dämonen-Algorithmus vorgestellt, die in das ITK-Framework eingebettet wurde.

Dieses Kapitel gliedert sich in zwei Abschnitte. Im ersten wird ein Überblick über die Implementierung des Dämonen-Algorithmus gegeben und im zweiten werden die Ergebnisse einiger Experimente mit den verschiedenen Varianten des Dämonen-Algorithmus erläutert. Die drei Update-Regeln, additiv, kompositiv und diffeomorph, werden in Kombination mit vier verschiedenen Dämonenkräften miteinander verglichen.

7.1 ITK-Implementierung des Dämonen-Algorithmus

Die in dieser Arbeit verwendete Implementierung des Dämonen-Algorithmus ist Teil des ITK-Frameworks [50, 23]. Es werden viele verschiedene Klassen verwendet. Im Folgenden sollen aber nur einige erwähnt werden, die zu der Berechnung der Update-Regeln und Dämonenkräfte benötigt werden. Die zwei wichtigsten Klassen sind der `DiffeomorphicDemonsRegistrationFilter` und der `FastSymmetricForcesDemonsRegistrationFilter`. Die Klasse `FastSymmetricForcesDemonsRegistrationFilter` registriert zwei Bilder mit dem additiven und die Klasse `DiffeomorphicDemonsRegistrationFilter` mit dem kompositiven und dem diffeo-

morphen Dämonen-Algorithmus. Beide Klassen basieren auf dem `DemonsRegistrationFilter` und berechnen das Update, also die Dämonenkräfte in jedem Iterationsschritt in der Funktion `DemonsRegistrationFunction`, welches mit der Funktion `ApplyUpdate` angewendet wird.

Im diffeomorphen Dämonen-Algorithmus wird das Update durch $y \leftarrow y \circ \exp(u)$ durchgeführt. Das Exponential von Vektorfeldern wird in `ApplyUpdate` durch die Klasse `ExponentialDeformationFieldImageFilter` berechnet, indem die in [2] beschriebene Methode (vgl. Kapitel 6.2) verwendet wird.

Die kompositive Update-Regel $y \leftarrow y \circ (id + u)$ ist eine lineare Approximation an die diffeomorphe Update-Regel [51, 47]:

$$y \circ \exp(u) = y \circ (id + u) + \mathcal{O}(\|u\|^2).$$

Dieses wird ausgenutzt, indem der kompositive Update-Schritt ebenfalls mit der Klasse `ExponentialDeformationFieldImageFilter` berechnet wird, jedoch mittels null Iterationen [50]. Wieviele Iterationen im diffeomorphen Update-Schritt für die Berechnung des Exponentials verwendet werden, geht nicht aus [50, 2] hervor.

Um die Effizienz des Algorithmus zu steigern, wird eine Multiresolutions-Methode [33, 38] verwendet, die in der Klasse `MultiResolutionsPDEDeformableRegistration` implementiert ist. Multiresolutionsmethoden gehören zu den Standardlösungsverfahren für Registrierungsprobleme. Auf die Vorteile, die derartige Methoden mit sich bringen, wird in dieser Arbeit nicht näher eingegangen.

Dem Dämonen-Algorithmus können mehrere Parameter übergeben werden, u.a. die Folgenden:

- die Anzahl an Iterationen für die Multiresolutions-Methode,
- die Standardabweichung σ_1 für die Glättung des Verrückungsfeldes,
- die Standardabweichung σ_2 für die Glättung der Transformation,
- die Art der Update-Regel: additiv, kompositiv oder diffeomorph,
- die Dämonenkraft, es gibt in dieser Implementierung vier Möglichkeiten dr_j^T in

$$u(x^j) = - \frac{\mathcal{R}(x^j) - \mathcal{T}(y(x^j))}{\|dr_j\|^2 + \alpha^2 (\mathcal{R}(x^j) - \mathcal{T}(y(x^j)))} dr_j^T$$



Abbildung 7.1: Die zu registrierenden Bilder: links Referenzbild, rechts Templatebild.

(vgl. Gleichung (6.7) in Kapitel 6) zu wählen: $dr_j^T = -\nabla\mathcal{R}(x^j)$, $dr_j^T = -\nabla\mathcal{T}(y(x^j))$, $dr_j^T = -\nabla\mathcal{T}_y(x^j)$ und $dr_j^T = -\frac{1}{2}(\nabla\mathcal{R}(x^j) + \nabla\mathcal{T}_y(x^j))$.

Im Folgenden werden die Update-Regeln in Kombination mit den vier Dämonenkräften miteinander verglichen.

7.2 Ergebnisse einer Bildregistrierung mit dem Dämonen-Algorithmus

In diesem Abschnitt werden anhand eines synthetischen Beispiels die verschiedenen Varianten des Dämonen-Algorithmus miteinander verglichen. Die Bilder, die registriert werden sollen, sind in Abb. 7.1 dargestellt. In Abb. 7.2 sind einige Ergebnisse der Bildregistrierung mit verschiedenen Update-Regeln in Kombination mit den vier Varianten der Dämonenkräfte abgebildet. Bis auf die Dämonenkräfte und die Update-Regel sind die Parameter in allen zwölf Registrierungen gleich gewählt worden: Für die Glättung des Verrückungsfeldes und der Transformation wurde jeweils eine Standardabweichung $\sigma_1 = \sigma_2 = 1$ gewählt. Es wurden drei Level für die Multi-Resolution mit jeweils 15, 10 und 5 Iterationsschritten festgesetzt.

Vergleich der Dämonenkräfte Getestet wurden die klassischen Kräfte ($dr_j^T = -\nabla\mathcal{R}(x^j)$, linke Spalte in Abb. 7.2), die Linearisierung des Gauß-Newton-Verfahrens im additiven Fall ($dr_j^T = -\nabla\mathcal{T}(y(x^j))$, zweite Spalte von links in Abb. 7.2), die im folgenden als additive GN-Kräfte bezeichnet werden (in [50] wird diese Linearisierung als *mapped moving image* bezeichnet), die Linearisierung des Gauß-Newton-Verfahrens im kompositiven Fall ($dr_j^T = -\nabla\mathcal{T}_y(x^j)$, zweite Spalte von rechts in Abb. 7.2), im Folgenden kompositive GN-Kräfte (in

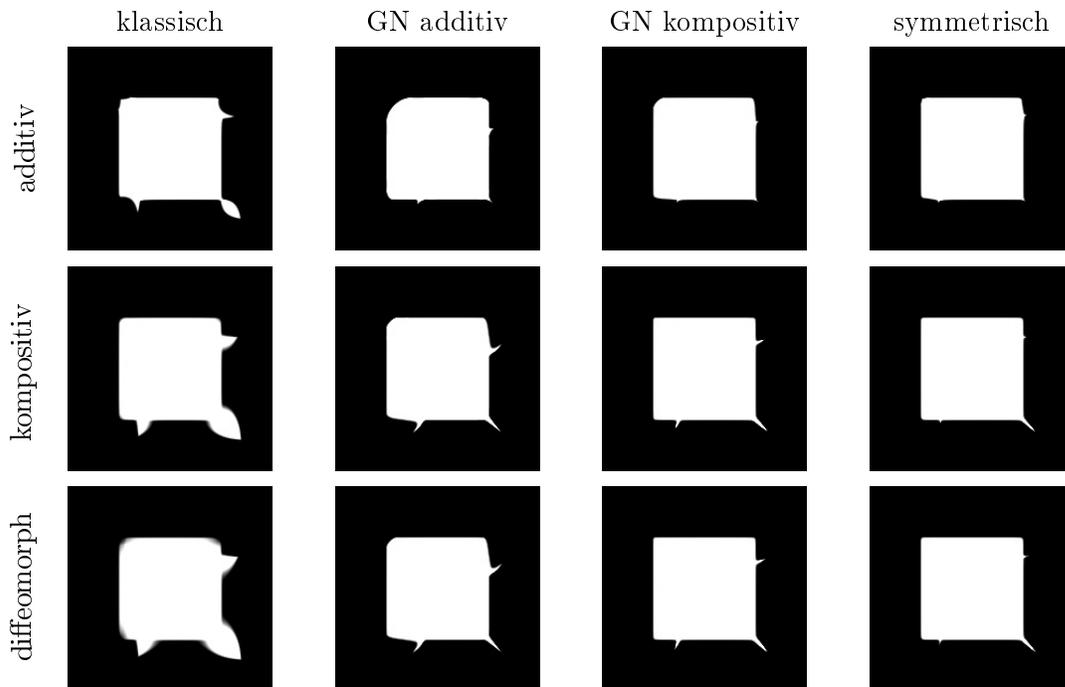


Abbildung 7.2: Ergebnisse der Registrierung mit dem Dämonen-Algorithmus mit einer Glättung des Verrückungsfeldes mit $\sigma_1 = \sigma_2 = 1$ und drei Multi-Resolutionsleveln mit 15, 10 und 5 Iterationen. Die Zeilen gehören zu den unterschiedlichen Update-Regeln und die Spalten zu den Dämonenkräften: Erste Spalte: $dr_j^T = -\nabla\mathcal{R}(x^j)$; zweite Spalte: $dr_j^T = -\nabla\mathcal{T}(y(x^j))$; dritte Spalte: $dr_j^T = -\nabla\mathcal{T}_y(x^j)$; vierte Spalte: $dr_j^T = -\frac{1}{2}(\nabla\mathcal{R}(x^j) + \nabla\mathcal{T}_y(x^j))$.

[50] *warped moving image*), und die symmetrischen Kräfte im kompositiven Fall ($dr_j^T = -\frac{1}{2}(\nabla\mathcal{R}(x^j) + \nabla\mathcal{T}_y(x^j))$, rechte Spalte in Abb. 7.2). Die symmetrischen Kräfte für den additiven Fall ($dr_j^T = -\frac{1}{2}(\nabla\mathcal{R}(x^j) + \nabla\mathcal{T}(y(x^j)))$) sind nicht in der hier verwendeten ITK-Methode implementiert.

Für alle drei Update-Regeln (in Abb. 7.2: erste Reihe additiv, zweite Reihe kompositiv, dritte Reihe diffeomorph) wird deutlich, dass die visuelle Ähnlichkeit des transformierten Bildes zu dem Referenzbild von links nach rechts zunimmt. Das bedeutet, dass bei der gleichen Anzahl an Iterationsschritten die klassischen Kräfte die visuell schlechtesten und die symmetrischen die visuell besten Ergebnisse erzielen. Dies konnte auch erwartet werden, da die klassischen Kräfte mit keinem Optimierungsverfahren, die Gauß-Newton-Kräfte mit dem Gauß-Newton-Verfahren auf Lie-Gruppen, das lineare Konvergenz aufweist [49], und die symmetrischen mit dem ESM-

Verfahren auf Lie-Gruppen, das annähernd quadratische Konvergenz aufweist [49, 31], in Beziehung stehen.

Auffallend ist jedoch, dass trotz dieses relativ einfachen Registrierungsproblems (es hat nur eine rigide Verrückung stattgefunden), mit keiner Variante eine optimale Transformation gefunden wurde. Dies hängt damit zusammen, dass bei dem optischen Fluss die Annahme getroffen wird, dass nur eine kleine Verrückung stattgefunden hat. Die Verrückung zwischen Referenz- und Templatebild ist relativ groß, so dass keine optimalen Ergebnisse zu erwarten sind. Verbessert werden können diese durch die Erhöhung der Anzahl der Multiresolutionslevel. Eine Verrückung, die beispielsweise bei einer feinen Auflösung noch über zehn Pixel/Voxel beträgt, kann bei einer groben Auflösung nur einen Pixel/Voxel betragen. Dies wäre wiederum eine kleine Verrückung und der Dämonen-Algorithmus kann auf dem groben Level gute Ergebnisse erzielen. Ohne Mutliresolutions-Methode kann der Dämonen-Algorithmus mehr als 300 Iterationen benötigen, um visuell ähnlich gute Ergebnisse zu erzielen. Weitere Experimente in dieser Richtung wurden jedoch nicht durchgeführt.

Vergleich der Update-Regeln Beim Vergleich der Update-Regeln (Vergleich der Zeilen in Abb. 7.2) fällt auf, dass mit dem additiven Dämonen-Algorithmus (Abb. 7.2, obere Zeile) zumindest visuell bessere Ergebnisse erzielt werden, als mit dem kompositiven oder diffeomorphen Algorithmus. Würde man neben den visuellen Ergebnissen zusätzlich die Transformationen vergleichen, würde auffallen, dass die durch den additiven Dämonen-Algorithmus berechneten weit weniger glatt sind als die der anderen. Eine größere Glattheit kann durch Erhöhung der Standardabweichungen erreicht werden, was jedoch in einer Verschlechterung des visuellen Ergebnisses resultieren würde. Hierzu wurden keine weiteren Experimente gemacht, da dies den Rahmen der Arbeit überschreiten würde.

Zusätzlich ist auffallend, dass die Ergebnisse mit der kompositiven Update-Regel (Abb. 7.2, mittlere Zeile) sehr ähnlich sind zu den Ergebnissen mit der diffeomorphen Update-Regel (Abb. 7.2, untere Zeile). Dies ist dadurch zu erklären, dass die kompositive Update-Regel eine lineare Approximation an die diffeomorphe ist (s.o.). Da im Dämonen-Algorithmus kleine Verrückungen angenommen werden, kann eine lineare Approximation eine gute Annäherung sein und gute Ergebnisse liefern.

In diesem Kapitel wurde das ITK-Framework [23] und der darin implementierte Dämonen-Algorithmus [50] benutzt, um die Unterschiede zwischen den verschiedenen Varianten des

Algorithmus aufzuzeigen. An einem synthetischen Beispiel konnte gezeigt werden, dass die symmetrischen Dämonenkräfte die besten visuellen Ergebnisse liefern. Die Verwendung der kompositiven und diffeomorphen Update-Regel führt zu visuell sehr ähnlichen Ergebnissen. Auf die Darstellung der berechneten Transformationen wurde hier verzichtet. Für eine genauere Evaluation des Dämonen-Algorithmus mit den unterschiedlichen Varianten siehe [47].

Kapitel 8

Zusammenfassung und Diskussion

In diesem Kapitel wird eine kurze Zusammenfassung der Arbeit gegeben und es werden einige Fragen diskutiert, die im Zusammenhang mit der Optimierung auf Lie-Gruppen, dem Dämonen-Algorithmus und seiner diffeomorphen Erweiterung noch offen geblieben sind.

8.1 Zusammenfassung

Typischerweise werden Registrierungsprobleme als Optimierungsprobleme interpretiert und eine Zielfunktion wird bezüglich einer Transformation minimiert. In dieser Arbeit wurde eine Registrierungsmethode beschrieben, die unter der Verwendung von Lie-Gruppen ein gegebenes Distanzmaß minimiert. Der Hauptteil dieser Arbeit bestand darin, die Theorie der Optimierungsverfahren auf Lie-Gruppen aufzuarbeiten, auf die in [49, 47] nicht näher eingegangen wurde. Ausführlich wurden in dieser Arbeit ausgesuchte Begriffe aus der Differentialgeometrie und der Lie-Gruppen-Theorie vorgestellt. Es wurden differenzierbare Mannigfaltigkeiten und Begriffe wie Tangentialvektoren, Vektorfelder und Differentiale eingeführt, mit denen die Beschreibung der Lie-Algebra einer Lie-Gruppe möglich war. Diese bildet den Raum der Suchrichtungen der Optimierungsverfahren auf Lie-Gruppen. Um eine Verknüpfung zwischen Lie-Gruppe und Lie-Algebra herzustellen, wurde die Exponentialabbildung eingeführt [20, 26, 45]. Die Differenzierbarkeit einer Funktion auf einer Lie-Gruppe wurde geklärt und eine Taylorentwicklung angegeben [29, 45].

Mit dieser theoretischen Vorarbeit konnte das Gauß-Newton-Verfahren auf Lie-Gruppen [49]

beschrieben werden, in dessen Update-Schritt die aktuelle Iterierte y durch $y \leftarrow y \circ \exp(u)$ mit der berechneten Suchrichtung u verknüpft wird. Um die Optimierung auf Lie-Gruppen zu veranschaulichen, wurde sie am Beispiel der Lie-Gruppe der rigiden Transformationen beschrieben.

Zusätzlich wurde der Dämonen-Algorithmus von Jean-Philippe Thirion [44, 43] näher betrachtet. Dieser ist ein Algorithmus zur Bildregistrierung, basiert jedoch nicht auf einem Optimierungsproblem. Die Bildregistrierung wird vielmehr als Diffusionsprozess interpretiert und die Dämonen verschieben die Bildpunkte eines Templatebildes entsprechend der Dämonenkräfte. Iterativ werden im Dämonen-Algorithmus für jeden Punkt im Template die Dämonenkräfte berechnet und das daraus resultierende Verrückungsfeld wird mit einer Gauß-Funktion geglättet und mit der aktuellen Iterierten verknüpft. Es wurden unterschiedliche Dämonenkräfte (klassisch und symmetrisch) und Verknüpfungsmethoden (additiv, kompositive und diffeomorph) vorgestellt.

Es konnte gezeigt werden [49, 47], dass unter Verwendung der Registrierungsmethode basierend auf Lie-Gruppen die Minimierung einer speziellen Zielfunktion in jedem Iterationsschritt des Optimierungsverfahren zu der Berechnung eines Ausdrucks führt, der ähnlich zu den Dämonenkräften ist [49, 47].

Im Falle des additiven Dämonen-Algorithmus bilden die betrachteten Abbildungen, die Vektorfelder auf dem \mathbb{R}^d , einen Vektorraum und damit eine triviale Lie-Gruppe und der Update-Schritt $y \leftarrow y \circ \exp(u)$ vereinfacht sich zu $y \leftarrow y + u$. Wird eine spezielle Zielfunktion mit dem Gauß-Newton-Verfahren minimiert, führt dies zu der Berechnung eines zu den Dämonenkräften ähnlichen Ausdrucks. Analog gilt dies für die kompositiven Dämonen mit $y \leftarrow y \circ (id + u)$. Somit kann der Dämonen-Algorithmus als Lösungsverfahren eines Optimierungsproblems interpretiert werden [49, 47].

Zusätzlich wurde eine diffeomorphe Erweiterung des Dämonen-Algorithmus vorgestellt, die darauf beruht, dass über der Lie-Gruppe der Diffeomorphismen minimiert wird [51]. Diese wurde in dieser Arbeit nur kurz beschrieben und noch nicht alle Fragen diesbezüglich geklärt (vgl. Abschnitt 8.2).

8.2 Diskussion und Ausblick

In diesem Abschnitt werden einige Aspekte des Dämonen-Algorithmus diskutiert und es werden einige Fragen beschrieben, die in Zusammenhang mit der Optimierung auf Lie-Gruppen aufgekommen sind, und noch nicht ausreichend geklärt werden konnten.

Es wurde beschrieben, wie Lie-Gruppen in der Registrierung eingesetzt werden können. Die Vorteile hierbei sind, dass durch deren Struktur in den Optimierungsverfahren keine Nebenbedingungen benötigt werden und durch die Exponentialabbildung eine Projektion von der Lie-Algebra in die Lie-Gruppe existiert, die in der Theorie gut studiert ist. In dieser Arbeit wurden jedoch nur die Eigenschaften der Exponentialabbildung eingeführt, die für das Verständnis des Gauß-Newton-Verfahrens auf Lie-Gruppen erforderlich waren.

Registrierungsmethoden unter Verwendung von Lie-Gruppen können nur sinnvoll eingesetzt werden, wenn die gesuchte Transformation Element einer Lie-Gruppe ist. In dieser Arbeit wurde die Lie-Gruppe der rigiden Transformationen genauer betrachtet. Neben dieser bilden noch andere Transformationen, die durch Matrizen dargestellt werden können, Lie-Gruppen. Beispiele für die rigide Registrierung auf Lie-Gruppen sind die Gruppe $SE(2)$ und die Gruppe der Rotationsmatrizen, für affine Registrierung auf Lie-Gruppen die Gruppe $GL(n, \mathbb{R})$ der invertierbaren Matrizen und für rigide volumenerhaltene Registrierung auf Lie-Gruppen die Gruppe $SL(n, \mathbb{R})$ der invertierbaren Matrizen mit einer Determinante von eins [55].

Eine weit verbreitete Methode, eine Zielfunktion bezüglich derartiger Transformationen zu minimieren, ist, eine Parametrisierung durchzuführen. Die Zielfunktion wird dann bezüglich der Parameter der Transformation (im rigiden Fall Rotationswinkel und Translationsvektor) optimiert [33, 38, 30, 13]. Ein nächster Schritt wäre, die beiden Methoden miteinander zu vergleichen, um zu überprüfen, ob die Optimierung auf Lie-Gruppen und die Optimierung über den Parameterräumen dieselben Ergebnisse liefern.

Der Dämonen-Algorithmus ist ein Verfahren zur Bildregistrierung, dass in der Praxis vor allem wegen seiner einfachen Implementierung und seiner Effizienz viel Anwendung gefunden hat. Er hat jedoch auch einige Nachteile, was hauptsächlich auf die nicht intuitive Wahl der Parameter und auf die Motivation über ein Diffusionsmodell und nicht über ein Optimierungsproblem zurückzuführen ist.

Im Dämonen-Algorithmus wird abwechselnd ein Verrückungsfeld mit den Dämonenkräften

berechnet und mit Gauß-Funktionen geglättet. In [49] wurde die Glättung der Verrückung u als eine Approximation an eine fluidale und die Glättung der Transformation $y = id + u$ als eine Approximation an eine diffusive Regularisierung interpretiert. Die Regularisierung mit Gauß-Funktionen wurde in dieser Arbeit nicht thematisiert, so dass noch zu klären ist, warum und wie weit diese Interpretation zutrifft.

Die Ergebnisse des Dämonen-Algorithmus sind sehr sensitiv gegenüber der Änderung der Standardabweichung σ der Gauß-Funktion. In den Werken [49, 47, 39, 44], die für diese Arbeit verwendet wurden, werden jedoch keine Angaben darüber gemacht, wie die Standardabweichung σ sinnvoll gewählt werden kann. In der ITK-Implementierung [50] wurde als Standardwahl $\sigma = 1$ jeweils für die Glättung der Verrückung und der Transformation gewählt. Dieses ist jedoch nicht für jede Anwendung eine sinnvolle Wahl.

Der Dämonen-Algorithmus ist ein Iterationsverfahren, es werden jedoch keine Abbruchbedingungen vorgeschlagen. Die einzige Abbruchbedingung ist die maximale Anzahl an Iterationsschritten. So werden womöglich zu viele oder zu wenige Iterationsschritte gemacht, was in einer längeren Rechenzeit oder schlechteren Ergebnissen resultieren könnte. Es kann nicht garantiert werden, dass das optimale Ergebnis gefunden wurde.

Für den Dämonen-Algorithmus gibt es außerdem keine Angaben zu Fehlerabschätzungen. Durch die Interpretation der Dämonenkräfte als Suchrichtungen eines Gauß-Newton-Verfahrens könnten die Fehlerabschätzungen des Optimierungsverfahrens auf den Dämonen-Algorithmus übertragen werden.

In dieser Arbeit lag der Fokus auf der Theorie der Lie-Gruppen und auf der Bildregistrierung unter Verwendung von Lie-Gruppen. In dem Registrierungsrahmen [49], der auf der Optimierung auf Lie-Gruppen beruht, wurde die Einbettung einer Lie-Gruppe in einen euklidischen Raum benutzt, um Ableitungen von Funktionen auf Lie-Gruppen bestimmen zu können (vgl. Gleichung (5.4)). Dafür muss jedoch die Differenzierbarkeit der Einbettung und der Abbildung, die die Darstellung der Transformation eines Punktes im Einbettungsraum realisiert, vorausgesetzt werden. Möglicherweise wird dieses durch die Theorie der Einbettungen bereitgestellt. Es lag jedoch nicht im Rahmen dieser Arbeit, sich mit Einbettungen zu beschäftigen, so dass dieser Teil ungeklärt geblieben ist.

Mit Hilfe des Registrierungsrahmens basierend auf der Optimierung auf Lie-Gruppen [49], ist es gelungen, den Dämonen-Algorithmus als Lösungsverfahren eines Optimierungsproblems zu

interpretieren. Bei den additiven Dämonen wird auf dem Vektorraum der Vektorfelder des \mathbb{R}^d minimiert, der eine triviale Lie-Gruppe mit der Addition als Gruppenverknüpfung und der Identitätsabbildung als Exponentialabbildung bildet. Im Fall der kompositiven Dämonen wird die Menge der nicht-parametrischen Transformationen mit der Komposition von Funktionen als Gruppenverknüpfung betrachtet. Nicht alle dieser Transformationen sind invertierbar, so dass sie keine Gruppe bilden, sondern nur ein Monoid [47] (ein Monoid besitzt die Eigenschaften einer Gruppe bis auf das invertierbare Element [46]). Deswegen ist erstmal nicht klar, dass die Lie-Gruppen-Methoden auf diese Menge angewendet werden können. In [47] wird argumentiert, dass die Invertierbarkeit in den Optimierungsmethoden auf Lie-Gruppen nicht ausgenutzt wird und der Registrierungsrahmen deswegen auch bei den kompositiven Dämonen angewendet werden kann.

Die spezielle Lie-Gruppen-Struktur des Registrierungsrahmens basierend auf Lie-Gruppen [49] wurde von dem Dämonen-Algorithmus nicht verwendet. Die Bildregistrierung unter Verwendung von Lie-Gruppen bietet einen wesentlich stärkeren theoretischen Hintergrund, als für den Dämonen-Algorithmus von Bedeutung ist. Erst mit der diffeomorphen Erweiterung des Dämonen-Algorithmus wurde die Lie-Gruppen-Struktur des Registrierungsrahmens ausgenutzt. Die Idee des diffeomorphen Dämonen-Algorithmus ist, mit den Optimierungsverfahren auf Lie-Gruppen auf der Menge der Diffeomorphismen zu optimieren. Die Menge der Diffeomorphismen bildet keinen Vektorraum, u.a. weil sie bezüglich der Addition nicht abgeschlossen ist. Sie bildet mit der Komposition \circ eine Gruppe [54]. Ob diese Gruppe eine Lie-Gruppe ist, ist jedoch noch unklar geblieben. Diffeomorphismen sind differenzierbar und können verknüpft und invertiert werden und erhalten diese Eigenschaft. Alle Diffeomorphismen bilden aber keine Lie-Gruppe. In [3] heißt es, dass nur perfekte, also C^∞ -Diffeomorphismen, eine Lie-Gruppe bilden, in [12] jedoch, dass Diffeomorphismen unendlich dimensionale Gruppen und differenzierbare Mannigfaltigkeiten bilden, aber keine Lie-Gruppen. Um diesbezüglich weitere Aussagen treffen zu können, müsste die Theorie der Gruppe der Diffeomorphismen genauer betrachtet werden. Dieses hätte aber den Rahmen dieser Arbeit überschritten.

Im diffeomorphen Dämonen-Algorithmus werden diskrete Diffeomorphismen betrachtet und keine C^∞ -Diffeomorphismen. Demzufolge ist zu erwarten, dass die durch ihn berechneten Transformationen allenfalls approximativ Diffeomorphie-Eigenschaften besitzen und damit nicht unbedingt invertierbar sind. Dieses ist jedoch nur eine Annahme und wurde nicht in

der Praxis überprüft.

Diese Arbeit hat einen ersten Einblick in Bildregistrierungsmethoden unter Verwendung von Lie-Gruppen gegeben. Ein nächster Schritt ist, diese Methoden mit der parametrischen Registrierung [33, 30] zu vergleichen.

Anhang A

Anhang

A.1 Nomenklatur

d	Dimension (üblicherweise $d \in \{2, 3\}$)
$\Omega \subset \mathbb{R}^d$	Gebiet, Definitionsbereich eines Bildes
$x = (x_1, \dots, x_d)^T$	Punkt (Pixelkoordinate) $x \in \Omega$
\mathcal{R}, \mathcal{T}	kontinuierliches Referenz- und Templatebild, $\mathcal{R}, \mathcal{T} : \Omega \subset \mathbb{R}^d \rightarrow \mathbb{R}$
R, T	diskretes Bilder \mathcal{R}, \mathcal{T} an den Punkten $x^1, \dots, x^m \in \mathbb{R}^d$ mit $R = (\mathcal{R}(x^1), \dots, \mathcal{R}(x^m))^T$ und $T = (\mathcal{T}(x^1), \dots, \mathcal{T}(x^m))^T$
$y = (y_1, \dots, y_d)^T$	Transformation, $y : \mathbb{R}^d \rightarrow \mathbb{R}^d, x \mapsto y(x)$
$u = (u_1, \dots, u_d)^T$	Verrückung, $u : \mathbb{R}^d \rightarrow \mathbb{R}^d, x \mapsto u(x)$
$y = (y(x^1)^T, \dots, y(x^m)^T)^T$	diskrete Transformation an den Punkten $x^1, \dots, x^m \in \mathbb{R}^d$
$\mathcal{T}_y(x) = \mathcal{T}(y(x))$	transformiertes Templatebild
$T(y)$	diskretes transformiertes Templatebild
$d\mathcal{T}, \nabla \mathcal{T}$	Ableitung und Gradient von \mathcal{T} , $d\mathcal{T}^T = \nabla \mathcal{T}$
$d\mathcal{T}_y, \nabla \mathcal{T}_y$	Ableitung und Gradient von $\mathcal{T} \circ y$, $d\mathcal{T}_y^T = dy^T d\mathcal{T}^T = \nabla \mathcal{T}_y$
$\mathcal{D}(\mathcal{T}_y, \mathcal{R})$	Distanzmaß
$\mathcal{S}(y)$	Regularisierer
$\partial_i g = \frac{\partial g}{\partial x_i}$	partielle Ableitung von $g : \mathbb{R}^d \rightarrow \mathbb{R}$ nach x_i
$\partial_{i,j}^2 g = \frac{\partial^2 g}{\partial x_i \partial x_j}$	zweite partielle Ableitung von $g : \mathbb{R}^d \rightarrow \mathbb{R}$ nach x_i und x_j
$\nabla g = \left(\frac{\partial g}{\partial x_1}, \dots, \frac{\partial g}{\partial x_d} \right)^T$	Gradient von $g : \mathbb{R}^d \rightarrow \mathbb{R}$

$\alpha, \alpha_1, \alpha_2 \in \mathbb{R}^+$	Parameter
$\ x\ , x \in \mathbb{R}^d$	euklidische Norm von x : $\ x\ = \sqrt{\sum_{i=1}^d x_i^2}$
$(g_1 * g_2)(x)$	Faltung von zwei Funktionen $g_1, g_2 : \mathbb{R}^d \rightarrow \mathbb{R}$ $(g_1 * g_2)(x) = \int_{\mathbb{R}^d} g_1(y)g_2(x-y)dy$
$f(x) \in \mathbb{R}^d$	Dämonenkraft im Punkt $x \in \mathbb{R}^d$
$G_\sigma^\mu : \mathbb{R}^d \rightarrow \mathbb{R}^d$	d-dim. Gauß-funktion, $G_\sigma^\mu = \frac{1}{(2\pi)^{\frac{d}{2}} \sigma^d} \exp(-\frac{\ x-\mu\ _2^2}{2\sigma^2})$
$u^{\text{Opt}} : \Omega \rightarrow \mathbb{R}^d$	Verrückung der Dämonen, basierend auf dem optischen Fluss
$u^{\text{Sym}} : \Omega \rightarrow \mathbb{R}^d$	Verrückung der symmetrischen Dämonen
\mathcal{M}, \mathcal{N}	differenzierbare Mannigfaltigkeiten
(\mathcal{U}, φ)	Karte auf \mathcal{M} , $\varphi : \mathcal{U} \subset \mathcal{M} \rightarrow \Omega \subset \mathbb{R}^n$
$\varphi(p) = (u_1, \dots, u_n)$	lokale Koordinaten von $p \in \mathcal{M}$ bzgl. der Karte φ
$C^\infty(\mathcal{M})$	Raum der stetig differenzierbaren Funktionen auf \mathcal{M} : $C^\infty(\mathcal{M}) = \{f : \mathcal{M} \rightarrow \mathbb{R} \text{ und } f \text{ stetig diffbar}\}$
$\frac{\partial}{\partial u_i}(f \circ \varphi^{-1}) \Big _{\varphi(p)}$	partielle Ableitung der Interpretation von f in der Karte φ
$c : [-\varepsilon, \varepsilon] \subset \mathbb{R} \rightarrow \mathcal{M}$	Kurve auf \mathcal{M}
\bar{c}	Tangentialvektor bzgl Definition 3.7
$\xi_p : C^\infty(\mathcal{M}) \rightarrow \mathbb{R}$	Tangentialvektor von \mathcal{M} in $p \in \mathcal{M}$ bzgl Definition 3.8
$T_p\mathcal{M}$	Tangentialraum von \mathcal{M} im Punkt $p \in \mathcal{M}$
$\mathcal{T}\mathcal{M}$	Tangentialbündel von \mathcal{M} : $\mathcal{T}\mathcal{M} = \bigcup_{p \in \mathcal{M}} T_p\mathcal{M}$
$X : \mathcal{M} \rightarrow \mathcal{T}\mathcal{M}$	Vektorfeld auf \mathcal{M} bzgl Definition 3.10
$X : C^\infty(\mathcal{M}) \rightarrow C^\infty(\mathcal{M})$	Vektorfeld auf \mathcal{M} bzgl Definition 3.11
$\mathfrak{X}(\mathcal{M})$	Menge der glatten Vektorfelder auf \mathcal{M}
$[X, Y](f)$	Vektorfeld-Kommutator: $[\cdot, \cdot] : \mathfrak{X}(\mathcal{M}) \times \mathfrak{X}(\mathcal{M}) \rightarrow \mathfrak{X}(\mathcal{M}), [X, Y](f) = X(Y(f)) - Y(X(f))$
DF	Differential von $F : \mathcal{M} \rightarrow \mathcal{N}$: $DF : \mathcal{T}\mathcal{M} \rightarrow \mathcal{T}\mathcal{N}$
$(DF)_p$	Differential von F im Punkte $p \in \mathcal{M}$: $(DF)_p : T_p\mathcal{M} \rightarrow T_{F(p)}\mathcal{N}$
\mathcal{G}	Lie-Gruppe
\mathfrak{g}	Lie-Algebra der Lie-Gruppe \mathcal{G}
$[\cdot, \cdot] : \mathfrak{g} \times \mathfrak{g} \rightarrow \mathfrak{g}$	Lie-Klammer
$L_\sigma : \mathcal{G} \rightarrow \mathcal{G}$	Linksverknüpfung: $L_\sigma(\tau) = \sigma * \tau$, für $\sigma, \tau \in \mathcal{G}$
$\mathfrak{X}^*(\mathcal{G})$	Menge der linksinvarianten Vektorfelder

$\exp : \mathfrak{g} \rightarrow \mathcal{G}$	Exponentialabbildung der Lie-Gruppe \mathcal{G}
$df_\sigma(0)$	Ableitung von f bzgl. den Koordinaten einer festen Karte: $(df_\sigma(0))_i = \left. \frac{\partial}{\partial u_i} f(\sigma * \exp(u)) \right _{u=0}$
$u \in \mathfrak{g}$	Element der Lie-Algebra \mathfrak{g} und Suchrichtung des Gauß-Newton-Verfahrens auf Lie-Gruppen
$\Theta : \mathcal{G} \rightarrow \mathbb{R}^N$	Einbettung einer Lie-Gruppe \mathcal{G} in einen \mathbb{R}^N
$w(\Theta(y), x)$	Darstellung der Transformation von x durch y in Einbettung Θ

A.2 Berechnungen

Eine Lösung der Gleichung des optischen Flusses (Gleichung (2.2))

Lemma A.1 *Das v mit minimaler Länge, das die Gleichung des optischen Flusses*

$$v(x) \cdot \nabla \mathcal{R}(x) = \mathcal{T}_y(x) - \mathcal{R}(x)$$

für $\nabla \mathcal{R} > 0$ erfüllt, ist

$$v(x(t_1)) = \frac{(\mathcal{T}_y(x(t_1)) - \mathcal{R}(x(t_1)))}{\|\nabla \mathcal{R}(x(t_1))\|^2} \nabla \mathcal{R}(x(t_1)).$$

Beweis:

Gesucht ist ein $v(x)$ minimaler Länge, das die Gleichung des optischen Flusses

$$v(x) \cdot \nabla \mathcal{R}(x) = \mathcal{T}_y(x) - \mathcal{R}(x).$$

erfüllt. Für jeden Punkt $x \in \mathbb{R}^d$ muss also mit $v = v(x) \in \mathbb{R}^d$, $r = \nabla \mathcal{R}(x) \in \mathbb{R}^d$ und $s = \mathcal{T}_y(x) - \mathcal{R}(x) \in \mathbb{R}$ gelten:

Finde ein v mit $r^T v = s$, so dass $\|v\| \stackrel{!}{=} \min$.

Jeder Vektor v kann dargestellt werden durch $v = \xi \cdot r + \eta \cdot r^\perp$, wobei r^\perp orthogonal zu r ist.

Es gilt

$$\begin{aligned} s &= r^T v = \xi \cdot r^T r + \underbrace{\eta \cdot r^T r^\perp}_{=0} \\ &= \xi \cdot r^T r \\ &\Rightarrow \xi = \frac{s}{r^T r} \quad (r^T r \in \mathbb{R} \text{ und } r^T r \neq 0). \end{aligned}$$

Es wird ein v mit minimaler Länge gesucht. Für die Länge von v gilt

$$\|v\|^2 \leq \|\xi \cdot r\|^2 + \|\eta \cdot r^\perp\|^2.$$

Ist $\eta = 0$, ist die Länge minimal. Also ist das gesuchte v

$$v = \xi \cdot r = \frac{s r}{r^T r} = \frac{s r}{\|r\|^2} = \frac{\mathcal{T}_y(x) - \mathcal{R}(x)}{\|\nabla \mathcal{R}(x)\|^2} \nabla \mathcal{R}(x).$$

□

Berechnung von (2.5)

Mit den Abkürzungen $\delta := R(x) - T(y(x))$ und $J := \nabla_x R(x)$ und $\alpha > 0$ ist

$$u(x) = \frac{\delta J}{\|J\|^2 + \alpha^2 \delta^2} \quad \text{und} \quad \|u(x)\| = \frac{|\delta| \|J\|}{\|J\|^2 + \alpha^2 \delta^2}$$

und es gilt:

$$\begin{aligned} & (\alpha|\delta| - \|J\|)^2 \geq 0 \\ \Leftrightarrow & \|J\|^2 + \alpha^2 \delta^2 - 2\alpha|\delta| \|J\| \geq 0 \\ \Leftrightarrow & \|J\|^2 + \alpha^2 \delta^2 \geq 2\alpha|\delta| \|J\| \\ \Leftrightarrow & \frac{1}{2\alpha} \geq \frac{|\delta| \|J\|}{\|J\|^2 + \alpha^2 \delta^2} \\ \Leftrightarrow & \frac{1}{2\alpha} \geq \|u(x)\|. \end{aligned}$$

Literaturverzeichnis

- [1] ABSIL, P.-A. ; MAHONY, R. ; SELPUCHRE, R.: *Optimization Algorithms on Matrix Manifolds*. Princeton University Press, 2007.
- [2] ARSIGNY, V. ; COMMOWICK, O. ; PENNEC, X. ; AYACHE, N.: *A log-euclidean framework for statistics on diffeomorphisms*. In: Proc. International Conference on Medical Image Computing and Computer Assisted Intervention (MICCAI), 2006, S. 924–931.
- [3] ASHBURNER, J.: *A fast diffeomorphic image registration algorithm*. In: NeuroImage 38 (2007), Nr. 1, S. 95 – 113.
- [4] BLOCH, E. D.: *A First Course in Geometric Topology and Differential Geometry*. 1. Auflage. Birkhäuser Boston, 1996.
- [5] BLUNDELL, S. J. ; BLUNDELL, K. M.: *Concepts in Thermal Physics*. 2. Auflage. Oxford University Press, USA, 2009.
- [6] BOOR, C. D.: *A Practical Guide to Splines*. Springer, 2001.
- [7] BRO-NIELSEN, M. ; GRAMKOW, C.: *Fast fluid registration of medical images*. In: HÖHNE, Karl (Hrsg.) ; KIKINIS, Ron (Hrsg.): *Visualization in Biomedical Computing*. Springer Berlin / Heidelberg, 1996. (Lecture Notes in Computer Science)
- [8] BROIT, C.: *Optimal registration of deformed images*, Department of Computer and Information Science, University of Pennsylvania, Diss., 1981.
- [9] BURG, K. ; HAF, H. ; WILLE, F.: *Höhere Mathematik für Ingenieure, 5 Bde., Bd.4, Vektoranalysis und Funktionentheorie*. Teubner Verlag, 1994.

-
- [10] CACHIER, P. ; BARDINET, E. ; DORMONT, D. ; PENNEC, X. ; AYACHE, N.: *Iconic feature based nonrigid registration: the PASHA algorithm*. In: *Comp. Vision and Image Understanding* 89 (2003), Feb.-Mar, Nr. 2-3, S. 272–298.
- [11] CAHILL, N. D. ; NOBLE, J. A. ; HAWKES, D. J.: *Demons algorithms for fluid and curvature registration*. In: *Proceedings of the 2009 IEEE International Symposium on Biomedical Imaging*, 2009, S. 730–733.
- [12] COOTES, T. F. ; MARSLAND, S. ; TWINING, C. J. ; SMITH, K. ; TAYLOR, C. J.: *Groupwise diffeomorphic non-rigid registration for automatic model building*. In: *ECCV (4)'04*, 2004, S. 316–327.
- [13] ELSEN, P. A. d. ; POL, E. J. D. ; VIERGEVER, M. A.: *Medical image matching-a review with classification*. In: *Engineering in Medicine and Biology Magazine, IEEE* 12 (1993), Nr. 1, S. 26–39.
- [14] FISCHER, B. ; MODERSITZKI, J.: *Fast diffusion registration*. In: *AMS Contemporary Mathematics, Inverse Problems, Image Analysis, and Medical Imaging* 313 (2002), S. 117–129.
- [15] FITZPATRICK, J. M. ; HILL, D. L. G. ; C. R. MAURER, Jr. ; SONKA, Milan (Hrsg.) ; FITZPATRICK, J. M. (Hrsg.): *Image Registration*. Bd. 2. SPIE–The International Society for Optical Engineering, 2009.
- [16] FORSTER, O.: *Vieweg Studium, Analysis 2. Differentialrechnung im \mathbb{R}^n , Gewöhnliche Differentialgleichungen*. 9. Auflage. Vieweg+Teubner Verlag, 2011.
- [17] GILL, P. E. ; MURRAY, W. ; WRIGHT, M. H.: *Practical Optimization*. Academic Press, 1982.
- [18] GOLUB, G. H. ; VAN LOAN, C. F.: *Matrix Computations (Johns Hopkins Studies in Mathematical Sciences)*. 3. Auflage. The Johns Hopkins University Press, 1996.
- [19] HABER, E. ; MODERSITZKI, J.: *Beyond mutual information: a simple and robust alternative*. In: BRAUER, W. (Hrsg.) ; MEINZER, H.-P. (Hrsg.) ; HANDELS, H. (Hrsg.) ; HORSCH, A. (Hrsg.) ; TOLXDORFF, T. (Hrsg.): *Bildverarbeitung für die Medizin 2005*. Springer Berlin Heidelberg, 2005 (Informatik aktuell), S. 350–354.

-
- [20] HALL, B. C.: *Lie Groups, Lie Algebras, and Representations: An Elementary Introduction*. 2. Auflage. Springer, 2003.
- [21] HELGASON, S.: *Differential Geometry, Lie Groups, and Symmetric Spaces (Graduate Studies in Mathematics)*. American Mathematical Society, 2001.
- [22] HORN, B. K. P. ; SCHUNK, B. G.: *Determining optical flow*. In: Artificial Intelligence 17 (1981), S. 185–203.
- [23] IBANEZ, L. ; SCHROEDER, W. ; NG, L. ; CATES, J.: *The ITK Software Guide*. 2. Auflage. Kitware, Inc., 2005.
- [24] ISERLES, A. ; MUNTKE-KAAS, H. Z. ; NØRSETT, S. P. ; ZANNA, A.: *Lie-group methods*. In: Acta Numerica 9 (2000), S. 215–365.
- [25] KÖNIGSBERGER, K.: *Analysis 2*. 5. korr. Auflage. Springer, 2009.
- [26] LEE, J. M.: *Introduction to Smooth Manifolds*. 1. Auflage. Springer, 2002.
- [27] LEE, J. M.: *Introduction to Topological Manifolds*. 2. Auflage. Springer, 2010.
- [28] MA, Y. ; SOATTO, S. ; KOSECKA, J. ; SASTRY, S. S.: *An Invitation to 3-D Vision*. 1. Auflage. Springer, 2003.
- [29] MAHONY, R. ; MANTON, J. H.: *The geometry of the newton method on non-compact lie groups*. In: Journal of Global Optimization 23 (2002), S. 309–327.
- [30] MAINTZ, J. ; VIERGEVER, M.: *A survey of medical image registration*. In: Medical Image Analysis 2 (1998), Nr. 1, S. 1–36.
- [31] MALIS, E.: *Improving vision-based control using efficient second-order minimization techniques*. In: Proc. IEEE International Conference on Robotics and Automation (ICRA'04), 2004.
- [32] MODERSITZKI, J.: *Numerical Methods for Image Registration*. Oxford University Press, 2004.
- [33] MODERSITZKI, J.: *FAIR: Flexible Algorithms for Image Registration*. Philadelphia : SIAM, 2009.

-
- [34] MÖLLER, T.: *Thirions Demons-Algorithmus und Newton-Verfahren auf Lie-Gruppen*. Private Conversation, Apr 2011.
- [35] NITHIANANTHAN, S. ; BROCK, K. K. ; DALY, M. J. ; CHAN, H. ; IRISH, J. C. ; SIEWERDSEN, J. H.: *Demons deformable registration for CBCT-guided procedures in the head and neck: Convergence and accuracy*. In: Medical Physics 36 (2009), Nr. 10, S. 4755–4764.
- [36] NOCEDAL, J. ; WRIGHT, S. J.: *Numerical Optimization*. 2. Auflage. Springer, 2006.
- [37] OWREN, B. ; WELFERT, B.: *The newton iteration on lie groups*. In: BIT Numerical Mathematics 40 (2000), S. 121–145.
- [38] PAPENBERG, N.: *Ein allgemeiner Registrierungsansatz mit Anwendung in der navigierten Leberchirurgie*, Institut für Mathematik, Universität zu Lübeck, Diss., 2008.
- [39] PENNEC, X. ; CACHIER, P. ; AYACHE, N.: *Understanding the “demon’s algorithm”: 3D non-rigid registration by gradient descent*. In: Proc. International Conference on Medical Image Computing and Computer Assisted Intervention (MICCAI) Bd. 1679, 1999, S. 597–605.
- [40] ROGELJ, P. ; KOVACIC, S.: *Symmetric image registration*. In: Medical Image Analysis 10 (2006), Nr. 3, S. 484 – 493. – Special Issue on The Second International Workshop on Biomedical Image Registration (WBIR’03).
- [41] SCHWARZ, H. ; KÖCKLER, N.: *Numerische Mathematik*. 8. Auflage. Vieweg+Teubner Verlag, 2011.
- [42] STEFANESCU, R. ; PENNEC, X. ; AYACHE, N.: *Grid powered nonlinear image registration with locally adaptive regularization*. In: Medical Image Analysis 8 (2004), Nr. 3, S. 325–342.
- [43] THIRION, J.-P.: *Fast non-rigid matching of 3D medical images* / INRIA. 1995 (2547). – Forschungsbericht.
- [44] THIRION, J.-P.: *Image matching as a diffusion process: an analogy with Maxwell’s demons*. In: Medical Image Analysis 2 (1998), Nr. 3, S. 243–260.

-
- [45] VARADARAJAN, V.S.: *Lie Groups, Lie Algebras, and Their Representation (Graduate Texts in Mathematics) (v. 102)*. Springer, 1984.
- [46] VEERARAJAN, T.: *Discrete Mathematics*. Tata McGraw-Hill, 2006.
- [47] VERCAUTEREN, T.: *Image registration and mosaicing for dynamic in vivo fibered confocal microscopy*, École Nationale Supérieure des Mines de Paris, Diss., Jan 2008.
- [48] VERCAUTEREN, T.: *A deformable registration toolbox*. MICCAI 2010 Tutorial: Intensity-based Deformable Registration, 2010.
- [49] VERCAUTEREN, T. ; PENNEC, X. ; MALIS, E. ; PERCHANT, A. ; AYACHE, N.: *Insight into efficient image registration techniques and the demons algorithm*. In: Proc. Information Processing in Medical Imaging (IPMI'07) Bd. 4584. Kerkrade, The Netherlands : Springer, Jul 2007 (*Lecture Notes in Computer Science*), S. 495–506.
- [50] VERCAUTEREN, T. ; PENNEC, X. ; PERCHANT, A. ; AYACHE, N.: *Diffeomorphic demons using ITK's finite difference solver hierarchy*. In: Insight Journal – ISC/NA-MIC Workshop on Open Science at MICCAI, 2007. – Source code available online.
- [51] VERCAUTEREN, T. ; PENNEC, X. ; PERCHANT, A. ; AYACHE, N.: *Diffeomorphic demons: efficient non-parametric image registration*. In: NeuroImage 45 (2009), Mar, Nr. 1, S. 61–72.
- [52] VIOLA, P. ; W.M. WELLS, III: *Alignment by maximization of mutual information*. In: IEEE International Conference on Computer Vision (1995), S. 16.
- [53] WANG, H. ; DONG, L. ; O'DANIEL, J. ; MOHAN, R. ; GARDEN, A. S. ; ANG, K. K. ; KUBAN, D. A. ; BONNEN, M. ; CHANG, J. Y. ; CHEUNG, R.: *Validation of an accelerated 'demons' algorithm for deformable image registration in radiation therapy*. In: Physics in Medicine and Biology 50 (2005), Nr. 12.
- [54] YOUNES, L.: *Shapes and Diffeomorphisms (Applied Mathematical Sciences)*. 1. Auflage. Springer, 2010.
- [55] ZANNA, A.: *Image registration by matrix group transformations / MedViz*. 2008. – MedViz Report. – Medical Visualisation Network at University of Bergen, Norway.