



UNIVERSITÄT ZU LÜBECK
INSTITUTE OF MATHEMATICS AND
IMAGE COMPUTING

Masterarbeit im Studiengang Computational Life Science

Registrierung von Flussdaten

Stella Graßhof

Thema ausgegeben von
Prof. Dr. rer. nat. Bernd Fischer
Institute of Mathematics and Image Computing

Arbeit betreut von
Dr. rer. nat. Stefan Heldmann
Fraunhofer MEVIS Project Group Image Registration

Abgabe: 01.03.2012



Eidesstattliche Erklärung

Hiermit versichere ich, die vorliegende Arbeit selbstständig und unter ausschließlicher Verwendung der angegebenen Literatur und Hilfsmittel erstellt zu haben.

Lübeck, den 01. März 2012

Unterschrift

Inhaltsverzeichnis

Notation	1
1 Motivation	2
2 Bildregistrierung	3
2.1 Definitionen	3
2.2 Parametrische Registrierung	5
2.3 Nichtparametrische Registrierung	7
2.4 Vektorwertiger Bilder	8
2.5 Basistransformation	11
2.6 Registrierung von Flussdaten	15
2.7 Übersicht der Zielfunktionen	18
3 Diskretisierung	19
3.1 Parametrische Zielfunktion	19
3.1.1 Gitter	20
3.1.2 Distanzmaß	21
3.1.3 Vektorschreibweise	21
3.1.4 Basistransformation	23
3.1.5 Die diskretisierte parametrische Zielfunktion und ihre Ableitung	25
3.2 Nichtparametrische Zielfunktion	27
3.2.1 Basistransformation	27
3.2.2 Approximation der Jacobimatrix	29
3.2.3 Invertierung der approximierten Jacobimatrix	30
3.2.4 Regularisierer	31
3.2.5 Die diskretisierte nichtparametrische Zielfunktion und ihre Ableitung	36
3.3 Nichtparametrische Zielfunktion mit Basistransformation als Parameter	38
3.3.1 Basistransformation	38
3.3.2 Strafterm	39
3.3.3 Die diskretisierte Zielfunktion und ihre Ableitung	41
3.4 Interpolation	45
4 Numerische Optimierungsverfahren	46
4.1 Bestimmung der Suchrichtung	46
4.2 Schrittweitenbestimmung	48
4.3 Abbruchkriterien	48
5 Ergebnisse	50
5.1 Validierung für parametrische Deformationen	50
5.1.1 Translation	50
5.1.2 Skalierung	51

5.1.3	Scherung	51
5.1.4	Rotation	52
5.1.5	Nicht affin-lineare Deformation	52
5.2	Validierung für nichtparametrische Deformationen	53
5.2.1	Lokale Deformationen	53
5.2.2	Globale Deformationen	55
5.2.3	Verhalten bei Über- und Unterregularisierung	58
5.3	Validierung für dreidimensionale Daten	64
6	Diskussion	66
	Literaturverzeichnis	66

Abbildungsverzeichnis

1.1	Veranschaulichung einer Problemstellung der Registrierung	2
2.1	Veranschaulichung des Unterschieds zwischen skalar- und vektorwertigen Bildern	9
2.2	vektorwertiges Bild (Vektorfeld) und mittels (2.15) deformiertes Bild	11
2.3	Veranschaulichung der Transformation eines Flussbildes.	12
2.4	Veranschaulichung der Basistransformation	13
2.5	Veranschaulichung der funktionierende Basistransformation	15
3.1	Veranschaulichung eines CC-Gitters	20
3.2	Zusammenhang zwischen CC-Gitter und dem Gittervektor \bar{x}	22
5.1	Ergebnis der parametrischen Registrierung einer Translation	50
5.2	Ergebnis der parametrischen Registrierung einer Skalierung	51
5.3	Ergebnis der parametrischen Registrierung einer Scherung	52
5.4	Ergebnis der parametrischen Registrierung einer Rotation	52
5.5	Ergebnis der parametrischen Registrierung einer Gefäßverengung	53
5.6	Ergebnis der nichtparametrischen Registrierung für lokal verengte Blutgefäße.	54
5.7	Ergebnis der nichtparametrischen Registrierung für lokal ausgedehnte Blutgefäße.	55
5.8	Ergebnis der nichtparametrischen Registrierung für ein einseitig gebogenes Blutgefäß.	56
5.9	Ergebnis der nichtparametrischen Registrierung für ein zweifach gebogenes Blutgefäß.	57
5.10	Überregularisierung	58
5.11	Unterregularisierung	59
5.12	Einfluss des Straftermparameters β für $\alpha = \alpha_{\text{opt}}$	61
5.13	Einfluss des Straftermparameters β für $\alpha = \alpha_{\text{opt}}$	61
5.14	Einfluss des Straftermparameters β für $\alpha < \alpha_{\text{opt}}$	62
5.15	Einfluss des Straftermparameters β für $\alpha < \alpha_{\text{opt}}$	62
5.16	Einfluss des Straftermparameters β für $\alpha > \alpha_{\text{opt}}$	63
5.17	Einfluss des Straftermparameters β für $\alpha > \alpha_{\text{opt}}$	63
5.18	Ergebnis der parametrischen Registrierung einer dreidimensionalen Rotation .	64
5.19	Registrierungsergebnis eines dreidimensionalen gebogenen Blutgefäßes	64
5.20	Fehler der dreidimensionalen Registrierung	65

Notationen

Symbole und Kurzschreibweisen

Mengen

\mathbb{N}	Natürliche Zahlen (ohne Null)
\mathbb{R}	Reelle Zahlen
$C^n(D, B)$	Raum, der alle n -mal stetig differenzierbaren Funktionen $f : D \rightarrow B$ enthält
$\mathbf{1}_n \in \mathbb{R}^n$	Einsvektor
$z_n \in \mathbb{R}^n$	Nullvektor
$e_k \in \mathbb{R}^n$	Einheitsvektor: Nullvektor mit genau einer Eins an der Stelle k
$E_n \in \mathbb{R}^{n \times n}, \quad n \in \mathbb{N}$	Einheitsmatrix
$Z_n \in \mathbb{R}^{n \times n}, \quad n \in \mathbb{N}$	Nullmatrix

Operatoren

\odot	Punktweise Multiplikation
\otimes	Kroneckerprodukt
$\partial_k := \partial_{x_k} = \frac{\partial}{\partial x_k}$	Partielle Differentiation nach der k -ten Komponente von x
$\nabla := (\partial_1, \dots, \partial_n)^T$	Nabla-Operator zur Bestimmung des Gradienten einer Funktion $f : \mathbb{R}^n \rightarrow \mathbb{R}^m: \nabla f := (J_f)^T \in \mathbb{R}^{n \times m}$
$J_f \in \mathbb{R}^{m \times n}$	Jacobimatrix von $f : \mathbb{R}^n \rightarrow \mathbb{R}^m$:
	$J_f(a) = \begin{pmatrix} \frac{\partial f_1}{\partial x_1}(a) & \dots & \frac{\partial f_1}{\partial x_n}(a) \\ \vdots & \ddots & \vdots \\ \frac{\partial f_m}{\partial x_1}(a) & \dots & \frac{\partial f_m}{\partial x_n}(a) \end{pmatrix} \in \mathbb{R}^{m \times n}, a \in \mathbb{R}^n$
$\Delta f = \sum_{i=1}^d \partial_{ii} f$	Laplace Operator, $f : \mathbb{R}^d \rightarrow \mathbb{R}$

Normen u.a.

$\ v\ _2 := \sqrt{\sum_{i=1}^n v_i^2}$	Euklidische Norm für Vektoren, $v \in \mathbb{R}^n$
$\ A\ _F = \sqrt{\sum_{i,j} a_{ij} ^2}$	Frobenius Norm für Matrizen, $A \in \mathbb{R}^{n \times n}$
$\text{diag}(v)$	erzeugt eine quadratische Null-Matrix mit dem Vektor v auf der Hauptdiagonalen
$\text{diag}(A)$	extrahiert die Diagonalelemente der Matrix A

1 Motivation

Die medizinische Bildgebung motiviert verschiedenste Problemstellungen der Bildregistrierung. Allgemein besteht das Ziel darin zwei strukturell vergleichbare Bilder in einer Abbildung sinnvoll miteinander zu kombinieren. Dabei soll ein Bild geometrisch so deformiert werden, dass korrespondierende Strukturen in beiden Bildern überlappen. Die zwei Abbildungen erscheinen dann „ähnlich“.

Beispielsweise können prä- und postoperative Röntgenbilder besser verglichen werden, indem eine Aufnahme unter Verwendung einer affin-linearen Abbildung so deformiert wird, dass Lageunterschiede eliminiert werden. In der Abbildung 1.1 sind die Röntgenaufnahmen von zwei Händen dargestellt, die durch eine derartige Funktion ineinander überführt werden können.

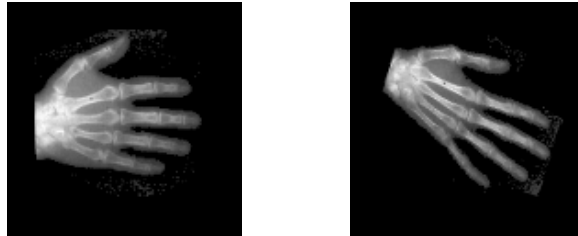


Abbildung 1.1: Veranschaulichung einer Problemstellung der Registrierung, mit Beispielbildern aus [4].

Bei Aufnahmen mit dem Verfahren des Phase Contrast Magnetic Resonance Imaging (pcMRI) entstehenden dreidimensionale Vektorfelder. Es handelt sich dabei um Flussmessungen, die insbesondere den Blutfluss darstellen und so u.a. die Visualisierung der Herzfunktion ermöglichen. Das Ziel ist langfristig diese Messungen zu verwenden, um Gefäßkrankheiten wie z.B. Arteriosklerose und Aneurysmen zu diagnostizieren, indem z.B. Verwirbelungen gefunden werden.

Motiviert durch diese Daten, stellt die vorliegende Arbeit eine Herangehensweise zur Registrierung von zwei- und dreidimensionalen Vektorfeldern vor. Die vorgestellten Ansätze sind vergleichbar mit der variationellen Herangehensweise aus [1], die zur Registrierung von tensorwertigen Bildern beschrieben werden, wie sie beim Diffusion Tensor Imaging (DTI) entstehen. Hier wird ein variationeller Ansatz zur Lösung des Bildregistrierungsproblems vorgestellt, der sich für skalarwertige Bilder bereits bewährt hat. Dafür wird ein Funktional, bestehend aus einem Distanz- und einem Regularisierungsterm, mit numerischen Optimierungsverfahren minimiert. Es zeigt sich, dass bei der Betrachtung von Flussdaten insbesondere das zu Grunde liegende Deformationsmodell und das Distanzmaß angepasst werden müssen. Für eine erfolgreiche Registrierung beinhaltet das vor allem eine Reorientierung der Flussvektoren bei der Deformation.

2 Bildregistrierung

Die medizinische Bildgebung bietet heute eine Vielzahl von Möglichkeiten den menschlichen Körper und die in ihm ablaufenden Prozesse zu visualisieren. Diese bilden oft die Grundlage einer guten Diagnostik. Das Ziel der Bildregistrierung ist es, mehrere Bilder in einer Darstellung zu vereinen, um die verfügbare Information vollständig in einer gemeinsamen Bildfusion zu bündeln.

Das Zusammenführen von zwei Bildern in einer Überlagerung ist nicht trivial, da korrespondierende Strukturen ggf. nicht zur Deckung kommen, selbst wenn die Einschränkung getroffen wird, dass die betrachteten Bilder mit demselben Aufnahmeverfahren erzeugt worden sind. Das eine Bild kann eine translatierte oder rotierte Version des anderen sein. Berücksichtigt man außerdem die zahlreichen nicht zu beeinflussenden Körperfunktionen, so wird es zu vielen weiteren Deformationen kommen, so verursacht beispielsweise die Atmung eine Ausdehnung des Brustkorbes.

Führt man sich das vor Augen, wird klar, dass beim Vergleich von zwei Bildern künstlich Unterschiede durch Lageunterschiede vorhanden sein können, obwohl sie grundsätzlich das Gleiche zeigen. Dadurch werden echte Abweichungen in den verschiedenen Darstellungen nicht erfasst und ein direkter Vergleich von zwei Bildern ist nicht möglich.

Daher soll ein Bild an ein zweites Bild so angeglichen werden, dass künstliche Unterschiede eliminiert werden und der echte Informationsgehalt erhalten bleibt. Dabei erfolgt eine geometrische Deformation, die zueinandergehörige Teile in beiden Bildern in Korrespondenz bringt.

Das Ziel der *Bildregistrierung* ist es, eine solche Deformation φ zu finden, die ein Templatebild T so deformiert, dass korrespondierende Strukturen in T und einem Referenzbild R aufeinander abgebildet werden.

Zur Lösung dieses *Bildregistrierungsproblems* wird im folgenden Kapitel ein variationeller Ansatz vorgestellt, der durch die Minimierung einer Zielfunktion die gesuchte Deformation berechnet. Dazu wird definiert, was ein Bild ist, wie es deformiert werden kann und wie Bilder durch ein Distanzmaß miteinander verglichen werden können. Ergänzend wird ein weiterer Strafterm motiviert.

Zu Beginn wird das Bildregistrierungsproblem für skalarwertige Bilder modelliert und später für die Anwendung an Flussbildern erweitert.

2.1 Definitionen

Die gesamte Problemstellung befasst sich mit der Betrachtung von Bildern. Als solides Fundament ist entsprechend eine problemangepasste Modellierung von Bildern notwendig.

Definition 2.1

Ein d -dimensionales Bild B ist definiert als eine stetige, differenzierbare Abbildung, die jedem Ort $x \in \mathbb{R}^d$ einen Grauwert $B(x)$ zuweist.

$$B : \mathbb{R}^d \rightarrow \mathbb{R}, \quad x \mapsto B(x), \quad B \in C^1(\mathbb{R}^d, \mathbb{R}), \quad d \in \mathbb{N} \quad (2.1)$$

Es wird angenommen, dass Bilder einen endlichen Grauwertbereich besitzen, so dass gilt:

$$\begin{aligned} &\exists c_{min}, c_{max} \in \mathbb{R}, \text{ so dass} \\ &-\infty < c_{min} \leq B(x) \leq c_{max} < \infty, \forall x \in \Omega \end{aligned}$$

Weiterhin gelte, dass Bilder eine endliche Energie besitzen, so dass

$$\int_{\mathbb{R}^d} |B(x)| \, dx < \infty$$

Mit \mathcal{I}_d wird die Menge bezeichnet, die alle d -dimensionalen Bilder umfasst:

$$\mathcal{I}_d := \{B : \mathbb{R}^d \rightarrow \mathbb{R} \mid B \text{ ist ein } d\text{-dimensionales Bild}\}.$$

Der Bildbereich $\Omega \subset \mathbb{R}^d$ eines Bildes B , ist definiert als ein rechteckiges Gebiet (offene, zusammenhängende Menge)

$$\Omega = (\omega_1, \omega_2) \times (\omega_3, \omega_4) \times \dots \subset \mathbb{R}^d,$$

und beschreibt einen endlichen Ausschnitt der Realität auf dem Bilder betrachtet werden. Anschaulich sind zwei Bilder R und T „ähnlich“, wenn sie sich wenig unterscheiden und somit deren „Distanz“ gering ist. Um den Unterschied zwischen Bildern zu quantifizieren wird eine Funktion benötigt, die zwei Bilder auf eine sinnvolle Maßzahl abbildet. Ein *Distanzmaß* \mathcal{D} ist ein Funktional, das zwei Bilder auf einen reellen Wert abbildet.

$$\mathcal{D} : \mathcal{I}_d \times \mathcal{I}_d \rightarrow \mathbb{R} \tag{2.2}$$

Typische Eigenschaften von Distanzmaßen sind:

- Symmetrie $\mathcal{D}(T, R) = \mathcal{D}(R, T)$
- Minimale Distanz $\forall R$ gilt $\mathcal{D}(R, R) = \min_T \mathcal{D}(T, R)$

Die Wahl des Distanzmaßes richtet sich insbesondere danach, welche Modalität die zu vergleichenden Bilder R und T besitzen, wobei die *Modalität* (Bildmodalität) das Aufnahmeverfahren bezeichnet, mit dem ein Bild erzeugt worden ist (z.B. MRT, Röntgen etc.). Insbesondere bei der multimodalen Registrierung, bei der Bilder mit unterschiedlicher Modalität verglichen werden, ist es wichtig, dass die letzte Eigenschaft erfüllt ist. Also, dass ein zu R korrespondierendes T in der jeweils anderen Modalität existiert, so dass gilt: $\mathcal{D}(R, T) = \mathcal{D}(R, R)$. Werden Bilder derselben Modalität verglichen, bietet sich das SSD-Maß an.

Definition 2.2 Das SSD-Maß (sum of squared differences) ist wie folgt definiert

$$SSD(T, R) := \frac{1}{2} \int_{\Omega} (T(x) - R(x))^2 \, dx \tag{2.3}$$

Für die multimodale Registrierung sind andere Distanzmaße notwendig, wie z.B. Cross-Correlation, Mutual Information und Normalized Gradient Fields. Da hier eine monomodale Anwendung vorliegt, werden diese nicht weiter betrachtet. Eine detaillierte Beschreibung ist zu finden in [4] und [5].

Das Ziel der Bildregistrierung ist es, eine Funktion zu finden, die T so verändert, dass das Distanzmaß von R und dem deformierten Templatebild möglichst klein wird. Die gesuchte Funktion φ wird als Deformation bezeichnet. Eine *Deformation* φ ist ein Vektorfeld, das jedem Ort $x \in \mathbb{R}^d$ einen neuen Ort $\varphi(x)$ zuweist:

$$\varphi : \mathbb{R}^d \rightarrow \mathbb{R}^d, \quad x \mapsto \varphi(x), \quad d \in \mathbb{N} \quad (2.4)$$

Ein durch φ deformierte Bild T wird mit $T \circ \varphi = T(\varphi(\cdot))$ bezeichnet. Das Referenzbild R bleibt unverändert.

Im späteren Verlauf wird es sich als sinnvoll erweisen, die Deformation φ getrennt von der Identität $\text{id} : \mathbb{R}^d \rightarrow \mathbb{R}^d, x \mapsto \text{id}(x = x)$ zu betrachten.

Definition 2.3 Die Verrückung u ist definiert als

$$u : \mathbb{R}^d \rightarrow \mathbb{R}^d, \quad u(x) := \varphi(x) - x \quad (2.5)$$

Nachdem grundlegende Termini eingeführt wurden, kann das zu Beginn vorgestellte Bildregistrierungsproblem in einer neuen Form dargestellt und wie folgt konkretisiert werden: Das Ziel besteht darin eine Deformation φ zu finden, die das Distanzmaß \mathcal{D} zwischen dem Referenzbild R und dem deformierten Templatebild ($T \circ \varphi$) minimiert. Es ergibt sich das Optimierungsproblem:

$$\min_{\varphi} \mathcal{D}(T \circ \varphi, R) \quad (2.6)$$

In dieser Form ist die Deformation φ beliebig, so dass es im Allgemeinen unendlich viele Lösungen φ gibt.

In Abhängigkeit des Typs der Deformation wird zwischen parametrischer und nichtparametrischer Registrierung unterschieden.

2.2 Parametrische Registrierung

Allgemein ist eine *parametrische Deformation* φ eine Funktion, die vollständig durch einen reellwertigen Parametervektor θ beschrieben werden kann.

Eine davon ist die *affine* Deformation:

$$\varphi_{\theta}(x) = A_{\theta}x + b_{\theta}, \quad \text{mit } A_{\theta} \in \mathbb{R}^{d \times d}, \det(A_{\theta}) \neq 0, b \in \mathbb{R}^d, \theta \in \mathbb{R}^{d^2+d} \quad (2.7)$$

Mit dieser Funktion können Rotationen, Scherungen, Skalierungen und Translationen beschrieben werden.

Eine *rigide* Deformation ist ein Spezialfall einer affinen Deformation. Rigide Abbildungen sind formerhaltend, also insbesondere längen- und winkelerhaltend und enthalten keine Spiegelungen.

A_{θ} hat dann folgende Eigenschaften:

- $\det(A) = 1$
- $A^T A = E_d$

Für den zweidimensionalen Fall ($d = 2$) ergibt sich als Abbildungsmatrix A_θ , die Rotationsmatrix

$$A_\theta = A(\alpha) = \begin{pmatrix} \cos(\alpha) & -\sin(\alpha) \\ \sin(\alpha) & \cos(\alpha) \end{pmatrix}$$

Eine Erweiterung auf den dreidimensionalen Fall ist unter Verwendung der entsprechenden Rotationsmatrix $\in \mathbb{R}^{3 \times 3}$ möglich.

Als Spezialfall für parametrische Deformationen werden in dieser Arbeit affine Abbildungen betrachtet. Diese Funktionen werden im Folgenden mit φ_θ bezeichnet.

Falls der Suchraum auf parametrische Deformationen wie in (2.7) beschränkt wird, kann die Zielfunktion aus (2.6) unter Verwendung des zugehörigen Parametervektors $\theta \in \mathbb{R}^{d^2+d}$ neu formuliert werden.

Das Optimierungsproblem aus (2.6) lautet für den parametrischen Fall somit

$$\min_{\theta} \mathcal{D}(T \circ \varphi_\theta, R) \quad (2.8)$$

Exemplarisch wird diese Zielfunktion für den zweidimensionalen Fall mit $T, R \in \mathcal{I}_2$ und $\Omega = (\omega_1, \omega_2) \times (\omega_3, \omega_4)$ vollständig beschrieben.

Mit $x \in \mathbb{R}^2$, $A_\theta := \begin{pmatrix} \theta_1 & \theta_2 \\ \theta_4 & \theta_5 \end{pmatrix} \in \mathbb{R}^{2 \times 2}$, $b_\theta := \begin{pmatrix} \theta_3 \\ \theta_6 \end{pmatrix} \in \mathbb{R}^2$, folgt für die Deformation

$$\varphi(x) := \varphi(x, \theta) = A_\theta x + b_\theta = \begin{pmatrix} \theta_1 x_1 + \theta_2 x_2 + \theta_3 \\ \theta_4 x_1 + \theta_5 x_2 + \theta_6 \end{pmatrix}$$

das deformierte Template kann dann angegeben werden als:

$$T(\varphi(x)) = T \begin{pmatrix} \varphi_1(x_1, x_2) \\ \varphi_2(x_1, x_2) \end{pmatrix} = T \begin{pmatrix} \theta_1 x_1 + \theta_2 x_2 + \theta_3 \\ \theta_4 x_1 + \theta_5 x_2 + \theta_6 \end{pmatrix}$$

Die Zielfunktion lautet entsprechend

$$\begin{aligned} \mathcal{D}(T \circ \varphi_\theta, R) &= \frac{1}{2} \int_{\Omega} (T(\varphi(x)) - R(x))^2 dx \\ &= \frac{1}{2} \int_{\omega_3}^{\omega_4} \int_{\omega_1}^{\omega_2} \left(T \begin{pmatrix} \theta_1 x_1 + \theta_2 x_2 + \theta_3 \\ \theta_4 x_1 + \theta_5 x_2 + \theta_6 \end{pmatrix} - R \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} \right)^2 dx_1 dx_2 \quad (2.9) \end{aligned}$$

Die Wahl einer affinen Abbildung als Deformation ist plausibel, da anschauliche Deformationen wie Translationen ermöglicht werden, so dass z.B. ermöglicht wird die veränderte Lage eines Patienten zu registrieren. Die Festlegung auf parametrische Deformationen schränkt den Suchraum stark ein. Im Folgenden sollen auch solche Deformationen berücksichtigt werden, die nicht vollständig durch einen Parametervektor beschrieben werden können. Diese Erweiterung ist insofern vielversprechend, da angenommen werden kann, dass weitere Deformationen gefunden werden können, die beispielsweise die Dehnung der Lunge bei der Atmung angemessen registrieren können.

2.3 Nichtparametrische Registrierung

Bei der Erweiterung von parametrischen auf *nichtparametrische* Deformationen, sollen auch solche Funktionen berücksichtigt werden, die nicht in der parametrischen Form darstellbar sind.

Ohne Einschränkungen hat das vorgestellte Optimierungsproblem im Allgemeinen unendlich viele Lösungen, die zudem unerwünschte Eigenschaften besitzen können. Eine Lösung φ könnte beispielsweise nicht stetig oder differenzierbar sein. Angestrebt wird eine Umformulierung des parametrischen Optimierungsproblems, so dass eine eindeutige, stetige Lösung existiert. Um sich diesem Ziel zu nähern, wird der Zielfunktion ein Strafterm hinzugefügt, der dafür sorgen soll, dass die Lösung φ die gewünschten Kriterien erfüllt. Dieser Term wird als *Regularisierer* bezeichnet.

Sei $V \subset \{\varphi \mid \varphi : \mathbb{R}^d \rightarrow \mathbb{R}^d\}$ ein geeigneter Funktionenraum. Ein *Regularisierer* S ist ein Funktional, das einer Deformation φ eine Zahl zuordnet.

$$S : V \rightarrow \mathbb{R}, \quad S : \varphi \mapsto \mathbb{R} \quad (2.10)$$

Der Regularisierer wird auch als Glätter bezeichnet. Typischerweise wird die Verrückung bestraft, statt der ursprünglichen Deformation φ .

Zwei übliche Regularisierer sind:

$$\text{Diffusiver Glätter} \quad \mathcal{S}_{\text{diff}}(u) = \frac{1}{2} \int_{\Omega} \sum_{k=1}^2 \|\nabla u_k\|^2 dx$$

Der diffusive Regularisierer bestraft große Werte der ersten partiellen Ableitungen. Somit werden glatte Deformationen φ begünstigt und Oszillationen und Sprünge in φ und der Verrückung u bestraft. Handelt es sich bei φ oder u um eine Translation, folgt $\mathcal{S}_{\text{diff}}(u) = 0$, da dann die ersten partiellen Ableitungen von u verschwinden.

$$\text{Curvature Glätter} \quad \mathcal{S}_{\text{curv}}(u) = \frac{1}{2} \int_{\Omega} \sum_{k=1}^d (\Delta u_k)^2 dx$$

Der curvature Regularisierer bestraft zweite Ableitungen um Krümmungen der Komponenten der Verrückung u zu minimieren und fordert so einen höheren Grad an Glattheit. Handelt es sich bei φ oder u um eine affine Abbildung, wie sie für die parametrische Registrierung vorausgesetzt wird, folgt $\mathcal{S}_{\text{curv}}(u) = 0$, da in diesem Fall die zweiten partiellen Ableitungen von u verschwinden.

Die hier vorgestellten Glätter sind gebräuchliche Beispiele aus einer Vielzahl von weiteren Varianten. Sie wurden für diese Arbeit ausgewählt, da sie einfach zu implementieren sind und sich bereits mehrfach in verschiedenen Anwendungen bewährt haben. Neben diesen gibt es beispielsweise noch den *elastischen Glätter*, der invariant gegenüber rigidem Transformationen ist. Wie bereits beim diffusiven Glätter sind nur erste Ableitungen Bestandteil des elastischen Regularisierers. Die drei erwähnten Regularisierer werden beschrieben in [4, 5].

Mit einem fest gewählten Regularisierer \mathcal{S} und dem zugehörigen Regularisierungsparameter α lautet die neue Zielfunktion für den nichtparametrischen Fall, für gegebene Bilder $R, T \in \mathcal{I}_d$ und mit fest gewähltem Distanzmaß \mathcal{D} somit:

$$\mathcal{D}(T \circ \varphi, R) + \alpha \mathcal{S}(u), \quad \alpha \in \mathbb{R}^+$$

Das nichtparametrische Optimierungsproblem ergibt sich wie folgt

$$\min_{\varphi} \mathcal{D}(T \circ \varphi, R) + \alpha \mathcal{S}(u) \quad (2.11)$$

Im Vergleich zum parametrischen Fall (2.8), hat sich der Suchraum in (2.11) verändert. Statt eines Parametervektors, wird jetzt eine Funktion gesucht.

Die bisher vorgestellten Zielfunktionen basieren auf der Annahme, dass die vorliegenden Bilder skalarwertig sind, wie in Definition (2.1) angegeben. Typische Beispiele für skalarwertige Bilder entstehen bei MRI- oder CT-Aufnahmen. In diesem Fall kann das Bildregistrierungsproblem unter Verwendung der vorgestellten Komponenten gelöst werden.

Daher liegt es nahe diese Ansätze und Bestandteile zu verwenden um Erweiterungen für Flussbilder zu erhalten, welche die Grundlage für diese Arbeit bilden.

Zu Beginn des nächsten Abschnitts wird zunächst definiert, was ein vektorwertiges Bild ist, um darauf aufbauende Funktionen entsprechend adaptieren zu können. Folglich ist es notwendig die Definition eines Distanzmaßes anzupassen. Tatsächlich müssen aber nicht alle Terme der Zielfunktion erweitert werden, da der Definitionsbereich von skalar- und vektorwertigen Bildern derselbe ist. Entsprechend bleiben die Definitionen der darauf aufbauenden Funktionen unverändert, dazu gehören Deformation, Verrückung und Regularisierer.

2.4 Vektorwertiger Bilder

Zur Veranschaulichung der vorliegenden Daten, ist der Unterschied zwischen skalar- und vektorwertigen Bildern in der Abbildung 2.1 dargestellt, wobei die linke Spalte jeweils ein skalarwertiges und die rechte ein vektorwertiges Bild zeigt.

Definition 2.4 Ein d -dimensionales, vektorwertiges Bild B ist definiert als eine stetige, differenzierbare Abbildung, die jedem Ort $x \in \mathbb{R}^d$ einen Vektor $B(x) \in \mathbb{R}^n$ zuweist

$$B : \mathbb{R}^d \rightarrow \mathbb{R}^n, \quad x \mapsto B(x), \quad B \in C^1(\mathbb{R}^d, \mathbb{R}^n), \quad d, n \in \mathbb{N} \quad (2.12)$$

Die n Komponenten des Bildes werden im Folgenden auch als Kanäle bezeichnet.

Das Bild B kann durch seine n Kanäle als Vektor mit skalarwertigen Bildern wie folgt dargestellt werden:

$$B = \begin{pmatrix} B_1 \\ B_2 \\ \vdots \\ B_n \end{pmatrix} \in \mathbb{R}^n, \quad B_k \in \mathcal{I}_d$$

Entsprechend ist $\mathcal{I}_{d,n}$ die Menge, die d -dimensionale, vektorwertige Bilder mit n Komponenten enthält. Diese lautet

$$\mathcal{I}_{d,n} := \left\{ B : \mathbb{R}^d \rightarrow \mathbb{R}^n \mid B = (B_1, \dots, B_n)^T, \quad B_k \in \mathcal{I}_d \right\}.$$

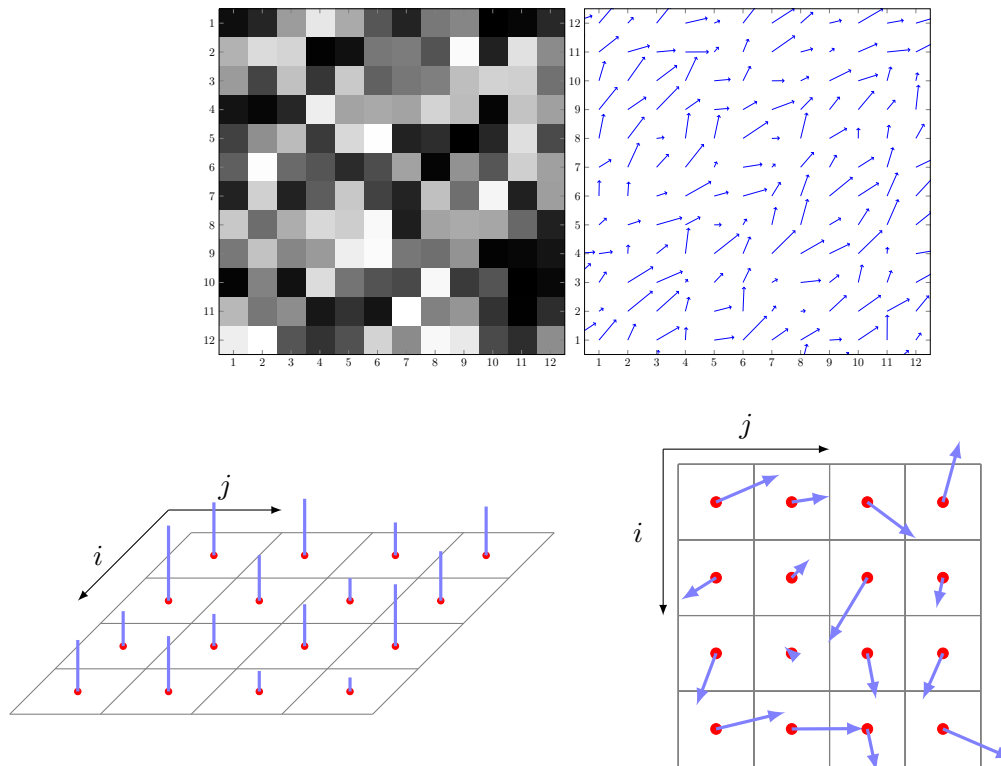


Abbildung 2.1: Veranschaulichung des Unterschieds zwischen skalar- und vektorwertigen Bildern

Bei dieser Definition bleibt der zugehörige Bildbereich Ω grundlegend erhalten, so dass eine Anpassung der Definitionen von Deformation (2.4), Verrückung (2.5), und des Regularisierers (2.10) nicht erforderlich ist, wohingegen der Begriff des Distanzmaßes aus (2.2) noch erweitert werden muss.

Intuitiv kann (2.2) für vektorwertige Bilder wie folgt formuliert werden:

Ein *Distanzmaß* \mathcal{D} ist ein Funktional, das zwei d -dimensionale Bilder mit n Komponenten auf einen reellen Wert abbildet.

$$\mathcal{D} : \mathcal{I}_{d,n} \times \mathcal{I}_{d,n} \rightarrow \mathbb{R} \quad (2.13)$$

Bisher ist aber noch nicht geklärt, wie Unterschiede zwischen vektorwertigen Bildern sinnvoll in einer Zahl quantifiziert werden können. Man kann sich ein vektorwertiges Bild, als eine Menge von Vektoren veranschaulichen, die paarweise verglichen werden können. Beispielsweise könnte man die absoluten Differenzen der einzelnen Bildvektorelemente von R und T betrachten oder die Drehwinkel und Magnituden jedes Vektors berechnen, da so jeweils wieder ein skalarer Wert für jeden Vektor entstehen würde. Die Funktion f , die jeweils den Drehwinkel oder die Magnitude für einen Bildvektor $B(x)$ berechnet, kann dann aufgefasst werden als ein skalarwertiges Bild, das einen Datenpunkt $x \in \mathbb{R}^d$ auf einen reellen Wert $f(B(x))$ abbildet. Damit wäre die Zielfunktion für skalarwertige Bilder wieder verwendbar, indem die entsprechende Funktion f integriert wird. Es würden sich sogar mehrere Distanzfunktionen aus dem SSD-Maß herleiten lassen, je nachdem welche Funktion f gewählt wird, so dass auch Kombinationen mehrerer Distanzmaße mit unterschiedlichen Gewichtungen betrachtet

werden können.

Bei diesen Ansätzen wird die in den Daten enthaltene Information allerdings nicht berücksichtigt. Da es sich dabei um Flussdaten handelt, wird einem Ort ein Flussvektor gleicher Dimension zugewiesen, so dass $n = d$ gilt. Ein solcher Flussvektor beschreibt für ein Teilchen, das sich an dem betrachteten Ort befindet, in welcher Richtung der zukünftige Ort liegt. Die Bewegung eines Teilchens wird durch die Zuweisung einer Richtung und der durch die Vektorlänge kodierten Geschwindigkeit beschrieben.

Dementsprechend ist es sinnvoll anzunehmen, dass zwei Flussbilder identisch sind, wenn für alle Orte gilt, dass derselbe Ort zur Abbildung auf denselben Vektor führt. Analog unterscheiden sie sich umso mehr, je größer der Unterschied zwischen deren Richtung und Länge ist. Diese Interpretation sollte in das neue Distanzmaß einfließen.

Es soll erreicht werden, dass die Länge Flussvektoren der Bilder an jeweils einem Ort übereinstimmt und, dass sie in dieselbe Richtung weisen, so dass die Distanz zwischen den Bildvektoren eines Ortes gering wird. Allgemein lässt sich der Unterschied von zwei Vektoren durch die Norm ihrer Differenz quantifizieren. Infolgedessen ist es zweckmäßig die Norm der Differenz von zwei Bildvektoren für alle Orte des Definitionsbereichs zu betrachten und diese als Maß zu verwenden.

Entsprechend kann das bisher verwendete SSD-Maß für skalarwertige Bilder aus (2.3) für Bilder $T, R \in \mathcal{I}_{d,d}$ angepasst werden, indem über die Elemente des Differenzvektors der Bildvektoren summiert wird, so dass sich ergibt:

$$\begin{aligned} \text{SSD}_d(T, R) &= \frac{1}{2} \int_{\Omega} \|T(x) - R(x)\|_2^2 dx & (2.14) \\ &= \frac{1}{2} \int_{\Omega} \sum_{k=1}^d (T_k(x) - R_k(x))^2 dx \\ &= \sum_{k=1}^d \text{SSD}(T_k, R_k) \end{aligned}$$

Im weiteren Verlauf bezeichnet \mathcal{D} das SSD-Maß für vektorwertige Bilder.

Zur Lösung des bekannten Optimierungsproblems erhält das Distanzmaß das deformierte Template als Argument. Für skalare $T \in \mathcal{I}_2$ ist dies bereits bekannt und lautet:

$$T \circ \varphi = T(\varphi_1(x_1, x_2), \varphi_2(x_1, x_2))$$

Entsprechend gilt für vektorwertige Bilder $T \in \mathcal{I}_{2,2}$

$$T \circ \varphi(x_1, x_2) = \begin{pmatrix} T_1(\varphi_1(x_1, x_2), \varphi_2(x_1, x_2)) \\ T_2(\varphi_1(x_1, x_2), \varphi_2(x_1, x_2)) \end{pmatrix} \quad (2.15)$$

Von dieser Erweiterung wird erwartet, dass eine Deformation des Bildbereichs zu einer passenden Deformation der Bildvektoren führt. Die Vektoren sollen entsprechend der Veränderung des Bildbereichs translatiert, skaliert oder rotiert werden. Alle Deformationen des Flussbildes können durch eine entsprechende Anpassung der Richtung und Länge erreicht werden.

Zu Beginn wird die Rotation eines Flussbildes betrachtet. Das mit dem Ansatz aus (2.15) deformierte Bild wird in Abbildung 2.2 dargestellt.

Aus dieser Illustration geht hervor, dass das Quadrat als Bezugssystem erfolgreich gedreht wird, während die enthaltenen Vektoren in ihrer ursprünglichen Richtung verharren. Dies ist vergleichbar mit dem Verhalten einer Kompassnadel, die immer nach Norden zeigt.

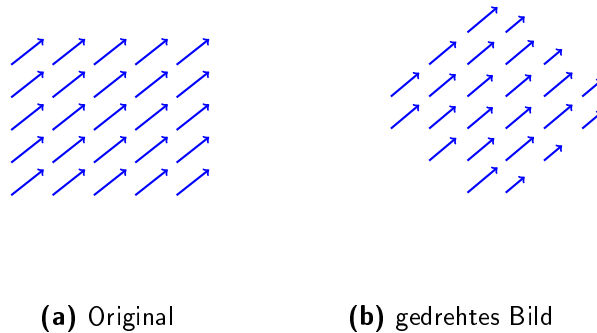


Abbildung 2.2: vektorwertiges Bild (Vektorfeld) und mittels (2.15) deformiertes Bild

Da dies nicht dem gewünschten Verhalten entspricht, wird eine Basistransformation benötigt. Im Folgenden wird eine Abbildung hergeleitet, welche die notwendige Reorientierung der Vektoren herbeiführt. Diese wird anschließend in die neue Zielfunktion integriert.

2.5 Basistransformation

Was erwartet man bei der Deformation eines vektorwertigen Bildes? Bei einer Translation sollen die Vektoren ebenfalls verschoben werden, bei einer Skalierung wird erwartet, dass sich die Vektoren entsprechend verkürzen oder verlängern und bei einer Rotation wird die Orientierung der Vektoren verändert.

Jede dieser Vektormodifikationen basiert auf einer Veränderung des Flussvektors.

Ein Flussbild bestehend aus Vektoren $v(x) \in \mathbb{R}^d$, enthält die Information darüber, in welche Richtung ein Teilchen fließt, wenn es sich an dem Ort x befindet. Zusätzlich kodiert die Länge des Vektors die Geschwindigkeit. Wird jeder Ort x als zeitabhängig betrachtet $x = x(t)$, dann beschreibt $v(x(t))$ die Änderung des Aufenthaltsortes eines Teilchens für den Zeitpunkt t , wenn das Teilchen am Ort $x(0)$ gestartet ist. Der zu $x(t)$ gehörige Flussvektor $v(x)$ beschreibt die Änderung des Ortes x in Abhängigkeit von der Zeit wie folgt

$$v(x(t)) = \frac{d}{dt}x(t)$$

Wird ψ als eine Ortstransformation aufgefasst, so ergibt sich der neue Ort nach der Transformation als

$$y(t) = \psi(x(t))$$

Der auf dem neuen Ortssystem y definierte Fluss wird mit w bezeichnet und kann wie folgt dargestellt werden

$$w(y(t)) = \frac{d}{dt}y(t) = \frac{d}{dt}\psi(x(t))$$

Der Zusammenhang zwischen v und w ist in Abbildung 2.3 dargestellt.

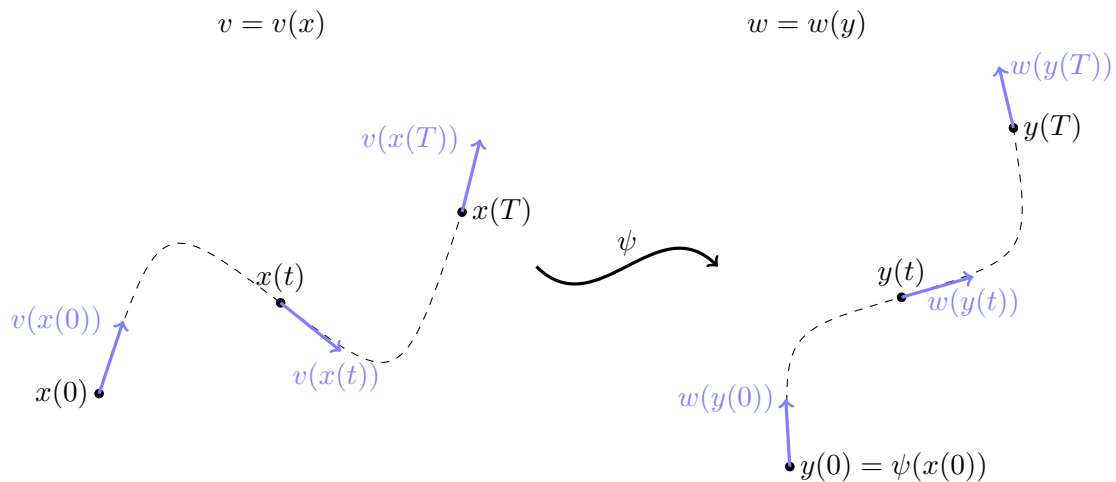


Abbildung 2.3: Veranschaulichung der Transformation eines Flussbildes.

Unter Verwendung der zugehörigen Jacobimatrix von ψ ergibt sich

$$w(y(t)) = \frac{d}{dt} \psi(x(t)) = J_{\psi}(x(t)) \cdot \underbrace{\frac{d}{dt} x(t)}_{=v(x)}$$

Der Zusammenhang zwischen dem neuen Flussbild w auf den transformierten Orten und dem alten Flussbild v kann demzufolge wie folgt beschrieben werden

$$w(y(t)) = w(\psi(x(t))) = J_{\psi}(x(t)) \cdot v(x(t))$$

Durch Vernachlässigung des Zeitindex t kann dies dargestellt werden als

$$w(\psi(x)) = J_{\psi}(x) \cdot v(x)$$

Unter der Annahme, dass ψ invertierbar ist, folgt mit der Substitution $x = \psi^{-1}(y)$

$$w(y) = J_{\psi}(\psi^{-1}(y)) \cdot v(\psi^{-1}(y))$$

Bisher enthält w sowohl ψ , als auch die zugehörige Umkehrabbildung ψ^{-1} . Die Identität von y kann beschrieben werden als

$$y = \psi(\psi^{-1}(y))$$

Durch Bildung der Ableitung folgt:

$$\begin{aligned} E_d &= J_{\psi}(\psi^{-1}(y)) \cdot J_{\psi^{-1}}(y) \\ \Leftrightarrow J_{\psi}(\psi^{-1}(y)) &= (J_{\psi^{-1}}(y))^{-1} \end{aligned}$$

Durch Einsetzen in w folgt

$$w(y) = (J_{\psi^{-1}}(y))^{-1} v(\psi^{-1}(y))$$

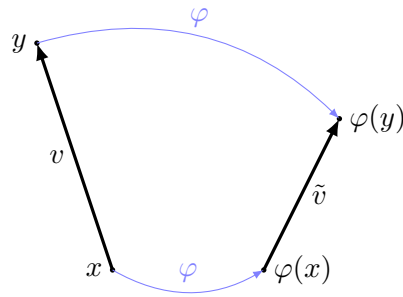


Abbildung 2.4: Veranschaulichung der Motivation der Basistransformation anhand von Verschiebungen des Start- und Endpunktes eines Vektors.

Wird ψ als die Deformation nach Lagrange-Koordinaten aufgefasst, dann ergibt sich die Transformation nach Euler-Koordinaten unter Verwendung der Substitution $\varphi = \psi^{-1}$ wie folgt:

$$w(y) = (J_\varphi(y))^{-1} \cdot v(\varphi(y)) \quad (2.16)$$

Die Matrix $(J_\varphi)^{-1}$ wird im Folgenden mit $B(\varphi)$ als *Basistransformation* bezeichnet.

$$B(\varphi)(x) := (J_\varphi(x))^{-1}$$

Eine alternative Herleitung dieser Basistransformation ergibt sich aus einer diskreten Sicht.

Alternative Herleitung der Basistransformation

Die Deformation eines Flussbildes, kann aufgefasst werden als eine Vektormodifikation, die auf der Veränderung des Start- und Endpunktes jedes Vektors basiert. Jeder neue Vektor im deformierten Bild, entsteht durch die Verschiebung des ursprünglichen Start- und Endpunktes. Das Ziel der Basistransformation ist es, das alte Bezugssystem wiederherzustellen.

Angenommen ein zweikomponentiges, vektorwertiges Bild $T = (T_1, T_2)$ besteht nur aus einem Vektor $v = y - x \in \mathbb{R}^2$ mit Startpunkt x und Endpunkt y der mittels der Deformation φ auf \tilde{v} transformiert wird, dann kann man sich diesen Sachverhalt wie in Abbildung 2.4 dargestellt veranschaulichen.

Für Vektoren beliebiger Dimension d , ergibt sich sofort:

$$\begin{aligned} \text{ursprünglicher Vektor: } v &= y - x && \in \mathbb{R}^d \\ \text{transformierter Vektor: } \tilde{v} &= \varphi(y) - \varphi(x) && \in \mathbb{R}^d \end{aligned}$$

Das Ziel besteht nun darin, eine Basistransformation zu erstellen, welche die ursprüngliche Orientierung durch die neue darstellt.

Mit der Jacobimatrix $J_\varphi \in \mathbb{R}^{d \times d}$ von $\varphi : \mathbb{R}^d \rightarrow \mathbb{R}^d$ lautet die Taylorentwicklung von φ am Punkt y mit Entwicklungspunkt x

$$\varphi(y) = \varphi(x) + J_\varphi(x)(y - x) + \mathcal{O}(h^2), \quad h = \|y - x\|$$

$$\begin{aligned}
 \Leftrightarrow & \quad \underbrace{\varphi(y) - \varphi(x)}_{=\tilde{v}} = J_\varphi(x) \underbrace{(y - x)}_{=v} + \mathcal{O}(h^2) \\
 \Leftrightarrow & \quad \tilde{v} = J_\varphi(x)v + \mathcal{O}(h^2) \\
 \Rightarrow & \quad \tilde{v} \approx J_\varphi(x)v
 \end{aligned}$$

unter der Voraussetzung, dass die Matrix J_φ invertierbar ist, ergibt sich

$$v \approx (J_\varphi(x))^{-1} \tilde{v} \quad (2.17)$$

Auch hier ergibt sich die Matrix $(J_\varphi)^{-1}$ als Basistransformationsmatrix:

$$B(\varphi)(x) = (J_\varphi(x))^{-1} \quad (2.18)$$

Da die Basistransformation sowohl von φ , als auch von x abhängig ist, muss für jeden Datenpunkt jeweils eine Basistransformationsmatrix in Abhängigkeit von der Deformation berechnet werden. Da die Invertierung der Jacobimatrix von φ dazu notwendig ist, wird angenommen, dass J_φ immer invertierbar ist. Je nach Form der Deformation φ ergeben sich verschiedene Basistransformationen $B(\varphi)$. Im Folgenden werden diese exemplarisch für zweidimensionale, vektorwertige Bilder $T \in \mathcal{I}_{2,2}$ mit zwei Komponenten $T_k \in \mathcal{I}_2$, $T_k : \mathbb{R}^2 \rightarrow \mathbb{R}$ vorgestellt. Die Darstellung für höhere Dimensionen ist analog.

Für *nichtparametrische* Deformationen φ , lautet die zugehörige Basistransformation

$$\begin{aligned}
 B(\varphi)(x) &= J_\varphi^{-1}(x) = \begin{pmatrix} \partial_1 \varphi_1(x) & \partial_2 \varphi_1(x) \\ \partial_1 \varphi_2(x) & \partial_2 \varphi_2(x) \end{pmatrix}^{-1} \in \mathbb{R}^{2 \times 2} \\
 &= (\partial_1 \varphi_1 \cdot \partial_2 \varphi_2 - \partial_2 \varphi_1 \partial_1 \varphi_2)^{-1} \begin{pmatrix} \partial_2 \varphi_2 & -\partial_2 \varphi_1 \\ -\partial_1 \varphi_2 & \partial_1 \varphi_1 \end{pmatrix} \quad (2.19)
 \end{aligned}$$

Für *parametrische* Deformationen der Form $\varphi_\theta(x) = A_\theta x + b_\theta$ (wie in (2.7)) lautet die zugehörige Jacobimatrix $J_{\varphi_\theta} = A_\theta$, so dass die zugehörige Basistransformation sofort angegeben werden kann als:

$$\begin{aligned}
 B(\varphi)(x) &= J_\varphi^{-1}(x) = A_\theta^{-1} = \begin{pmatrix} \theta_1 & \theta_2 \\ \theta_4 & \theta_5 \end{pmatrix}^{-1} \\
 &= \frac{1}{\theta_1 \theta_5 - \theta_2 \theta_4} \begin{pmatrix} \theta_5 & -\theta_2 \\ -\theta_4 & \theta_1 \end{pmatrix} \quad (2.20)
 \end{aligned}$$

Da diese Matrix konstant in x ist, folgt, dass die Basistransformation für affine Deformationen identisch für alle Datenpunkte x ist:

$$\varphi(x) = \varphi_\theta(x) \quad \Rightarrow \quad B(\varphi)(x) = A_\theta^{-1}, \quad \forall x$$

Dies ist eine starke Vereinfachung zum nichtparametrischen Fall, da die Jacobimatrix nicht für jeden Datenpunkt einzeln ausgewertet werden muss.

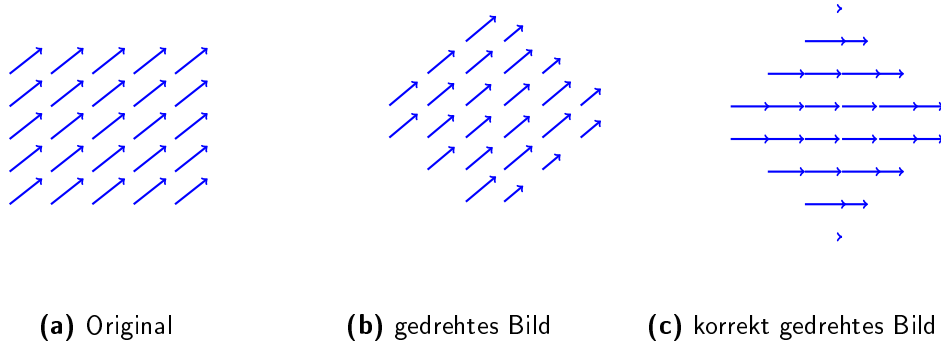


Abbildung 2.5: Veranschaulichung der funktionierende Basistransformation

Wird eine der Basistransformationen angewandt auf ein zweidimensionales, deformiertes Bild $T(\varphi) \in \mathcal{I}_{2,2}$, lautet das transformierte Bild entsprechend wie folgt:

$$B(\varphi) \cdot (T \circ \varphi) = \begin{pmatrix} B_{11}(\varphi) \cdot T_1(\varphi) + B_{12}(\varphi) \cdot T_2(\varphi) \\ B_{21}(\varphi) \cdot T_1(\varphi) + B_{22}(\varphi) \cdot T_2(\varphi) \end{pmatrix}, \quad B_{ij} : \mathbb{R}^2 \rightarrow \mathbb{R}$$

Die Abbildung 2.5 illustriert die Rotation eines quadratischen Vektorfeldes unter Verwendung des alten Ansatzes aus (2.15) ohne Basistransformation in der zweiten Spalte und mit Basistransformation aus (2.18) in der letzten Spalte. Diese Darstellung belegt, dass die Vektoren des Vektorfeldes mit der Erweiterung sinnvoll transformiert werden. Im Folgenden werden die Zielfunktionen, die bereits für skalarwertige Bilder definiert wurden, auf vektorwertige erweitert, indem die in diesem Abschnitt hergeleitete Basistransformation sinnvoll ergänzt wird.

2.6 Registrierung von Flussdaten

Die Zielfunktionen aus (2.8) und (2.11) für skalarwertige Bilder können mittels der im letzten Abschnitt hergeleiteten Basistransformation für vektorwertige Bilder $T, R \in \mathcal{I}_{d,d}$ entsprechend erweitert werden.

Die Zielfunktion $\mathcal{J}_p : \mathbb{R}^{(d^2+d)} \rightarrow \mathbb{R}$ für die parametrische Registrierung von Flussdaten kann wie folgt dargestellt werden:

$$\begin{aligned} \mathcal{J}_p(\theta) &= \mathcal{D} (A_\theta^{-1} \cdot (T \circ \varphi_\theta), R) \\ &= \frac{1}{2} \int_{\Omega} \|A_\theta^{-1} \cdot (T \circ \varphi_\theta(x)) - R(x)\|_2^2 dx \end{aligned} \tag{2.21}$$

mit:

$$T, R \in \mathcal{I}_{d,d}, \quad \varphi_\theta : \mathbb{R}^d \rightarrow \mathbb{R}^d, \quad A_\theta^{-1} \in \mathbb{R}^{d \times d}$$

Hier ist die Basistransformation für alle Punkte $x \in \mathbb{R}^d$ identisch, so dass sie in einer Matrix A_θ^{-1} zusammengefasst werden kann.

Eine Darstellung durch Parameter θ ist bei der nichtparametrischen Deformation nicht möglich. Durch wenige Veränderungen an der parametrischen Darstellung (2.21), kann die Zielfunktion für die nichtparametrische Registrierung $J_{np} : V \rightarrow \mathbb{R}$ wie folgt beschrieben werden:

$$\begin{aligned} \mathcal{J}_{np}(\varphi) &= \mathcal{D}\left((J_\varphi)^{-1}(T \circ \varphi), R\right) + \alpha \mathcal{S}(u) \\ &= \frac{1}{2} \int_{\Omega} \left\| (J_\varphi)^{-1}(x) \cdot (T \circ \varphi(x)) - R(x) \right\|_2^2 dx + \alpha \mathcal{S}(u) \end{aligned} \quad (2.22)$$

mit einem geeigneten Funktionenraum $V \subset \{\varphi \mid \varphi : \mathbb{R}^d \rightarrow \mathbb{R}^d\}$

$$\alpha \in \mathbb{R}^+, \quad T, R \in \mathcal{I}_{d,d}, \quad \varphi : \mathbb{R}^d \rightarrow \mathbb{R}^d, \quad (J_\varphi)^{-1} : \mathbb{R}^d \rightarrow \mathbb{R}^{d \times d}$$

Unter Verwendung dieser Zielfunktion, muss die Inverse der Jacobimatrix von φ in jedem Iterationsschritt bestimmt und zudem an jedem Datenpunkt ausgewertet werden. Bisher ist nicht garantiert, dass J_φ invertierbar ist, obwohl das die Bedingung für eine erfolgreiche Reorientierung ist. Um diesen Problemen zu begegnen, kann die Basistransformation als zusätzlicher Optimierungsparameter aufgefasst werden. Die Zielfunktion hängt dann von der Deformation φ und der Basistransformationsmatrix B gleichermaßen ab, so dass eine erste Erweiterung der nichtparametrischen Zielfunktion lautet:

$$\mathcal{D}(B \cdot (T \circ \varphi), R) + \alpha \mathcal{S}(u), \quad B : \mathbb{R}^d \rightarrow \mathbb{R}^{d \times d}$$

Allerdings ist diese Darstellung unzureichend, da die Basistransformation $B(x)$ in dieser Form ohne weitere Einschränkungen beliebig sein kann. Es ist insbesondere möglich, dass sie stark von $(J_\varphi)^{-1}$ abweicht, was zur Folge hat, dass die Reorientierung der Vektoren nicht korrekt ausgeführt wird. Daher wird ein weiterer Strafterm \mathcal{P} eingeführt, der dafür sorgen soll, dass sich $B(x)$ und $(J_\varphi)^{-1}(x)$ geeignet ähnlich sind.

Gesucht ist eine Funktion \mathcal{P} , die einer Basistransformation B einen reellen, positiven Wert zuweist, so dass solche Basistransformationsmatrizen B bestraft werden, die sich von der inversen Jacobimatrix der Deformation φ hinreichend unterscheiden. Entsprechend sollen solche B einen möglichst großen Wert erhalten, deren Abweichung von $(J_\varphi)^{-1}$ besonders groß sind. Das Ziel besteht entsprechend darin, die Differenz zwischen $B(x)$ und $(J_\varphi)^{-1}(x)$ zu verringern. Dazu müssen Matrizen miteinander verglichen werden und deren Unterschied auf einen skalaren Wert abgebildet werden. Wie bereits bei dem SSD-Distanzmaß und den vorgestellten Regularisierern, wird dazu eine Norm verwendet.

Die erste Näherung des neuen Strafterms \mathcal{P} bildet die Forderung ab, die Differenz zwischen B und $(J_\varphi)^{-1}$ zu verringern

$$\left\| B - (J_\varphi)^{-1} \right\|_F^2$$

Der Vergleich der beiden Matrizen ist auch möglich ohne die Inverse der Matrix J_φ explizit zu berechnen: Wird B als Representant von $(J_\varphi)^{-1}$ aufgefasst, so sollte die Matrixmultiplikation dieser beiden Matrizen die Einheitsmatrix ergeben oder dieser zumindest möglichst ähnlich sein, was zu der Forderung führt, dass die Differenz der beiden Terme möglichst gering sein sollte.

$$\|B \cdot J_\varphi - E_d\|_F^2$$

Die Basistransformation B und die Jacobimatrix J_φ werden beide weiterhin an jedem Ort x ausgewertet. Um die Anforderungen an jede Basistransformation zu formulieren, muss die vorgestellte Norm für jeden Ort des Bildbereichs Ω ausgewertet werden, so dass sich der endgültige Strafterm wie folgt formulieren lässt:

$$\mathcal{P}(\varphi, B) = \int_{\Omega} \|B(x) \cdot J_\varphi(x) - E_d\|_F^2 dx \quad (2.23)$$

$$B, J_\varphi : \mathbb{R}^d \rightarrow \mathbb{R}^{d \times d}, \quad E_d \in \mathbb{R}^{d \times d}$$

Mit der Funktion \mathcal{P} und einem dazugehörigen positiven Gewichtungsfaktor β lässt sich die erweiterte nichtparametrische Zielfunktion wie folgt darstellen:

$$\mathcal{J}_{\text{npb}}(\varphi, B) = \mathcal{D}(B(T \circ \varphi), R) + \alpha \mathcal{S}(u) + \beta \mathcal{P}(\varphi, B) \quad (2.24)$$

dabei steht npb für „nicht parametrisch mit Basistransformation“

$$\mathcal{J}_{\text{npb}} : V \times \mathbb{R}^{d \times d} \rightarrow \mathbb{R}, \quad \mathcal{J}_{\text{npb}} : (\varphi, B) \mapsto \mathcal{J}_{\text{npb}}(\varphi, B),$$

mit:

$$V \subset \left\{ \varphi \mid \varphi : \mathbb{R}^d \rightarrow \mathbb{R}^d \right\}, \quad B : \mathbb{R}^d \rightarrow \mathbb{R}^{d \times d}$$

$$\alpha, \beta \in \mathbb{R}^+, \quad T, R \in \mathcal{I}_{d,d}$$

Offensichtlich gilt:

$$\mathcal{J}_{\text{npb}}(\varphi, (J_\varphi)^{-1}) = \mathcal{J}_{\text{np}}(\varphi) + \beta \mathcal{P}(\varphi, (J_\varphi)^{-1})$$

da sich die nichtparametrische Zielfunktionen \mathcal{J}_{np} aus (2.22) und deren Erweiterung \mathcal{J}_{npb} aus (2.24) nur durch den Strafterm \mathcal{P} unterscheiden.

Exemplarisch für $d = 2$ ergibt sich aus (2.23) mit der neuen Parametermatrix B

$$B(x) := \begin{pmatrix} B_{11}(x) & B_{12}(x) \\ B_{21}(x) & B_{22}(x) \end{pmatrix} \in \mathbb{R}^{2 \times 2}$$

folgender Strafterm:

$$\begin{aligned} \mathcal{P}(\varphi, B) &= \left\| \begin{pmatrix} B_{11} & B_{12} \\ B_{21} & B_{22} \end{pmatrix} \begin{pmatrix} \partial_1 \varphi_1 & \partial_2 \varphi_1 \\ \partial_1 \varphi_2 & \partial_2 \varphi_2 \end{pmatrix} - \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \right\|_F^2 \\ &= \left\| \begin{pmatrix} B_{11} \cdot \partial_1 \varphi_1 + B_{12} \cdot \partial_1 \varphi_2 - 1 & B_{11} \cdot \partial_2 \varphi_1 + B_{12} \cdot \partial_2 \varphi_2 \\ B_{21} \cdot \partial_1 \varphi_1 + B_{22} \cdot \partial_1 \varphi_2 & B_{21} \cdot \partial_2 \varphi_1 + B_{22} \cdot \partial_2 \varphi_2 - 1 \end{pmatrix} \right\|_F^2 \\ &= (B_{11} \cdot \partial_1 \varphi_1 + B_{12} \cdot \partial_1 \varphi_2 - 1)^2 + (B_{11} \cdot \partial_2 \varphi_1 + B_{12} \cdot \partial_2 \varphi_2)^2 \\ &\quad + (B_{21} \cdot \partial_1 \varphi_1 + B_{22} \cdot \partial_1 \varphi_2)^2 + (B_{21} \cdot \partial_2 \varphi_1 + B_{22} \cdot \partial_2 \varphi_2 - 1)^2 \end{aligned}$$

Bisher sind drei verschiedene Zielfunktionen motiviert worden. Da die schrittweisen Erweiterungen aus den kompakten Schreibweisen der Funktionen besonders klar hervorgehen, werden diese im Folgenden nacheinander aufgelistet, um die Gemeinsamkeiten und Unterschiede der entsprechenden Zielfunktionen übersichtlich darzustellen.

2.7 Übersicht der Zielfunktionen

Im Folgenden gilt:

$$\alpha, \beta \in \mathbb{R}^+, \quad \varphi, T, R: \mathbb{R}^d \rightarrow \mathbb{R}^d, \quad A_\theta^{-1}, (J_\varphi)^{-1}, B \in \mathbb{R}^{d \times d}$$

Für parametrische Deformationen gibt es nur eine Basistransformationsmatrix, die für alle Datenpunkte identisch ist. Die zugehörige Zielfunktion lautet:

$$\mathcal{J}_p(\theta) = \mathcal{D}(A_\theta^{-1}(T \circ \varphi_\theta), R)$$

Bei einer Erweiterung auf nichtparametrische Deformationen, ist eine Basistransformation für jeden Datenpunkt notwendig. Zusätzlich werden glatte Deformationen φ begünstigt um zufriedenstellende Ergebnisse zu erhalten.

$$\mathcal{J}_{np}(\varphi) = \mathcal{D}((J_\varphi)^{-1}(T \circ \varphi), R) + \alpha \mathcal{S}(u)$$

Um die Invertierung der Jacobimatrix der Deformation φ zu vermeiden, wird die Basistransformation als zusätzlicher Optimierungsparameter aufgefasst. Zusätzlich werden mit dem Strafterm \mathcal{P} Basistransformationen bestraft, die der inversen Jacobimatrix von φ nicht geeignet ähnlich sind, um die Reorientierungseigenschaft der Basistransformation näherungsweise zu erhalten.

$$\mathcal{J}_{npb}(\varphi, B) = \mathcal{D}(B \cdot (T \circ \varphi), R) + \alpha \mathcal{S}(u) + \beta \mathcal{P}(\varphi, B)$$

Die Minimima der vorgestellten Zielfunktionen können nicht analytisch berechnet werden, da insbesondere bei den zwei nichtparametrischen Varianten die nichtlinearen Funktionen wie Regularisierer \mathcal{S} und Strafterm \mathcal{P} große Probleme bereiten. Es ist üblich den Ansatz „first-discretize-then-optimize“ zu verwenden, der vorsieht, dass die Zielfunktion eines Optimierungsproblems erst diskretisiert und dann mit einem numerischen Optimierungsverfahren minimiert wird. Diese Herangehensweise hat sich bereits in verschiedenen Anwendungen mit sehr guten Resultaten bewährt, siehe z.B. [4].

Im nächsten Kapitel werden die vorgestellten Zielfunktionen diskretisiert. Zusätzlich werden die zugehörigen Ableitungen hergeleitet, da diese für die numerischen Optimierungsverfahren notwendig sind.

3 Diskretisierung

Das im Kapitel 2 vorgestellte Registrierungsproblem ist nicht direkt mit analytischen Methoden lösbar, d.h. das Minimum der Zielfunktionen muss numerisch berechnet werden.

Das Ziel dieses Kapitels ist die Diskretisierung der im Kapitel 2 vorgestellten drei Zielfunktionen. Diese ist notwendig, um die numerischen Optimierungsverfahren zur Lösung des Bildregistrierungsproblems anwenden zu können.

Da die nichtparametrischen Kostenfunktionen Erweiterungen der parametrischen sind, wird zu Beginn die Diskretisierung von \mathcal{J}_p vorgestellt. Aufbauend darauf werden die verbleibenden zwei nichtparametrischen Zielfunktionen diskretisiert.

Da zur Anwendung der numerischen Optimierungsverfahren aus Kapitel 4 die Ableitungen der Funktionen benötigt werden, werden diese an der jeweiligen Stelle in diesem Kapitel angegeben.

Bestandteil aller Zielfunktionen ist das SSD-Maß, welches ein Integral enthält. Zur Approximation der Integrale des Distanzmaßes und der Strafterme wird die Mittelpunkregel verwendet. Zur Diskretisierung der Basistransformation und der Regularisierer ist außerdem die Approximation von partiellen Ableitungen erster und zweiter Ordnung notwendig, die im Folgenden durch finite Differenzen berechnet werden.

Zu Gunsten der Übersichtlichkeit, werden die Diskretisierungen für 2D beschrieben, eine Erweiterung für 3D ist analog möglich. Weiterhin bezeichnet $\Omega = (\omega_1, \omega_2) \times (\omega_3, \omega_4)$ den Bildbereich, und $m = (m_1, m_2)$ die Anzahl der Bildpunkte in der jeweiligen Dimension.

3.1 Parametrische Zielfunktion

Die parametrische Zielfunktion für Flussbilder $T, R \in \mathcal{I}_{2,2}$ lautet bisher:

$$\begin{aligned}\mathcal{J}_p(\theta) &= \mathcal{D}(A_\theta^{-1}(T \circ \varphi_\theta), R) \\ &= \frac{1}{2} \int_{\Omega} \|A_\theta^{-1} \cdot (T \circ \varphi_\theta(x)) - R(x)\|_2^2 dx \\ &= \frac{1}{2} \int_{\omega_3}^{\omega_4} \int_{\omega_1}^{\omega_2} \|A_\theta^{-1} \cdot (T \circ \varphi_\theta(x)) - R(x)\|_2^2 dx_1 dx_2\end{aligned}$$

Der wesentliche Bestandteil dieser Funktion ist das enthaltene Integral, daher wird es zuerst diskretisiert. Zur numerische Integration soll die *Mittelpunktregel* (Rechteckregel) verwendet werden.

Dafür wird der Bildbereich Ω gleichmäßig in Rechtecke unterteilt, um den Integranden jeweils an den Punkten auszuwerten, die in der Mitte jedes Teilintervalls liegen.

Die an den Stützstellen ausgewertete Funktion bildet dann die diskretisierte Repräsentation ihrer kontinuierlichen Abbildung.

3.1.1 Gitter

Um die vorliegenden kontinuierlichen Funktionen in ihre diskretisierten Entsprechungen zu überführen, wird ein gleichmäßiges Gitter mit $m_1 m_2$ Punkten definiert, an dem später alle Funktionen ausgewertet werden.

Beim Cell-Centered Gitter (*CC-Gitter*) liegen die Datenpunkte zentriert in jeder Zelle vor, so dass sich die Gitterpunkte $(x_i, y_j)_{(i,j)=(1,1)}^{(m_1, m_2)}$ auf dem Gebiet Ω mit den Schrittweiten h_1, h_2 wie folgt darstellen lassen:

$$\begin{aligned} x_i &= \omega_1 - \frac{h_1}{2} + ih_1, & i &= 1, \dots, m_1, & h_1 &= \frac{\omega_2 - \omega_1}{m_1} \\ y_j &= \omega_3 - \frac{h_2}{2} + jh_2, & j &= 1, \dots, m_2, & h_2 &= \frac{\omega_4 - \omega_3}{m_2} \end{aligned} \quad (3.1)$$

Der Bildbereich Ω kann daher auch wie folgt dargestellt werden:

$$\Omega = (\omega_1, \omega_2) \times (\omega_3, \omega_4) = \left(x_1 - \frac{h_1}{2}, x_{m_1} + \frac{h_1}{2}\right) \times \left(y_1 - \frac{h_2}{2}, y_{m_2} + \frac{h_2}{2}\right) \quad (3.2)$$

Ein Beispiel für ein CC-Gitter mit 16 Punkten ist in Abbildung 3.1 dargestellt.

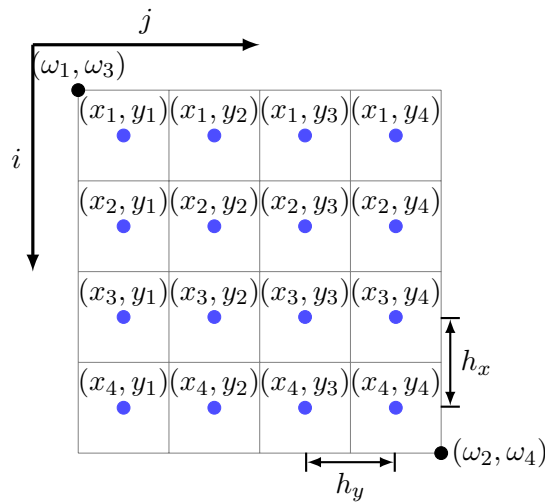


Abbildung 3.1: Veranschaulichung eines CC-Gitters mit $(m_1, m_2) = (4, 4)$ auf dem Bildbereich $\Omega = (\omega_1, \omega_2) \times (\omega_3, \omega_4)$

Numerische Integration mit der Mittelpunkregel (2D)

Angenommen der Integrationsbereich Ω einer zweifach stetig differenzierbaren Funktion $f : \mathbb{R}^2 \rightarrow \mathbb{R}$ wird durch ein gleichmäßiges zweidimensionales CC-Gitter mit Gitterpunkten (x_i, y_j) und der Auflösung (h_1, h_2) abgedeckt.

Dann kann das Integral der Funktion f über den Bereich Ω wie folgt dargestellt werden:

$$\int_{\Omega} f(x, y) \, dx dy = h_1 h_2 \sum_{i=1}^{m_1} \sum_{j=1}^{m_2} f(x_i, y_j) + \mathcal{O}(h_1^2 + h_2^2) \quad (3.3)$$

Da bei dieser Approximation jeweils nur die Funktionswerte verwendet werden, die zentriert in der Mitte einer Zelle vorliegen, wird sie als *Mittelpunktregel* bezeichnet [3].

Im Folgenden wird diese angewendet um das Integral des SSD-Maßes zu approximieren. Die in der Zielfunktion \mathcal{J}_p zusätzlich enthaltene Deformation und die Basistransformation werden im Anschluss ergänzt.

3.1.2 Distanzmaß

Das SSD-Maß für vektorwertige Bilder $T, R \in \mathcal{I}_{2,2}$ lautet

$$\begin{aligned} \mathcal{D}(T, R) &= \frac{1}{2} \int_{\Omega} \|T(x) - R(x)\|_2^2 dx \\ &= \frac{1}{2} \int_{\omega_3}^{\omega_4} \int_{\omega_1}^{\omega_2} \sum_{k=1}^2 (T_k(x_1, x_2) - R_k(x_1, x_2))^2 dx_1 dx_2 \end{aligned}$$

Unter der Voraussetzung, dass die Bilder R und T zweifach stetig differenzierbare Funktionen sind und der Bildbereich Ω durch ein gleichmäßiges zweidimensionales CC-Gitter wie in (3.1) abgedeckt wird, kann das Integral mit der Mittelpunktregel approximiert werden.

Angewandt auf das SSD-Maß für vektorwertige Bilder $T, R \in \mathcal{I}_{2,2}$ ergibt sich mit Auflösungen $h = (h_1, h_2)$ demzufolge:

$$\mathcal{D}(T, R) = \frac{h_1 h_2}{2} \sum_{i=1}^{m_1} \sum_{j=1}^{m_2} \sum_{k=1}^2 (T_k(x_i, y_j) - R_k(x_i, y_j))^2 + \mathcal{O}(h_1^2 + h_2^2)$$

Wird der Fehlerterm vernachlässigt, ergibt sich folgende Diskretisierung:

$$\mathcal{D}^h(T, R) = \frac{h_1 h_2}{2} \sum_{i=1}^{m_1} \sum_{j=1}^{m_2} \sum_{k=1}^2 (T_k(x_i, y_j) - R_k(x_i, y_j))^2$$

Die Summenschreibweise kann durch Verwendung einer Vektorschreibweise für die diskretisierten Bilder vereinfacht werden.

3.1.3 Vektorschreibweise

Im Folgenden werden die diskretisierten Bilder, welche sich durch punktweise Auswertungen ergeben, unter Verwendung des CC-Gitters jeweils in einem Vektor zusammengefasst.

Mit

$$\begin{aligned} x &= (x_1, \dots, x_{m_1})^T \in \mathbb{R}^{m_1} \\ y &= (y_1, \dots, y_{m_2})^T \in \mathbb{R}^{m_2} \end{aligned}$$

kann das CC-Gitter in einem Vektor \bar{x} zusammengefasst werden

$$\bar{x} = \begin{pmatrix} x \otimes \mathbb{1}_{m_2} \\ \mathbb{1}_{m_1} \otimes y \end{pmatrix} = \left(\overbrace{\underbrace{x_1, \dots, x_1}_{m_2}, \dots, \underbrace{x_{m_1}, \dots, x_{m_1}}_{m_2}}^{m_2 \cdot m_1} \mid \overbrace{\underbrace{y_1, y_2, \dots, y_{m_2}}_{m_2}, \dots, \underbrace{y_1, y_2, \dots, y_{m_2}}_{m_2}}^{m_1 \cdot m_2} \right)^T \in \mathbb{R}^{2m_1 m_2}$$

Die Gitterpunkte (x_i, y_j) können mit dem Gittervektor \bar{x} wie folgt in Verbindung gebracht werden

$$(x_i, y_j) = (\bar{x}_{i+(j-1)m_1}, \bar{x}_{i+(j-1)m_1+m_1 m_2}), \quad i = 1, \dots, m_1, \quad j = 1, \dots, m_2$$

In \bar{x} sind die ersten $m_1 m_2$ Einträge die x -Werte, die $m_1 m_2$ danach entsprechen den y -Werten.

1 •	5 •	9 •	13 •
2 •	6 •	10 •	14 •
3 •	7 •	11 •	15 •
4 •	8 •	12 •	16 •

Abbildung 3.2: Veranschaulichung eines CC-Gitters mit $(m_1, m_2) = (4, 4)$ auf dem Bildbereich $\Omega = (\omega_1, \omega_2) \times (\omega_3, \omega_4)$ mit der korrespondierenden Nummerierung für \bar{x} . Die zwei Komponenten des Datenpunkts mit Nummer k ergeben sich aus dem Gittervektor \bar{x} als $(\bar{x}_k, \bar{x}_{k+m_1 m_2})$.

Die Bilder $T, R \in \mathcal{I}_{2,2}$ können punktweise an den Gitterpunkten ausgewertet werden, so dass jedem der $m_1 m_2$ Gitterpunkte jeweils zwei Werte zugewiesen werden. Das diskretisierte, an allen Gitterpunkten ausgewertete Bild T wird ab jetzt mit \bar{T} bezeichnet.

$$\begin{aligned} \bar{T} &: \mathbb{R}^{2m_1 m_2} \rightarrow \mathbb{R}^{2m_1 m_2}, \quad \bar{x} \mapsto \bar{T}(\bar{x}), \\ \bar{T} &:= \bar{T}(\bar{x}) = \begin{pmatrix} \bar{T}_1(\bar{x}) \\ \bar{T}_2(\bar{x}) \end{pmatrix} \in \mathbb{R}^{2m_1 m_2}, \quad \bar{T}_k \in \mathbb{R}^{m_1 m_2} \\ \bar{T}_k &:= \bar{T}_k(\bar{x}) = \begin{pmatrix} T_k(\bar{x}_1, \bar{x}_{m_1 m_2+1}) \\ T_k(\bar{x}_2, \bar{x}_{m_1 m_2+2}) \\ \vdots \\ T_k(\bar{x}_{m_1 m_2}, \bar{x}_{2m_1 m_2}) \end{pmatrix} \in \mathbb{R}^{m_1 m_2}, \quad T_k \in \mathcal{I}_2 \end{aligned} \quad (3.4)$$

Die diskretisierte, auf dem CC-Gitter \bar{x} ausgewertete Deformation $\bar{\varphi}$ kann auf dieselbe Art dargestellt werden.

$$\bar{\varphi} : \mathbb{R}^{2m_1 m_2} \rightarrow \mathbb{R}^{2m_1 m_2}, \quad \bar{x} \mapsto \bar{\varphi}(\bar{x}),$$

$$\begin{aligned}\bar{\varphi} &:= \bar{\varphi}(\bar{x}) = \begin{pmatrix} \bar{\varphi}_1(\bar{x}) \\ \bar{\varphi}_2(\bar{x}) \end{pmatrix} \in \mathbb{R}^{2m_1m_2}, \quad \bar{\varphi}_k \in \mathbb{R}^{m_1m_2} \\ \bar{\varphi}_k &: \mathbb{R}^{2m_1m_2} \rightarrow \mathbb{R}^{m_1m_2}, \quad \bar{x} \mapsto \bar{\varphi}_k(\bar{x}), \\ \bar{\varphi}_k &:= \bar{\varphi}_k(\bar{x}) = \begin{pmatrix} \varphi_k(\bar{x}_1, \bar{x}_{m_1m_2+1}) \\ \varphi_k(\bar{x}_2, \bar{x}_{m_1m_2+2}) \\ \vdots \\ \varphi_k(\bar{x}_{m_1m_2}, \bar{x}_{2m_1m_2}) \end{pmatrix} \in \mathbb{R}^{m_1m_2}\end{aligned}$$

Da die parametrische Deformation vollständig durch den Parametervektor θ beschrieben wird, wird die vorgestellte Diskretisierung für die affin-linearen Deformationen konkretisiert.

Die kontinuierliche Darstellung der parametrischen Deformation $\varphi_\theta : \mathbb{R}^2 \rightarrow \mathbb{R}^2$, $\theta \in \mathbb{R}^6$ lautet:

$$\varphi_\theta(x) = A_\theta x + b_\theta = \begin{pmatrix} \theta_1 x_1 + \theta_2 x_2 + \theta_3 \\ \theta_4 x_1 + \theta_5 x_2 + \theta_6 \end{pmatrix} = \begin{pmatrix} x_1 & x_2 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & x_1 & x_2 & 1 \end{pmatrix} \cdot \theta$$

so dass die diskretisierte Deformation wie folgt dargestellt werden kann:

$$\bar{\varphi}_\theta(\bar{x}) = Q \cdot \theta \in \mathbb{R}^{2m_1m_2} \tag{3.5}$$

mit

$$\begin{aligned}P_1 &= (E_{m_1m_2} \ Z_{m_1m_2}), \quad P_2 = (Z_{m_1m_2} \ E_{m_1m_2}) \in \mathbb{R}^{m_1m_2 \times 2m_1m_2} \\ Q_0 &= (P_1 \cdot \bar{x}, P_2 \cdot \bar{x}, \mathbf{1}_{m_1m_2}) \in \mathbb{R}^{m_1m_2 \times 3}, \quad Q = \begin{pmatrix} Q_0 & 0 \\ 0 & Q_0 \end{pmatrix} \in \mathbb{R}^{2m_1m_2 \times 6}\end{aligned}$$

Das diskretisierte deformierte Template ergibt sich folglich zu

$$\bar{T}(\bar{\varphi}) \text{ bzw. } \bar{T}(\bar{\varphi}_\theta)$$

Das diskretisierte SSD-Maß zum Vergleich zweier Bilder kann jetzt, unter Verwendung der Vektorschreibweise für die Bilder R und T wie folgt dargestellt werden:

$$\mathcal{D}^h(\bar{T}, \bar{R}) = \frac{h_1 h_2}{2} \|\bar{T} - \bar{R}\|_2^2$$

Zur vollständigen Diskretisierung der parametrischen Zielfunktion muss nun noch die Basistransformation diskretisiert werden.

3.1.4 Basistransformation

Die bisherigen Ansätze motivieren dazu die Basistransformation als eine Matrixmultiplikation der Form $\bar{B} \cdot \bar{T}$, mit $\bar{B}(\bar{\varphi}) \in \mathbb{R}^{2m_1m_2 \times 2m_1m_2}$ darzustellen.

Aus dem Abschnitt 2.5 geht hervor, dass die Basistransformationsmatrix die Inverse der Jacobimatrix von φ ist. Im parametrischen Fall kann die Jacobimatrix von φ aus den übergebenen Parametern θ direkt angegeben werden. In diesem Fall muss nur noch die Inverse der Matrix A_θ , als Bestandteil der affinen Transformation, berechnet werden.

Im zweidimensionalen Fall ergeben sich sechs Parameter $\theta \in \mathbb{R}^6$. Wie aus der kontinuierlichen Darstellung von $(J_\varphi)^{-1}$ in Gleichung (2.20) hervorgeht, sind vier davon Bestandteil der Basistransformationsmatrix.

Liegt eine parametrische Deformation φ_θ der folgenden Form vor

$$\begin{aligned} \varphi_\theta : \mathbb{R}^2 &\rightarrow \mathbb{R}^2, \quad \varphi_\theta : x \mapsto \varphi_\theta(x) \\ \varphi_\theta(x) &= A_\theta x + b_\theta, \quad A_\theta = \begin{pmatrix} \theta_1 & \theta_2 \\ \theta_4 & \theta_5 \end{pmatrix} \in \mathbb{R}^{2 \times 2}, \quad b_\theta = (\theta_3, \theta_6)^T, x \in \mathbb{R}^2, \quad \theta_i \in \mathbb{R} \end{aligned}$$

folgt für die Basistransformationsmatrix

$$B(\varphi_\theta) = (J_\varphi)^{-1} \stackrel{(2.20)}{=} A_\theta^{-1} = \begin{pmatrix} \theta_1 & \theta_2 \\ \theta_4 & \theta_5 \end{pmatrix}^{-1} = \frac{1}{\theta_1\theta_5 - \theta_2\theta_4} \begin{pmatrix} \theta_5 & -\theta_2 \\ -\theta_4 & \theta_1 \end{pmatrix} \quad (3.6)$$

Da diese Matrix offensichtlich unabhängig von x ist, folgt dass diese Basistransformationsmatrix für alle Datenpunkte identisch ist. Das heißt für den diskreten Fall, dass jeder der $m_1 m_2$ Gitterpunkte genau diese Basistransformationsmatrix B besitzt, so dass diese Matrix verwendet werden kann um die Basistransformationsmatrix $\bar{B} \in \mathbb{R}^{2m_1 m_2 \times 2m_1 m_2}$ zur Reorientierung von vektorwertigen, diskretisierten Bildern $\bar{T} \in \mathbb{R}^{2m_1 m_2}$ zu konstruieren.

Wird die Basistransformation auf ein zweidimensionales Flussbild $T = (T_1, T_2)^T \in \mathcal{I}_{2,2}$ angewandt, ergibt sich:

$$B(\varphi_\theta) \cdot T = A_\theta^{-1} \cdot T = \frac{1}{\theta_1\theta_5 - \theta_2\theta_4} \begin{pmatrix} \theta_5 T_1 - \theta_2 T_2 \\ -\theta_4 T_1 + \theta_1 T_2 \end{pmatrix} \in \mathbb{R}^2$$

Die Erweiterung auf diskretisierte Flussbilder \bar{T} kann analog beschrieben werden als

$$\begin{aligned} \bar{B}(\bar{\varphi}_\theta) \cdot \bar{T} &= \frac{1}{\theta_1\theta_5 - \theta_2\theta_4} \begin{pmatrix} \theta_5 \bar{T}_1 - \theta_2 \bar{T}_2 \\ -\theta_4 \bar{T}_1 + \theta_1 \bar{T}_2 \end{pmatrix} \in \mathbb{R}^{2m_1 m_2} \\ \bar{T} &= \begin{pmatrix} \bar{T}_1 \\ \bar{T}_2 \end{pmatrix} \in \mathbb{R}^{2m_1 m_2}, \quad \bar{T}_1, \bar{T}_2 \in \mathbb{R}^{m_1 m_2} \end{aligned}$$

Die diskrete Basistransformation $\bar{B} \in \mathbb{R}^{2m_1 m_2}$ ergibt sich folglich als eine Blockdiagonalmatrix der Form

$$\bar{B}(\bar{\varphi}_\theta) = \begin{pmatrix} \backslash & \backslash \\ \backslash & \backslash \end{pmatrix}$$

mit

$$\begin{aligned} \bar{B}(\bar{\varphi}_\theta) &:= \frac{1}{\theta_1\theta_5 - \theta_2\theta_4} \begin{pmatrix} \theta_5 P_1 - \theta_2 P_2 \\ -\theta_4 P_1 + \theta_1 P_2 \end{pmatrix} \in \mathbb{R}^{2m_1 m_2 \times 2m_1 m_2} \\ P_1 &:= (E_{m_1 m_2} \quad Z_{m_1 m_2}) \in \mathbb{R}^{m_1 m_2 \times 2m_1 m_2} \\ P_2 &:= (Z_{m_1 m_2} \quad E_{m_1 m_2}) \in \mathbb{R}^{m_1 m_2 \times 2m_1 m_2} \end{aligned} \quad (3.7)$$

Das basistransformierte, deformierte Template für parametrische Deformationen kann somit wie folgt dargestellt werden:

$$\overline{B}(\overline{\varphi}_\theta) \cdot \overline{T}(\overline{\varphi}_\theta) \in \mathbb{R}^{2m_1 m_2} \quad (3.8)$$

Als Ableitung der Basistransformation \overline{B} würde sich an dieser Stelle ein Tensor ergeben. Die vorliegende Matrix-Vektor-Multiplikation kann als eine einfache Vektormultiplikation dargestellt werden, so dass sich als Ableitung der Basistransformation eine Matrix ergibt.

Wird eine Hilfsfunktion b zur Invertierung der affinen Parameter θ verwendet, kann die bisherige Matrix-Vektormultiplikation zur Berechnung des basistransformierten, deformierten Templates wie folgt umformuliert werden:

$$\overline{B}(\overline{\varphi}_\theta) \cdot \overline{T}(\overline{\varphi}_\theta) = \begin{pmatrix} b_{11}(\theta) \cdot \overline{T}_1(\overline{\varphi}_\theta) + b_{12}(\theta) \cdot \overline{T}_2(\overline{\varphi}_\theta) \\ b_{21}(\theta) \cdot \overline{T}_1(\overline{\varphi}_\theta) + b_{22}(\theta) \cdot \overline{T}_2(\overline{\varphi}_\theta) \end{pmatrix} \quad (3.9)$$

mit $b_{ij} : \mathbb{R}^6 \rightarrow \mathbb{R}$, $i, j \in \{1, 2\}$, als Teilfunktionen von $b : \mathbb{R}^6 \rightarrow \mathbb{R}^4$

$$b(\theta) = \begin{pmatrix} b_{11}(\theta) \\ b_{21}(\theta) \\ b_{12}(\theta) \\ b_{22}(\theta) \end{pmatrix} = \begin{pmatrix} c(\theta) \cdot \theta_5 \\ -c(\theta) \cdot \theta_4 \\ -c(\theta) \cdot \theta_2 \\ c(\theta) \cdot \theta_1 \end{pmatrix} \in \mathbb{R}^4$$

mit $c : \mathbb{R}^6 \rightarrow \mathbb{R}$

$$c(\theta) = \frac{1}{\theta_1 \theta_5 - \theta_2 \theta_4}$$

Die Teilfunktionen b_{ij} entsprechen den Elementen der Basistransformationsmatrix aus (3.6). Die Bestandteile der parametrischen Zielfunktion \mathcal{J}_p sind vollständig diskretisiert worden und können in der diskretisierten Zielfunktion \mathcal{J}_p^h zusammengeführt werden.

3.1.5 Die diskretisierte parametrische Zielfunktion und ihre Ableitung

Die parametrische Zielfunktion lautet

$$\mathcal{J}_p(\theta) = \mathcal{D}(A_\theta^{-1}(T \circ \varphi_\theta), R)$$

Die diskretisierte Zielfunktion kann durch ihre diskretisierten Bestandteile wie folgt beschrieben werden

$$\mathcal{J}_p^h(\theta) = \mathcal{D}^h(\overline{B}(\overline{\varphi}_\theta) \cdot \overline{T}(\overline{\varphi}_\theta), \overline{R})$$

Mit den Hilfsfunktionen aus (3.9) folgt

$$\begin{aligned} \mathcal{J}_p^h(\theta) &= \mathcal{D}^h \left(\begin{pmatrix} b_{11}(\theta) \cdot \overline{T}_1(\overline{\varphi}_\theta) + b_{12}(\theta) \cdot \overline{T}_2(\overline{\varphi}_\theta) \\ b_{21}(\theta) \cdot \overline{T}_1(\overline{\varphi}_\theta) + b_{22}(\theta) \cdot \overline{T}_2(\overline{\varphi}_\theta) \end{pmatrix}, \overline{R} \right) \\ &= \frac{h_1 h_2}{2} \left\| \begin{pmatrix} b_{11}(\theta) \cdot \overline{T}_1(\overline{\varphi}_\theta) + b_{12}(\theta) \cdot \overline{T}_2(\overline{\varphi}_\theta) - \overline{R}_1 \\ b_{21}(\theta) \cdot \overline{T}_1(\overline{\varphi}_\theta) + b_{22}(\theta) \cdot \overline{T}_2(\overline{\varphi}_\theta) - \overline{R}_2 \end{pmatrix} \right\|_2^2 \end{aligned} \quad (3.10)$$

Die zugehörige Ableitung wird im Folgenden bestimmt.

Um diesen Vorgang zu vereinfachen, wird eine Hilfsfunktion r definiert, welche die Differenz des basistransformierten, deformierten Templates und des Referenzbildes beschreibt:

$$r(\theta) = \begin{pmatrix} b_{11}(\theta) \cdot \bar{T}_1(\bar{\varphi}_\theta) + b_{12}(\theta) \cdot \bar{T}_2(\bar{\varphi}_\theta) - \bar{R}_1 \\ b_{21}(\theta) \cdot \bar{T}_1(\bar{\varphi}_\theta) + b_{22}(\theta) \cdot \bar{T}_2(\bar{\varphi}_\theta) - \bar{R}_2 \end{pmatrix} \in \mathbb{R}^{2m_1m_2}$$

Die parametrische diskretisierte Zielfunktion kann durch das Residuum r wie folgt beschrieben werden

$$\mathcal{J}_p^h(\theta) = \frac{h_1 h_2}{2} r(\theta)^T r(\theta)$$

Zur Herleitung der Ableitung der parametrischen Zielfunktion wird zunächst die innere Funktion r differenziert. Dazu ist wiederum die Bestimmung der Ableitung der enthaltenen Teilfunktionen notwendig.

Die Ableitung von r lautet

$$\nabla r(\theta) = \left(\sum_{k=1}^2 \nabla \left(b_{1k}(\theta) \cdot \bar{T}_k(\bar{\varphi}_\theta) \right), \sum_{k=1}^2 \nabla \left(b_{2k}(\theta) \cdot \bar{T}_k(\bar{\varphi}_\theta) \right) \right) \in \mathbb{R}^{6 \times 2m_1m_2}$$

Die zugehörige Ableitung der einzelnen Komponenten nach θ lautet:

$$\nabla \left(b_{ij}(\theta) \cdot \bar{T}_k(\bar{\varphi}_\theta) \right) = \nabla b_{ij}(\theta) \cdot \bar{T}_k(\bar{\varphi}_\theta)^T + b_{ij}(\theta) \cdot \underbrace{\nabla \bar{\varphi}_\theta \cdot \nabla_x \bar{T}_k(\bar{\varphi}_\theta)}_{\nabla \bar{T}_k(\bar{\varphi}_\theta)} \in \mathbb{R}^{6 \times m_1m_2}$$

Die Ableitung des Templates $\nabla \bar{T}_k$ ergibt sich aus der Ableitung der gewählten Bildfunktion aus \mathcal{I}_2 (siehe 3.4).

Mit der diskretisierten parametrischen Deformation $\bar{\varphi}_\theta(\bar{x}) = Q \cdot \theta \in \mathbb{R}^{2m_1m_2}$ aus (3.5) folgt die zugehörige Ableitung

$$\nabla \bar{\varphi}_\theta = Q^T \in \mathbb{R}^{6 \times 2m_1m_2}$$

Die Ableitungen der Teilfunktionen der Basistransformation lauten wie folgt:

$$\begin{aligned} \nabla c(\theta) &= c(\theta)^2 \cdot (-1, 1, 0, 1, -1, 0)^T \in \mathbb{R}^6 \\ \nabla b_{11}(\theta) &= \theta_5 \cdot \nabla c(\theta) + c(\theta) \cdot (0, 0, 0, 0, 1, 0)^T \in \mathbb{R}^6 \\ \nabla b_{21}(\theta) &= -\theta_4 \cdot \nabla c(\theta) + c(\theta) \cdot (0, 0, 0, -1, 0, 0)^T \\ \nabla b_{12}(\theta) &= -\theta_2 \cdot \nabla c(\theta) + c(\theta) \cdot (0, -1, 0, 0, 0, 0)^T \\ \nabla b_{22}(\theta) &= \theta_1 \cdot \nabla c(\theta) + c(\theta) \cdot (1, 0, 0, 0, 0, 0)^T \end{aligned}$$

Die Ableitung der parametrischen Zielfunktion kann entsprechend wie folgt angegeben werden:

$$\nabla J_p^h(\theta) = h_1 h_2 \nabla r(\theta) \cdot r(\theta) \in \mathbb{R}^6$$

Im Folgenden werden die verbleibenden zwei nichtparametrischen Zielfunktionen diskretisiert.

3.2 Nichtparametrische Zielfunktion

Die nichtparametrische Zielfunktion für Flussbilder $T, R \in \mathcal{I}_{2,2}$ lautet

$$\mathcal{J}_{\text{np}}(\varphi) = \mathcal{D}((J_\varphi)^{-1}(T \circ \varphi), R) + \alpha \mathcal{S}(u)$$

Die Zielfunktionen \mathcal{J}_{np} und \mathcal{J}_{p} enthalten jeweils das SSD-Maß, das bereits diskretisiert wurde. Allerdings unterscheiden sie sich hinsichtlich der Basistransformation, die in dem Argument des Distanzmaßes enthalten ist.

Im Folgenden wird die Diskretisierung der Basistransformation für nichtparametrische Deformationen vorgestellt.

3.2.1 Basistransformation

Für nichtparametrischen Deformationen kann die zugehörige Jacobimatrix nicht durch einen Parametervektor angegeben werden. Da sie zudem nicht ortsinvariant ist, ist es notwendig, die Jacobimatrix zu approximieren und anschließend an allen Gitterpunkten auszuwerten. Im Folgenden wird erläutert wie die an den Gitterpunkten \bar{x} ausgewertete Deformation $\bar{\varphi}$ mit zentralen Differenzenquotienten numerisch differenziert werden kann und wie die daraus entstehende Approximation der Jacobimatrix J_φ zur gesuchten Basistransformation führt.

Numerische Differentiation mit finiten Differenzen

In diesem Abschnitt wird eine Schätzung der partiellen Ableitungen von φ vorgestellt. Jede Komponente φ_k des Vektorfeldes φ , bildet einen Vektor auf einen reellen Wert ab und ist daher eine skalarwertige Funktion.

Unter der Annahme, dass φ_k dreimal stetig differenzierbar ist, kann die Taylorentwicklung zweiten Grades formuliert und ausgewertet werden. Für die Taylorentwicklung am Ort x in Richtung $\pm h e_j$, wobei $h \in \mathbb{R}$ und e_j der j -te Einheitsvektor ist, gilt

$$\varphi_k(x + h e_j) = \varphi_k(x) + h e_j^T \nabla \varphi_k(x) + \frac{h^2}{2} e_j^T \nabla^2 \varphi_k(x) e_j + \mathcal{O}(h^3)$$

und

$$\varphi_k(x - h e_j) = \varphi_k(x) - h e_j^T \nabla \varphi_k(x) + \frac{h^2}{2} e_j^T \nabla^2 \varphi_k(x) e_j + \mathcal{O}(h^3)$$

Durch Addition der beiden Gleichungen folgt

$$\begin{aligned} \varphi_k(x + h e_j) - \varphi_k(x - h e_j) &= 2h e_j^T \nabla \varphi_k(x) + \mathcal{O}(h^3) \\ \Leftrightarrow e_j^T \nabla \varphi_k(x) &= \frac{1}{2h} (\varphi_k(x + h e_j) - \varphi_k(x - h e_j)) + \mathcal{O}(h^2) \end{aligned}$$

Dies entspricht der j -ten partiellen Ableitung von φ_k am Ort x , das heißt

$$\partial_j \varphi_k(x) = e_j^T \nabla \varphi_k(x)$$

Als Approximation für die j -te partielle Ableitung von φ_k ergibt sich entsprechend

$$\partial_j^h \varphi_k(x) := \frac{1}{2h} (\varphi_k(x + he_j) - \varphi_k(x - he_j))$$

Der dabei entstehende Quotient wird als *zentraler Differenzenquotient* bezeichnet. Übertragen auf die CC-Gitterpunkte in \bar{x} ergibt sich als Approximation der partiellen Ableitungen

$$\partial_1 \varphi_k(x_i, y_j) \approx \partial_1^{h_1} \varphi_k(x_i, y_j) := \frac{1}{2h_1} (\varphi_k(x_{i+1}, y_j) - \varphi_k(x_{i-1}, y_j))$$

$$\partial_2 \varphi_k(x_i, y_j) \approx \partial_2^{h_2} \varphi_k(x_i, y_j) := \frac{1}{2h_2} (\varphi_k(x_i, y_{j+1}) - \varphi_k(x_i, y_{j-1}))$$

Die Jacobimatrix von φ ausgewertet an einen Gitterpunkt (x_i, y_j) kann somit wie folgt approximiert werden

$$J_\varphi(x_i, y_j) \approx J_\varphi^h(x_i, y_j) = \begin{pmatrix} \partial_1^{h_1} \varphi_1(x_i, y_j) & \partial_2^{h_2} \varphi_1(x_i, y_j) \\ \partial_1^{h_1} \varphi_2(x_i, y_j) & \partial_2^{h_2} \varphi_2(x_i, y_j) \end{pmatrix} \in \mathbb{R}^{2 \times 2} \quad (3.11)$$

Entsprechend lautet die zum Datenpunkt (x_i, y_j) gehörige approximierte Basistransformation

$$\begin{aligned} B^h(\varphi)(x_i, y_j) &= \left(J_\varphi^h(x_i, y_j) \right)^{-1} \\ &= c(\varphi)(x_i, y_j) \cdot \begin{pmatrix} \partial_2^{h_2} \varphi_2(x_i, y_j) & -\partial_2^{h_2} \varphi_1(x_i, y_j) \\ -\partial_1^{h_1} \varphi_2(x_i, y_j) & \partial_1^{h_1} \varphi_1(x_i, y_j) \end{pmatrix} \\ c(\varphi)(x_i, y_j) &:= \frac{1}{\partial_1^{h_1} \varphi_1(x_i, y_j) \cdot \partial_2^{h_2} \varphi_2(x_i, y_j) - \partial_1^{h_1} \varphi_2(x_i, y_j) \cdot \partial_2^{h_2} \varphi_1(x_i, y_j)} \end{aligned}$$

Für jeden Gitterpunkt ergibt sich eine Basistransformation dieser Art, was demzufolge zu $m_1 m_2$ (2×2) Matrizen führt. Deren Elemente werden im Folgenden aus $\bar{\varphi}$ geschätzt und in eine große Basistransformationsmatrix $\bar{B}(\bar{\varphi})$ zusammengefasst, so dass die Reorientierung aller Flussvektoren \bar{T} mit einer einzigen Matrixmultiplikation ausgeführt werden kann, d.h. die reorientierten, deformierten Flussbilder lassen sich schreiben als $\bar{B}(\bar{\varphi}) \cdot \bar{T}(\bar{\varphi})$.

Die Basistransformation für nichparametrische Deformationen entsteht durch die Berechnung der Inversen der Jacobimatrix von φ . Zur Herleitung der Basistransformationsmatrix \bar{B} bietet es sich daher an diese zwei Vorgänge in zwei unterschiedlichen Funktionen abzubilden, die anschließend verbunden werden.

Für eine kompakte Schreibweise der Ableitungen von φ wird folgende Hilfsmatrix definiert

$$D_n := \begin{pmatrix} 0 & 1 & & & \\ -1 & \ddots & \ddots & & \\ & \ddots & \ddots & 1 & \\ & & & -1 & 0 \end{pmatrix} \in \mathbb{R}^{n \times n}, \quad n \in \mathbb{N}$$

Die approximierte partielle Ableitung der ersten Dimension der Teilfunktion φ_k , ausgewertet an allen Gitterpunkten, kann dann wie folgt dargestellt werden

$$D_x \cdot \bar{\varphi}_k(\bar{x}) = \partial_1^{h_1} \bar{\varphi}_k(\bar{x}) = \begin{pmatrix} \partial_1^{h_1} \varphi_k(x_1, y_1) \\ \partial_1^{h_1} \varphi_k(x_1, y_2) \\ \vdots \\ \partial_1^{h_1} \varphi_k(x_{m_1}, y_{m_2}) \end{pmatrix} \in \mathbb{R}^{m_1 m_2}, \quad k = 1, 2 \quad (3.12)$$

mit

$$D_x = \frac{1}{2h_1} (E_{m_2} \otimes D_{m_1}) \in \mathbb{R}^{m_1 m_2 \times m_1 m_2}$$

Analog für die zweite Raumdimension

$$D_y \cdot \bar{\varphi}_k(\bar{x}) = \partial_2^{h_2} \bar{\varphi}_k(\bar{x})$$

mit folgender Matrix

$$D_y = \frac{1}{2h_2} (D_{m_2} \otimes E_{m_1}) \in \mathbb{R}^{m_1 m_2 \times m_1 m_2}$$

Die Elemente der vier Vektoren $D_x \cdot \bar{\varphi}_1(\bar{x})$, $D_x \cdot \bar{\varphi}_2(\bar{x})$, $D_y \cdot \bar{\varphi}_1(\bar{x})$ und $D_y \cdot \bar{\varphi}_2(\bar{x})$ enthalten jeweils eines der vier Elemente der Basistransformation, die zu einem Gitterpunkt gehört.

3.2.2 Approximation der Jacobimatrix

Die Schätzung der ersten partiellen Ableitung ausgewertet an allen Gitterpunkten, wurde bereits in (3.12) vorgestellt. Die partielle Ableitung der Komponente φ_k entlang einer Dimension kann mittels einer Matrix-Vektormultiplikation aus $\bar{\varphi}$ näherungsweise bestimmt werden. Die folgende Funktion $J : \mathbb{R}^{2m_1 m_2} \rightarrow \mathbb{R}^{4m_1 m_2}$ mit Teilfunktionen $J_{ij} : \mathbb{R}^{2m_1 m_2} \rightarrow \mathbb{R}^{m_1 m_2}$ approximiert alle notwendigen Ableitungen:

$$J(\bar{\varphi}) = \begin{pmatrix} J_{11}(\bar{\varphi}) \\ J_{21}(\bar{\varphi}) \\ J_{12}(\bar{\varphi}) \\ J_{22}(\bar{\varphi}) \end{pmatrix} := \begin{pmatrix} D_x \bar{\varphi}_1 \\ D_x \bar{\varphi}_2 \\ D_y \bar{\varphi}_1 \\ D_y \bar{\varphi}_2 \end{pmatrix} = D \cdot \bar{\varphi} \in \mathbb{R}^{4m_1 m_2} \quad (3.13)$$

mit der Nullmatrix $Z := Z_{m_1 m_2}$ und

$$D := \begin{pmatrix} D_x & Z \\ Z & D_x \\ D_y & Z \\ Z & D_y \end{pmatrix} \quad (3.14)$$

Die approximierte Jacobimatrix aus (3.11), ausgewertet an dem Gitterpunkt $(\bar{x}_k, \bar{x}_{k+m_1 m_2})$, ergibt sich durch Zusammenführen der k -ten Komponenten, bezeichnet mit $(J_{ij}(\bar{\varphi}))_k$, der vier Hilfsfunktionen J_{ij}

$$J_\varphi^h(\bar{x}_k, \bar{x}_{k+m_1 m_2}) = \begin{pmatrix} (J_{11}(\bar{\varphi}))_k & (J_{12}(\bar{\varphi}))_k \\ (J_{21}(\bar{\varphi}))_k & (J_{22}(\bar{\varphi}))_k \end{pmatrix} \in \mathbb{R}^{2 \times 2}$$

Diese Darstellung veranschaulicht, dass $J(\bar{\varphi})$ die Elemente der Jacobimatrizen für jeden der $(m_1 m_2)$ Datenpunkte enthält. Die Funktion, die jede einzelne dieser Matrizen invertiert, wird im Folgenden betrachtet.

3.2.3 Invertierung der approximierten Jacobimatrix

Die Funktion $b : \mathbb{R}^{4m_1m_2} \rightarrow \mathbb{R}^{4m_1m_2}$ invertiert die geschätzten Jacobimatrizen in J komponentenweise unter Verwendung der Teilfunktionen $b_{ij} : \mathbb{R}^{4m_1m_2} \rightarrow \mathbb{R}^{m_1m_2}$.

$$b(J) := \begin{pmatrix} b_{11}(J) \\ b_{21}(J) \\ b_{12}(J) \\ b_{22}(J) \end{pmatrix} := \begin{pmatrix} \bar{c}(J) \odot J_{22} \\ -\bar{c}(J) \odot J_{21} \\ -\bar{c}(J) \odot J_{12} \\ \bar{c}(J) \odot J_{11} \end{pmatrix} \in \mathbb{R}^{4m_1m_2} \quad (3.15)$$

$$\bar{c}(J) := \frac{1}{(J_{11} \odot J_{22} - J_{12} \odot J_{21})} \in \mathbb{R}^{m_1m_2}$$

wobei die Division hier elementweise gemeint ist.

Das basistransformierte, deformierte Template kann nun wie folgt dargestellt werden

$$\begin{pmatrix} b_{11}(J(\bar{\varphi})) \odot \bar{T}_1(\bar{\varphi}) + b_{12}(J(\bar{\varphi})) \odot \bar{T}_2(\bar{\varphi}) \\ b_{21}(J(\bar{\varphi})) \odot \bar{T}_1(\bar{\varphi}) + b_{22}(J(\bar{\varphi})) \odot \bar{T}_2(\bar{\varphi}) \end{pmatrix} \in \mathbb{R}^{2m_1m_2}$$

Für eine kompakte Schreibweise der Form $\bar{B}(\bar{\varphi}) \cdot \bar{T}(\bar{\varphi})$ wird die Basistransformationsmatrix \bar{B} wie folgt definiert

$$\bar{B}(\bar{\varphi}) := \begin{pmatrix} \text{diag}(b_{11}(J(\bar{\varphi}))) & \text{diag}(b_{12}(J(\bar{\varphi}))) \\ \text{diag}(b_{21}(J(\bar{\varphi}))) & \text{diag}(b_{22}(J(\bar{\varphi}))) \end{pmatrix} \in \mathbb{R}^{2m_1m_2 \times 2m_1m_2} \quad (3.16)$$

Die Basistransformation für nichtparametrische Deformationen wird im Folgenden in das diskretisierte Distanzmaß \mathcal{D}^h integriert.

Das Distanzmaß mit dem basistransformierten, deformierten Template für nichtparametrische Deformationen lautet

$$\mathcal{D}((J_\varphi)^{-1}(T \circ \varphi), R)$$

Analog zu der Diskretisierung für parametrische Deformationen aus (3.10) kann mit den Hilfsfunktionen b_{ij} aus (3.15) das enthaltene basistransformierte, deformierte Template wie folgt dargestellt werden

$$\begin{aligned} \mathcal{D}^h(\bar{B}(\bar{\varphi}) \cdot \bar{T}(\bar{\varphi}), \bar{R}) &= \mathcal{D}^h \left(\begin{pmatrix} b_{11}(J(\bar{\varphi})) \odot \bar{T}_1(\bar{\varphi}) + b_{12}(J(\bar{\varphi})) \odot \bar{T}_2(\bar{\varphi}) \\ b_{21}(J(\bar{\varphi})) \odot \bar{T}_1(\bar{\varphi}) + b_{22}(J(\bar{\varphi})) \odot \bar{T}_2(\bar{\varphi}) \end{pmatrix}, \bar{R} \right) \\ &= \frac{h_1 h_2}{2} \left\| \begin{pmatrix} b_{11}(J(\bar{\varphi})) \odot \bar{T}_1(\bar{\varphi}) + b_{12}(J(\bar{\varphi})) \odot \bar{T}_2(\bar{\varphi}) \\ b_{21}(J(\bar{\varphi})) \odot \bar{T}_1(\bar{\varphi}) + b_{22}(J(\bar{\varphi})) \odot \bar{T}_2(\bar{\varphi}) \end{pmatrix} - \bar{R} \right\|_2^2 \end{aligned}$$

Ebenfalls analog zum Fall für parametrische Deformationen kann das in der Norm enthaltene Argument in einer Hilfsfunktion r zusammengefasst werden

$$r(\bar{\varphi}) = \begin{pmatrix} b_{11}(J(\bar{\varphi})) \odot \bar{T}_1(\bar{\varphi}) + b_{12}(J(\bar{\varphi})) \odot \bar{T}_2(\bar{\varphi}) - \bar{R}_1 \\ b_{21}(J(\bar{\varphi})) \odot \bar{T}_1(\bar{\varphi}) + b_{22}(J(\bar{\varphi})) \odot \bar{T}_2(\bar{\varphi}) - \bar{R}_2 \end{pmatrix} \in \mathbb{R}^{2m_1m_2}$$

so dass sich das Distanzmaß wie folgt darstellen lässt:

$$\mathcal{D}^h(\bar{B}(\bar{\varphi}) \cdot \bar{T}(\bar{\varphi}), \bar{R}) = \frac{h_1 h_2}{2} r(\bar{\varphi})^T r(\bar{\varphi}) \quad (3.17)$$

Zur vollständigen Diskretisierung der Zielfunktion

$$\mathcal{J}_{\text{np}}(\varphi) = \mathcal{D}((J_\varphi)^{-1}(T \circ \varphi), R) + \alpha \mathcal{S}(u)$$

fehlt jetzt noch die Diskretisierung des Regularisierers S , die im Folgenden hergeleitet wird.

3.2.4 Regularisierer

Im Abschnitt 2.3 wurden zwei Regularisierer vorgestellt. In diesem Abschnitt werden die diskretisierten Varianten erst für den eindimensionalen, dann als Erweiterung für den zweidimensionalen Fall hergeleitet. Zur Vereinfachung der Schreibweise wird die Verrückung im ein- und zweidimensionalen Fall als $u(x) = \varphi(x) - x$ bezeichnet. Die entsprechende Dimension ergibt sich aus dem jeweiligen Kontext.

Diffusiver Glätter

Der eindimensionale diffusive Glätter lautet:

$$\mathcal{S}_{\text{diff}}(u) = \frac{1}{2} \int_a^b (u'(x))^2 dx$$

$$u : \mathbb{R} \rightarrow \mathbb{R}, \quad x \in \mathbb{R}, \quad \Omega = (a, b), \quad a < b$$

mit partieller Integration kann dieser umgeschrieben werden als:

$$\begin{aligned} \mathcal{S}_{\text{diff}}(u) &= \frac{1}{2} \int_a^b (u'(x))^2 dx = \frac{1}{2} \int_a^b u'(x) \cdot u'(x) dx \\ &= -\frac{1}{2} \int_a^b u''(x) \cdot u(x) dx + \frac{1}{2} \underbrace{[u'(x) \cdot u(x)]_a^b}_{\stackrel{(*)}{=} 0} \end{aligned} \quad (3.18)$$

Folgende zwei Randbedingungen sind üblich und führen dazu, dass (\star) Null wird:

$$\left. \begin{array}{l} \text{Dirichlet-RB: } u(a) = u(b) = 0 \\ \text{Neumann-RB: } u'(a) = u'(b) = 0 \end{array} \right\} \text{ oder } \Rightarrow [u'(x) \cdot u(x)]_a^b = 0$$

Wenn mindestens eine dieser Randbedingungen erfüllt ist, verschwindet der letzte Term, so dass sich $\mathcal{S}_{\text{diff}}$ schreiben lässt als:

$$\mathcal{S}_{\text{diff}}(u) = -\frac{1}{2} \int_a^b u''(x) \cdot u(x) dx$$

Das Ziel besteht nun darin aus $\mathcal{S}_{\text{diff}}$ den diskretisierten Regularisierer $\mathcal{S}_{\text{diff}}^h$ herzuleiten. Die Herangehensweise ist analog zu der bisher für andere Bestandteile der Zielfunktion präsentierten. Der diskretisierte Regularisierer ergibt sich aus der kontinuierlichen Form durch Auswertung auf dem CC-Gitter \bar{x} . Daher wird an Stelle von $\varphi, \bar{\varphi}$ betrachtet. Die Verrückung \bar{u} ergibt sich entsprechend.

Im Folgenden wird erst das enthaltene Integral mit bekannten Methoden und dann die enthaltene zweifache partielle Ableitung approximiert.

$$\text{mit } \gamma = \begin{cases} 3 & , \text{ Dirichlet Randbedingung} \\ 1 & , \text{ Neumann Randbedingung} \end{cases}$$

Wird dies in die bisherige Darstellung von $\mathcal{S}_{\text{diff}}$ nach Anwendung der Mittelpunkregel eingesetzt, folgt:

$$\begin{aligned} \mathcal{S}_{\text{diff}}(u) &= -\frac{1}{2} \int_a^b u(x) \cdot u''(x) dx \\ &= -\frac{h}{2} \sum_{i=1}^M u(x_i) \cdot u''(x_i) + \mathcal{O}(h^2) \\ &= -\frac{h}{2} \sum_{i=1}^M u(x_i) \cdot \frac{1}{h^2} (u(x_{i-1}) + u(x_{i+1}) - 2u(x_i)) + \mathcal{O}(h^2) \\ &= \mathcal{S}_{\text{diff}}^h(u) + \mathcal{O}(h^2) \\ \mathcal{S}_{\text{diff}}^h(u) &:= \frac{h}{2} \sum_{i=1}^M u(x_i) \cdot \frac{1}{h^2} (2u(x_i) - u(x_{i-1}) - u(x_{i+1})) \end{aligned}$$

Unter Verwendung der Darstellung mit Matrixmultiplikation, ergibt sich für den diskretisierten eindimensionalen Regularisierer:

$$\mathcal{S}_{\text{diff}}^h(\bar{u}) = \frac{1}{2h} \bar{u}^T A_M^h \bar{u}$$

Mit wenigen Erweiterungen kann die Approximation des zweidimensionalen diffusiven Regularisierers aus dem eindimensionalen Fall gewonnen werden.

Diskretisierung des zweidimensionalen Regularisierers

Im Folgenden bezeichnet u die zweidimensionale Verrückung, mit $u : \mathbb{R}^2 \rightarrow \mathbb{R}^2$ und Komponenten u_k aus $u = (u_1, u_2)^T \in \mathbb{R}^2$.

Der zweidimensionale diffusive Glätter ist definiert als:

$$\begin{aligned} \mathcal{S}_{\text{diff}}(u) &= \frac{1}{2} \int_{\Omega} \sum_{k=1}^2 \|\nabla u_k\|_2^2 dx = \frac{1}{2} \int_{\Omega} \sum_{k=1}^2 \sum_{i=1}^2 (\partial_i u_k)^2 dx \quad (3.22) \\ &= \frac{1}{2} \sum_{k=1}^2 \sum_{i=1}^2 \int_{\Omega} (\partial_i u_k)^2 dx \end{aligned}$$

Wird nur das enthaltene Integral

$$\int_{\Omega} (\partial_i u_k)^2 dx = \int_{\omega_3}^{\omega_4} \int_{\omega_1}^{\omega_2} (\partial_i u_k)^2 dx_1 dx_2 \quad (3.23)$$

betrachtet, folgt für $i = 1$ mit partieller Integration, analog zum 1D Fall (3.18) für das innere Integral:

$$\int_{\omega_1}^{\omega_2} (\partial_1 u_k)^2 dx_1 = - \int_{\omega_1}^{\omega_2} u_k \cdot (\partial_{11} u_k) dx_1 + \left[u_k \cdot (\partial_1 u_k) \right]_{x_1=\omega_1}^{x_1=\omega_2}$$

Wird die Funktion u_k als eine Funktion in x_1 aufgefasst, verschwindet das hintere Integral, wenn die eindimensionalen Dirichlet- oder Neumann-Randbedingung für die Dimension in x_1 erfüllt ist. Gilt eine der beiden folgenden Randbedingungen

$$\text{Dirichlet-RB: } u_k(\omega_1, x_2) = u_k(\omega_2, x_2) = 0, \quad \forall x_2 \in [\omega_3, \omega_4]$$

$$\text{Neumann-RB: } \partial_1 u_k(\omega_1, x_2) = \partial_1 u_k(\omega_2, x_2) = 0, \quad \forall x_2 \in [\omega_3, \omega_4]$$

dann folgt

$$\int_{\omega_1}^{\omega_2} (\partial_1 u_k)^2 dx_1 = - \int_{\omega_1}^{\omega_2} u_k \cdot (\partial_{11} u_k) dx_1$$

Analog folgt für das Integral mit $i = 2$

$$\int_{\omega_3}^{\omega_4} (\partial_2 u_k)^2 dx_2 = - \int_{\omega_3}^{\omega_4} u_k \cdot (\partial_{22} u_k) dx_2 + \left[u_k \cdot (\partial_2 u_k) \right]_{x_2=\omega_3}^{x_2=\omega_4}$$

Falls wiederum eine der beiden Randbedingungen erfüllt ist

$$\text{Dirichlet-RB: } u_k(x_1, \omega_3) = u_k(x_1, \omega_4) = 0, \quad \forall x_1 \in [\omega_1, \omega_2]$$

$$\text{Neumann-RB: } \partial_2 u_k(x_1, \omega_3) = \partial_2 u_k(x_1, \omega_4) = 0, \quad \forall x_1 \in [\omega_1, \omega_2]$$

dann folgt

$$\int_{\omega_3}^{\omega_4} (\partial_2 u_k)^2 dx_2 = - \int_{\omega_3}^{\omega_4} u_k \cdot (\partial_{22} u_k) dx_2$$

Falls u_k eine der zweidimensionalen Randbedingungen erfüllt

$$\text{Dirichlet-RB: } u_k(\omega_1, x_2) = u_k(\omega_2, x_2) = 0, \quad \forall x_2 \in [\omega_3, \omega_4] \wedge$$

$$u_k(x_1, \omega_3) = u_k(x_1, \omega_4) = 0, \quad \forall x_1 \in [\omega_1, \omega_2]$$

$$\text{Neumann-RB: } \partial_1 u_k(\omega_1, x_2) = \partial_1 u_k(\omega_2, x_2) = 0, \quad \forall x_2 \in [\omega_3, \omega_4] \wedge$$

$$\partial_2 u_k(x_1, \omega_3) = \partial_2 u_k(x_1, \omega_4) = 0, \quad \forall x_1 \in [\omega_1, \omega_2]$$

ergibt sich zusammenfassend für die Summe der Integrale aus (3.23)

$$\sum_{i=1}^2 \int_{\Omega} (\partial_i u_k)^2 dx = - \int_{\Omega} u_k \cdot (\partial_{11} u_k) dx - \int_{\Omega} u_k \cdot (\partial_{22} u_k) dx$$

$$\begin{aligned}
 &= - \int_{\Omega} u_k \cdot (\partial_{11} u_k + \partial_{22} u_k) dx \\
 &= - \int_{\Omega} u_k \cdot (\Delta u_k) dx
 \end{aligned}$$

Damit kann der diffusive Glätter aus (3.22) analog zum eindimensionalen Fall formuliert werden als:

$$S_{\text{diff}}(u) = -\frac{1}{2} \sum_{k=1}^2 \int_{\Omega} u_k \cdot (\Delta u_k) dx$$

Wie bereits im eindimensionalen Fall wird das in $S_{\text{diff}}(u)$ enthaltene Integral, sowie der Laplace-Operator approximiert, um den diskretisierten Glätter $S_{\text{diff}}^h(u)$ zu erhalten.

Unter der Voraussetzung, dass u mindestens dreimal stetig differenzierbar ist, folgt mit der zweidimensionalen Mittelpunkteregel aus (3.3) für den zweidimensionalen Regularisierer:

$$S_{\text{diff}}(u) = -\frac{h_1 h_2}{2} \sum_{k=1}^2 \sum_{i=1}^{m_1} \sum_{j=1}^{m_2} u_k(x_i, y_j) \cdot \Delta u_k(x_i, y_j) + \mathcal{O}(h_1^2 + h_2^2)$$

Analog zur eindimensionalen numerischen Differentiation, kann der Laplace-Operator Δ durch eine diskretisierte Form Δ^h approximiert werden.

$$\Delta u_k = \Delta^h u_k + \mathcal{O}(h_1^2 + h_2^2)$$

Unter Verwendung des diskretisierten Operators für die zweifache partielle Ableitung im eindimensionalen Fall ∂_{11}^h aus (3.20) folgt für den diskretisierten Laplace-Operator, ausgewertete an den CC-Gitterpunkten (x_i, y_j)

$$\begin{aligned}
 \Delta^h u_k(x_i, y_j) &= \frac{1}{h_1^2} (u_k(x_{i+1}, y_j) + u_k(x_{i-1}, y_j) - 2u_k(x_i, y_j)) + \\
 &\quad \frac{1}{h_2^2} (u_k(x_i, y_{j+1}) + u_k(x_i, y_{j-1}) - 2u_k(x_i, y_j))
 \end{aligned}$$

Der auf alle Gitterpunkte angewandte Laplace Operator kann somit wie folgt formuliert werden

$$\begin{aligned}
 \Delta^h \bar{u} &= A_{\text{diff}}^h \bar{u} \\
 A^h &= \frac{1}{h_1^2} (E_{m_2} \otimes A_{m_1}^h) + \frac{1}{h_2^2} (A_{m_2}^h \otimes E_{m_1}) \\
 A_{\text{diff}}^h &:= \begin{pmatrix} A^h & \\ & A^h \end{pmatrix} \in \mathbb{R}^{2m_1 m_2 \times 2m_1 m_2}
 \end{aligned}$$

Beide Approximationen werden nun verwendet, um den diskretisierten, zweidimensionalen Regularisierer zu erhalten:

$$\begin{aligned}
 S_{\text{diff}}^h(u) &= \frac{h_1 h_2}{2} \sum_{k=1}^2 \sum_{i=1}^{m_1} \sum_{j=1}^{m_2} u_k(x_i, y_j) \cdot \Delta^h u_k(x_i, y_j) \\
 &= \frac{h_1 h_2}{2} \bar{u}^T A_{\text{diff}}^h \bar{u}
 \end{aligned}$$

Curvature Glätter

Analog zum diffusiven Regularisierer kann der Curvature Regularisierer mit den bereits bekannten Methoden diskretisiert werden.

$$\begin{aligned} \mathcal{S}_{\text{curv}}(u) &= \frac{1}{2} \sum_{k=1}^2 \int_{\Omega} (\Delta u_k(x))^2 dx \\ &= \frac{h_1 h_2}{2} \sum_{k=1}^2 \sum_{i=1}^{m_1} \sum_{j=1}^{m_2} \left(\Delta^h u_k(x_i, y_j) \right)^2 + \mathcal{O}(h_1^2 + h_2^2) \end{aligned}$$

Daher definieren wir den diskretisierten Curvature Regularisierer als

$$\begin{aligned} \mathcal{S}_{\text{curv}}^h(\bar{u}) &:= \frac{h_1 h_2}{2} \bar{u}^T \underbrace{\left(A_{\text{diff}}^h \right)^T A_{\text{diff}}^h}_{=A_{\text{curv}}^h} \bar{u} \\ \mathcal{S}_{\text{curv}}^h(\bar{u}) &:= \frac{h_1 h_2}{2} \bar{u}^T A_{\text{curv}}^h \bar{u} \end{aligned}$$

Wie gezeigt wurde, können beide Regularisierer in einer quadratischen Form wie folgt dargestellt werden:

$$\mathcal{S}^h(\bar{u}) := \frac{h_1 h_2}{2} \bar{u}^T A^h \bar{u} \quad (3.24)$$

Da die Matrix A^h symmetrisch ist, ergibt sich die Ableitung des diskretisierten Regularisierers zu

$$\nabla \mathcal{S}^h(\bar{u}) := h_1 h_2 A^h \bar{u} \quad (3.25)$$

Im Folgenden wird der diskretisierte Regularisierer immer mit \mathcal{S}^h bezeichnet.

3.2.5 Die diskretisierte nichtparametrische Zielfunktion und ihre Ableitung

Die nichtparametrische Zielfunktion lautet

$$\mathcal{J}_{\text{np}}(\varphi) = \mathcal{D}((J_\varphi)^{-1}(T \circ \varphi), R) + \alpha \mathcal{S}(u)$$

Ihre diskretisierte Repräsentation ergibt sich aus den diskretisierten Bestandteilen wie folgt:

$$\mathcal{J}_{\text{np}}^h(\bar{\varphi}) = \mathcal{D}^h(\bar{B}(\bar{\varphi}) \cdot \bar{T}(\bar{\varphi}), \bar{R}) + \alpha \mathcal{S}^h(\bar{u})$$

Mit der Darstellung des Distanzmaßes aus (3.17) und dem Regularisierer aus (3.24) kann die Zielfunktion beschrieben werden als

$$\mathcal{J}_{\text{np}}^h(\bar{\varphi}) = \frac{h_1 h_2}{2} r(\bar{\varphi})^T r(\bar{\varphi}) + \alpha \frac{h_1 h_2}{2} \bar{u}^T A^h \bar{u}$$

Die Ableitung dieser Funktion nach ihrem Optimierungsparameter φ ergibt sich aus den differenzierten Teilfunktionen wie folgt:

$$\nabla \mathcal{J}_{\text{np}}^h(\bar{\varphi}) = \nabla \mathcal{D}^h(\bar{B}(\bar{\varphi}) \cdot \bar{T}(\bar{\varphi}), \bar{R}) + \alpha \nabla \mathcal{S}^h(u)$$

Die Ableitung des diskretisierten Regularisierers

$$\mathcal{S}^h(\bar{u}) = \frac{h_1 h_2}{2} \bar{u}^T A^h \bar{u}$$

ist bereits aus (3.25) bekannt und lautet:

$$\nabla \mathcal{S}^h(\bar{u}) = h_1 h_2 A^h \bar{u}$$

Die Ableitung der Distanzfunktion

$$\mathcal{D}^h(\bar{B}(\bar{\varphi}) \cdot \bar{T}(\bar{\varphi}), \bar{R}) = \frac{h_1 h_2}{2} r(\bar{\varphi})^T r(\bar{\varphi})$$

lautet

$$\nabla \mathcal{D}^h(\bar{B}(\bar{\varphi}) \cdot \bar{T}(\bar{\varphi}), \bar{R}) = h_1 h_2 \nabla r(\bar{\varphi}) \cdot r(\bar{\varphi}) \in \mathbb{R}^{2m_1 m_2}$$

Die enthaltene Ableitung der Hilfsfunktion r

$$r(\bar{\varphi}) = \begin{pmatrix} b_{11}(J(\bar{\varphi})) \odot \bar{T}_1(\bar{\varphi}) + b_{12}(J(\bar{\varphi})) \odot \bar{T}_2(\bar{\varphi}) - \bar{R}_1 \\ b_{21}(J(\bar{\varphi})) \odot \bar{T}_1(\bar{\varphi}) + b_{22}(J(\bar{\varphi})) \odot \bar{T}_2(\bar{\varphi}) - \bar{R}_2 \end{pmatrix}$$

ergibt sich durch elementweise Differentiation entsprechend zu

$$\nabla r(\bar{\varphi}) = \left(\sum_{k=1}^2 \nabla(b_{1k}(J(\bar{\varphi})) \odot \bar{T}_k(\bar{\varphi})), \sum_{k=1}^2 \nabla(b_{2k}(J(\bar{\varphi})) \odot \bar{T}_k(\bar{\varphi})) \right) \in \mathbb{R}^{2m_1 m_2 \times 2m_1 m_2}$$

Die Ableitung der Summanden jeder Komponente ergibt sich durch Ableitung der enthaltenen Funktionen b aus (3.15) und J aus (3.13) wie folgt:

$$\begin{aligned} \nabla(b_{ik}(J(\bar{\varphi})) \odot \bar{T}_k(\bar{\varphi})) &= \nabla_{\varphi}(b_{ik}(J(\bar{\varphi})) \odot \bar{T}_k(\bar{\varphi})) \in \mathbb{R}^{2m_1 m_2 \times m_1 m_2} \\ &= \nabla J(\bar{\varphi}) \cdot \nabla b_{ik}(J(\bar{\varphi})) \cdot \text{diag}(\bar{T}_k(\bar{\varphi})) + \nabla \bar{T}_k(\bar{\varphi}) \cdot \text{diag}(b_{ik}(J(\bar{\varphi}))) \end{aligned}$$

jeweils mit $\nabla b_{ij} \in \mathbb{R}^{4m_1 m_2 \times m_1 m_2}$.

$$\begin{aligned} \bar{c}(J) &:= \frac{1}{(J_{11} \odot J_{22} - J_{12} \odot J_{21})} \\ \nabla b_{11} &= \begin{pmatrix} \text{diag}(-\bar{c}(J)^2 \odot (J_{22})^2) \\ \text{diag}(\bar{c}(J)^2 \odot J_{12} \odot J_{22}) \\ \text{diag}(\bar{c}(J)^2 \odot J_{21} \odot J_{22}) \\ \text{diag}(-\bar{c}(J)^2 \odot J_{11} + \bar{c}(J)) \end{pmatrix} \\ \nabla b_{21} &= \begin{pmatrix} \text{diag}(\bar{c}(J)^2 \odot J_{22} \odot J_{21}) \\ \text{diag}(\bar{c}(J)^2 \odot J_{12} \odot J_{21} - \bar{c}(J)) \\ \text{diag}(-\bar{c}(J)^2 \odot (J_{21})^2) \\ \text{diag}(\bar{c}(J)^2 \odot J_{11} \odot J_{21}) \end{pmatrix} \end{aligned}$$

$$\nabla b_{12} = \begin{pmatrix} \text{diag}(\bar{c}(J)^2 \odot J_{22} \odot J_{12}) \\ \text{diag}(-\bar{c}(J)^2 \odot (J_{12})^2) \\ \text{diag}(-\bar{c}(J)^2 \odot J_{21} \odot J_{12} - \bar{c}(J)) \\ \text{diag}(\bar{c}(J)^2 \odot J_{11} \odot J_{12}) \end{pmatrix}$$

$$\nabla b_{22} = \begin{pmatrix} \text{diag}(-\bar{c}(J)^2 \odot J_{22} \odot J_{11} + \bar{c}(J)) \\ \text{diag}(\bar{c}(J)^2 \odot J_{12} \odot J_{11}) \\ \text{diag}(\bar{c}(J)^2 \odot J_{21} \odot J_{11}) \\ \text{diag}(-\bar{c}(J)^2 \odot (J_{11})^2) \end{pmatrix}$$

wobei die Potenzen elementweise gemeint sind.

Die Ableitung der Funktion zur Approximation der Jacobimatrix aus (3.13)

$$J(\bar{\varphi}) = D \cdot \bar{\varphi}$$

lautet entsprechend

$$\nabla J(\bar{\varphi}) = D^T \in \mathbb{R}^{2m_1 m_2 \times 4m_1 m_2}$$

Wie bereits im letzten Kapitel erwähnt, sind die Ableitungen der Komponenten des Flussbildes $\nabla \bar{T}_k$ bedingt durch die gewählte Bildfunktion aus $\mathcal{I}_{2,2}$ und müssen für den jeweiligen Fall bestimmt werden.

3.3 Nichtparametrische Zielfunktion mit Basistransformation als Parameter

Die nichtparametrische Zielfunktion mit Basistransformation als zusätzlichen Optimierungsparameter lautet

$$\mathcal{J}_{\text{npb}}(\varphi, B) = \mathcal{D}(B \cdot (T \circ \varphi), R) + \alpha \mathcal{S}(u) + \beta \mathcal{P}(B, \varphi)$$

In den vorangegangenen Kapiteln wurde bereits der Regularisierer \mathcal{S} und das enthaltene Distanzmaß \mathcal{D} diskretisiert. Eine Anpassung ist noch hinsichtlich der im Argument enthaltenen Basistransformation notwendig.

3.3.1 Basistransformation

Die Basistransformation als Optimierungsparameter wird als eine Funktion aufgefasst, die jedem Datenpunkt eine Matrix zuordnet. Entsprechend weist die Abbildung B jedem Gitterpunkt vier Werte zu, die zur Reorientierung im zweidimensionalen Fall notwendig sind.

$$\bar{B}(\bar{x}) = \begin{pmatrix} B_{11}(\bar{x}) \\ B_{21}(\bar{x}) \\ B_{12}(\bar{x}) \\ B_{22}(\bar{x}) \end{pmatrix} \in \mathbb{R}^{4m_1 m_2}, \quad B_{ij} : \mathbb{R}^{2m_1 m_2} \rightarrow \mathbb{R}^{m_1 m_2} \quad (3.26)$$

Die Teilfunktionen B_{ij} repräsentieren dabei jeweils eine Komponente der 2×2 Basistransformation B für einen Datenpunkt. Das basistransformierte, deformierte Template kann unter Verwendung dieser Funktion als eine Vektormultiplikation dargestellt werden:

$$\begin{pmatrix} B_{11} \odot \bar{T}_1(\bar{\varphi}) + B_{12} \odot \bar{T}_2(\bar{\varphi}) \\ B_{21} \odot \bar{T}_1(\bar{\varphi}) + B_{22} \odot \bar{T}_2(\bar{\varphi}) \end{pmatrix}$$

Das diskretisierte Distanzmaß mit dem passenden Argument ergibt sich zu

$$D^h \left(\begin{pmatrix} B_{11} \odot \bar{T}_1(\bar{\varphi}) + B_{12} \odot \bar{T}_2(\bar{\varphi}) \\ B_{21} \odot \bar{T}_1(\bar{\varphi}) + B_{22} \odot \bar{T}_2(\bar{\varphi}) \end{pmatrix}, \bar{R} \right)$$

Wie bereits bei den Zielfunktionen \mathcal{J}_p und \mathcal{J}_{np} , kann das diskretisierte Distanzmaß unter Verwendung einer Hilfsfunktion r formuliert werden

$$r(\bar{\varphi}, \bar{B}) = \begin{pmatrix} B_{11}(\bar{x}) \odot \bar{T}_1(\bar{\varphi}) + B_{12}(\bar{x}) \odot \bar{T}_2(\bar{\varphi}) - \bar{R}_1 \\ B_{21}(\bar{x}) \odot \bar{T}_1(\bar{\varphi}) + B_{22}(\bar{x}) \odot \bar{T}_2(\bar{\varphi}) - \bar{R}_2 \end{pmatrix}$$

Das in der Zielfunktion \mathcal{J}_{npb} enthaltene diskretisierte SSD-Maß lautet entsprechend

$$D^h \left(\begin{pmatrix} B_{11} \odot \bar{T}_1(\bar{\varphi}) + B_{12} \odot \bar{T}_2(\bar{\varphi}) \\ B_{21} \odot \bar{T}_1(\bar{\varphi}) + B_{22} \odot \bar{T}_2(\bar{\varphi}) \end{pmatrix}, \bar{R} \right) = \frac{h_1 h_2}{2} r(\bar{\varphi}, \bar{B})^T \cdot r(\bar{\varphi}, \bar{B}) \quad (3.27)$$

Diese Darstellung wird später die Bildung der Ableitung erleichtern.

3.3.2 Strafterm

Der Strafterm aus (2.23) lautet

$$\mathcal{P}(\varphi, B) = \int_{\Omega} \|B(x) \cdot J_{\varphi}(x) - E_2\|_F^2 dx$$

Der Strafterm wird im Folgenden in drei Schritten diskretisiert:

Zu Beginn wird die in der Matrixdifferenz enthaltene Jacobimatrix der Deformation φ mit der Funktion J aus (3.13) durch finite Differenzen approximiert.

Anschließend wird die Frobeniusnorm in eine Vektornorm überführt und am Ende wird zur Approximation des Integrals die Mittelpunkregel wie in den vorangegangenen Kapiteln verwendet.

Approximation der Jacobimatrix

Die Basistransformationsmatrix für den Punkt $x \in \mathbb{R}^2$ lautet

$$B(x) = \begin{pmatrix} B_{11}(x) & B_{12}(x) \\ B_{21}(x) & B_{22}(x) \end{pmatrix} \in \mathbb{R}^{2 \times 2}$$

Die approximierte Jacobimatrix aus (3.11) ausgewertet an einem Datenpunkt lautet

$$J_{\varphi}(x) \approx J_{\varphi}^h(x) := \begin{pmatrix} \partial_{x_1}^h \varphi_1(x) & \partial_{x_2}^h \varphi_1(x) \\ \partial_{x_1}^h \varphi_2(x) & \partial_{x_2}^h \varphi_2(x) \end{pmatrix}$$

Zur Vereinfachung der Schreibweise wird im Folgenden auf den Index φ verzichtet

$$J^h(x) = \begin{pmatrix} J_{11}^h(x) & J_{12}^h(x) \\ J_{21}^h(x) & J_{22}^h(x) \end{pmatrix} = J_\varphi^h(x) \in \mathbb{R}^{2 \times 2}$$

Die in der Frobeniusnorm enthaltene Matrixdifferenz ergibt sich unter Verwendung der Approximation zu

$$\begin{aligned} B(x) \cdot J_\varphi(x) - E_2 &\approx B(x) \cdot J^h(x) - E_2 \\ &= \begin{pmatrix} B_{11}(x)J_{11}^h(x) + B_{12}(x)J_{21}^h(x) - 1 & B_{11}(x)J_{21}^h(x) + B_{12}(x)J_{22}^h(x) \\ B_{21}(x)J_{11}^h(x) + B_{22}(x)J_{21}^h(x) & B_{21}(x)J_{12}^h(x) + B_{22}(x)J_{22}^h(x) - 1 \end{pmatrix} \end{aligned}$$

Die Einträge dieser Matrix werden spaltweise in einem Vektor untereinander geschrieben und in einer Hilfsfunktion p wie folgt zusammengefasst:

$$p(J^h, B)(x) = \begin{pmatrix} B_{11}(x)J_{11}^h(x) + B_{12}(x)J_{21}^h(x) - 1 \\ B_{21}(x)J_{11}^h(x) + B_{22}(x)J_{21}^h(x) \\ B_{11}(x)J_{21}^h(x) + B_{12}(x)J_{22}^h(x) \\ B_{21}(x)J_{12}^h(x) + B_{22}(x)J_{22}^h(x) - 1 \end{pmatrix} := \begin{pmatrix} p_1(x) \\ p_2(x) \\ p_3(x) \\ p_4(x) \end{pmatrix} \in \mathbb{R}^4$$

Mit dieser Funktion kann die Matrixnorm umformuliert werden.

Ersetzen der Frobeniusnorm

Die Frobeniusnorm im Strafterm kann mit der Funktion p durch eine euklidische Vektornorm ersetzt werden:

$$\begin{aligned} \|B(x) \cdot J_\varphi(x) - E_2\|_F^2 &\approx \|B(x) \cdot J^h(x) - E_2\|_F^2 \\ &= \left\| \begin{pmatrix} B_{11}(x)J_{11}^h(x) + B_{12}(x)J_{21}^h(x) - 1 \\ B_{21}(x)J_{11}^h(x) + B_{22}(x)J_{21}^h(x) \\ B_{11}(x)J_{21}^h(x) + B_{12}(x)J_{22}^h(x) \\ B_{21}(x)J_{12}^h(x) + B_{22}(x)J_{22}^h(x) - 1 \end{pmatrix} \right\|_2^2 \\ &= \left(p(J^h, B)(x) \right)^T \cdot \left(p(J^h, B)(x) \right) \\ &= \sum_{k=1}^4 (p_k(x))^2 \end{aligned}$$

Approximation des Integrals mit der Mittelpunkregel

Das Ersetzen der Frobeniusnorm durch ihre Approximation ergibt als Näherung des Integrals über den Bildbereich Ω

$$\int_{\Omega} \|B(x) \cdot J_\varphi(x) - E_2\|_F^2 dx \approx \int_{\Omega} \sum_{k=1}^4 (p_k(x))^2 dx$$

Unter der Voraussetzung, dass die $p_k(x)$ mindestens zweifach stetig differenzierbare Funktionen sind, kann das Integral mit der Mittelpunkregel an den CC-Gitterpunkten (x_i, y_j) angegeben werden als

$$\int_{\Omega} \sum_{k=1}^4 (p_k(x))^2 dx = h_1 h_2 \cdot \sum_{j=1}^{m_2} \sum_{i=1}^{m_1} \sum_{k=1}^4 (p_k(x_i, y_j))^2 + \mathcal{O}(h_1^2 + h_2^2)$$

Bei Vernachlässigung des Strafterms folgt die Approximation des Integrals

$$\int_{\Omega} \sum_{k=1}^4 (p_k(x))^2 dx \approx h_1 h_2 \cdot \sum_{j=1}^{m_2} \sum_{i=1}^{m_1} \sum_{k=1}^4 (p_k(x_i, y_j))^2$$

Als Approximation des Fehlerterms wird daher folgender Term gewählt:

$$\mathcal{P}^h(\varphi, B) = \int_{\Omega} \|B(x) \cdot J_{\varphi}(x) - E_2\|_F^2 dx \approx h_1 h_2 \cdot \sum_{k=1}^4 \sum_{j=1}^{m_2} \sum_{i=1}^{m_1} (p_k(x_i, y_j))^2$$

Vektorschreibweise

Die Vektorschreibweise für die diskretisierte Deformation wird konstruiert, indem die Teilfunktionen p_k an allen Gitterpunkten ausgewertet und anschließend konkateniert werden. Dazu wird die zu Beginn vorgestellte Funktion \bar{B} aus (3.26) verwendet, welche die Basistransformation für alle Datenpunkte liefert. Außerdem werden die Teilfunktionen J_{ij} zur Schätzung der Komponenten der Jacobimatrizen aus (3.13) angewandt. Die Hilfsfunktion p wird zur vektorwertigen Hilfsfunktion wie folgt

$$p(\bar{\varphi}, \bar{B})(\bar{x}) = \begin{pmatrix} \bar{p}_1(\bar{\varphi}, \bar{B})(\bar{x}) \\ \bar{p}_2(\bar{\varphi}, \bar{B})(\bar{x}) \\ \bar{p}_3(\bar{\varphi}, \bar{B})(\bar{x}) \\ \bar{p}_4(\bar{\varphi}, \bar{B})(\bar{x}) \end{pmatrix} \in \mathbb{R}^{4m_1 m_2}$$

mit

$$\begin{aligned} \bar{p}_1(\bar{\varphi}, \bar{B})(\bar{x}) &= B_{11}(\bar{x}) \odot J_{11}(\bar{\varphi}) + B_{12}(\bar{x}) \odot J_{21}(\bar{\varphi}) - 1 \\ \bar{p}_2(\bar{\varphi}, \bar{B})(\bar{x}) &= B_{21}(\bar{x}) \odot J_{11}(\bar{\varphi}) + B_{22}(\bar{x}) \odot J_{21}(\bar{\varphi}) \\ \bar{p}_3(\bar{\varphi}, \bar{B})(\bar{x}) &= B_{11}(\bar{x}) \odot J_{12}(\bar{\varphi}) + B_{12}(\bar{x}) \odot J_{22}(\bar{\varphi}) \\ \bar{p}_4(\bar{\varphi}, \bar{B})(\bar{x}) &= B_{21}(\bar{x}) \odot J_{12}(\bar{\varphi}) + B_{22}(\bar{x}) \odot J_{22}(\bar{\varphi}) - 1 \end{aligned}$$

Angewandt auf den Strafterm

$$\mathcal{P}(\varphi, B) = \int_{\Omega} \|B(x) \cdot J_{\varphi}(x) - E_2\|_F^2 dx$$

folgt der diskretisierte Strafterm $\mathcal{P}^h : \mathbb{R}^{6m_1 m_2} \rightarrow \mathbb{R}$

$$\mathcal{P}^h(\bar{\varphi}, \bar{B}) = h_1 h_2 (p(\bar{\varphi}, \bar{B}))^T \cdot p(\bar{\varphi}, \bar{B}) \quad (3.28)$$

Die vollständig diskretisierte Zielfunktion kann nun aus den diskretisierten Bestandteilen zusammengesetzt werden.

3.3.3 Die diskretisierte Zielfunktion und ihre Ableitung

Die Zielfunktion lautet bisher:

$$\mathcal{J}_{\text{npb}}(\varphi, B) = \mathcal{D}(B \cdot (T \circ \varphi), R) + \alpha \mathcal{S}(u) + \beta \mathcal{P}(\varphi, B)$$

Die diskretisierte Zielfunktion ergibt sich aus den diskretisierten Bestandteilen zu

$$\mathcal{J}_{\text{npb}}^h(\bar{\varphi}, \bar{B}) = D^h \left(\begin{pmatrix} B_{11} \odot \bar{T}_1(\bar{\varphi}) + B_{12} \odot \bar{T}_2(\bar{\varphi}) \\ B_{21} \odot \bar{T}_1(\bar{\varphi}) + B_{22} \odot \bar{T}_2(\bar{\varphi}) \end{pmatrix}, \bar{R} \right) + \alpha \mathcal{S}^h(\bar{u}) + \beta \mathcal{P}^h(\bar{\varphi}, \bar{B})$$

mit den entsprechenden Definitionen aus (3.27), (3.24) und (3.28) folgt

$$\mathcal{J}_{\text{npb}}^h(\bar{\varphi}, \bar{B}) = \frac{h_1 h_2}{2} r(\bar{\varphi}, \bar{B})^T \cdot r(\bar{\varphi}, \bar{B}) + \alpha \frac{h_1 h_2}{2} \bar{u}^T A^h \bar{u} + \beta h_1 h_2 (p(\bar{\varphi}, \bar{B}))^T \cdot p(\bar{\varphi}, \bar{B})$$

Die Ableitung ergibt sich erneut als Summe der Ableitungen der Summanden. Diesmal muss beachtet werden, dass neben der Bestimmung der Ableitung entsprechend φ , zusätzlich nach dem neuen Optimierungsparameter B differenziert werden muss, um die vollständige Ableitung zu erhalten.

$$\nabla \mathcal{J}_{\text{npb}}^h(\bar{\varphi}, \bar{B}) = \nabla D^h \left(\begin{pmatrix} B_{11} \odot \bar{T}_1(\bar{\varphi}) + B_{12} \odot \bar{T}_2(\bar{\varphi}) \\ B_{21} \odot \bar{T}_1(\bar{\varphi}) + B_{22} \odot \bar{T}_2(\bar{\varphi}) \end{pmatrix}, \bar{R} \right) + \alpha \nabla \mathcal{S}^h(\bar{u}) + \beta \nabla \mathcal{P}^h(\bar{\varphi}, \bar{B})$$

Im Folgenden werden die Ableitungen der Summanden einzeln hergeleitet.

Ableitung des Distanzmaßes

Das diskretisierte Distanzmaß aus (3.27) lautet:

$$D^h \left(\begin{pmatrix} B_{11} \odot \bar{T}_1(\bar{\varphi}) + B_{12} \odot \bar{T}_2(\bar{\varphi}) \\ B_{21} \odot \bar{T}_1(\bar{\varphi}) + B_{22} \odot \bar{T}_2(\bar{\varphi}) \end{pmatrix}, \bar{R} \right) = \frac{h_1 h_2}{2} r(\bar{\varphi}, \bar{B})^T \cdot r(\bar{\varphi}, \bar{B})$$

Die Verwendung der Hilfsfunktion r erlaubt eine kompakte Darstellung der zugehörigen Ableitung wie folgt:

$$\nabla D^h \left(\begin{pmatrix} B_{11} \odot \bar{T}_1(\bar{\varphi}) + B_{12} \odot \bar{T}_2(\bar{\varphi}) \\ B_{21} \odot \bar{T}_1(\bar{\varphi}) + B_{22} \odot \bar{T}_2(\bar{\varphi}) \end{pmatrix}, \bar{R} \right) = h_1 h_2 \nabla r(\bar{\varphi}, \bar{B})^T \cdot r(\bar{\varphi}, \bar{B})$$

Die Ableitung der Hilfsfunktion $\nabla r(\bar{\varphi}, \bar{B})$ ergibt sich aus den Ableitungen der Teilfunktionen.

Die Elemente von

$$\begin{aligned} r(\bar{\varphi}, \bar{B})(\bar{x}) &= \begin{pmatrix} B_{11}(\bar{x}) \odot \bar{T}_1(\bar{\varphi}) + B_{12}(\bar{x}) \odot \bar{T}_2(\bar{\varphi}) - \bar{R}_1 \\ B_{21}(\bar{x}) \odot \bar{T}_1(\bar{\varphi}) + B_{22}(\bar{x}) \odot \bar{T}_2(\bar{\varphi}) - \bar{R}_2 \end{pmatrix} \\ &= \begin{pmatrix} \sum_{k=1}^2 B_{1k}(\bar{x}) \odot \bar{T}_k(\bar{\varphi}) - \bar{R}_1 \\ \sum_{k=1}^2 B_{2k}(\bar{x}) \odot \bar{T}_k(\bar{\varphi}) - \bar{R}_2 \end{pmatrix} \end{aligned}$$

werden einzeln differenziert. Die Ableitung nach φ und B lauten jeweils

$$\nabla_{\varphi} (B_{ij}(\bar{x}) \odot \bar{T}_j(\bar{\varphi}) - \bar{R}_i) = \nabla \bar{T}_j(\bar{\varphi}) \cdot \text{diag}(\bar{T}_j(\bar{\varphi}) \odot B_{ij}(\bar{x})) \in \mathbb{R}^{2m_1 m_2 \times m_1 m_2}$$

$$\nabla_{B_{kl}} (B_{ij}(\bar{x}) \odot \bar{T}_j(\bar{\varphi}) - \bar{R}_i) = \begin{cases} \text{diag}(\bar{T}_j(\bar{\varphi})) & , (k, l) = (i, j) \\ Z_{m_1 m_2} & , \text{sonst} \end{cases}$$

Für die Ableitungen von r folgt

$$\begin{aligned} \nabla_{\varphi} r(\bar{\varphi}, B) &= \left(\sum_{k=1}^2 \nabla_{\varphi} (B_{1k}(\bar{x}) \odot \bar{T}_k(\bar{\varphi})) , \sum_{k=1}^2 \nabla_{\varphi} (B_{2k}(\bar{x}) \odot \bar{T}_k(\bar{\varphi})) \right) \\ \nabla_{B_{i1}} r(\bar{\varphi}, B) &= \left(\text{diag}(\bar{T}_1(\bar{\varphi})) \quad Z_{m_1 m_2} \right) \in \mathbb{R}^{m_1 m_2 \times 2 m_1 m_2} \\ \nabla_{B_{i2}} r(\bar{\varphi}, B) &= \left(\quad Z_{m_1 m_2} \quad \text{diag}(\bar{T}_2(\bar{\varphi})) \right) \\ \nabla_B r(\bar{\varphi}, B) &= \begin{pmatrix} \nabla_{B_{11}} r(\bar{\varphi}, B) \\ \nabla_{B_{21}} r(\bar{\varphi}, B) \\ \nabla_{B_{12}} r(\bar{\varphi}, B) \\ \nabla_{B_{22}} r(\bar{\varphi}, B) \end{pmatrix} \end{aligned}$$

Werden die Ableitungen zusammengeführt, ergibt sich:

$$\nabla r(\bar{\varphi}, \bar{B}) = \begin{pmatrix} \nabla_{\varphi} r(\bar{\varphi}, \bar{B}) \\ \nabla_B r(\bar{\varphi}, \bar{B}) \end{pmatrix} \in \mathbb{R}^{6 m_1 m_2 \times 2 m_1 m_2}$$

Ableitung des Regularisierers

Die Ableitung des Regularisierers nach φ ist bereits aus (3.25) bekannt und lautet:

$$\nabla_{\varphi} \mathcal{S}^h(\bar{u}) = h_1 h_2 A^h \bar{u}$$

Da die Basistransformation \bar{B} nicht im Regularisierer enthalten ist, verschwindet die Ableitung für alle Teilfunktionen B_{ij}

$$\nabla_{B_{ij}} \mathcal{S}^h(\bar{u}) = z_{m_1 m_2} = 0$$

Die zusammengesetzte Ableitung ergibt sich zu

$$\nabla \mathcal{S}^h(\bar{u}) = \begin{pmatrix} \nabla_{\varphi} \mathcal{S}^h(\bar{u}) \\ \nabla_{B_{11}} \mathcal{S}^h(\bar{u}) \\ \nabla_{B_{21}} \mathcal{S}^h(\bar{u}) \\ \nabla_{B_{12}} \mathcal{S}^h(\bar{u}) \\ \nabla_{B_{22}} \mathcal{S}^h(\bar{u}) \end{pmatrix} = \begin{pmatrix} h_1 h_2 A^h \bar{u} \\ z_{m_1 m_2} \\ z_{m_1 m_2} \\ z_{m_1 m_2} \\ z_{m_1 m_2} \end{pmatrix} \in \mathbb{R}^{6 m_1 m_2}$$

Ableitung des Strafterms

Der diskretisierte Strafterm lautet:

$$\mathcal{P}^h(\bar{\varphi}, \bar{B}) = h_1 h_2 (p(\bar{\varphi}, \bar{B}))^T \cdot p(\bar{\varphi}, \bar{B})$$

Analog zur Vorgehensweise bei der Bestimmung der Ableitung des Distanzmaßes, kann die zugehörige Ableitung unter Verwendung der Hilfsfunktion wie folgt angegeben werden:

$$\nabla \mathcal{P}^h(\bar{\varphi}, \bar{B}) = 2 h_1 h_2 \nabla (p(\bar{\varphi}, \bar{B})) \cdot p(\bar{\varphi}, \bar{B})$$

Die Ableitung $\nabla (p(\bar{\varphi}, \bar{B}))$ ergibt sich aus den differenzierten Teilfunktionen \bar{p}_k .

Mit den Teilfunktionen der Basistransformation B_{ij} aus (3.26) und den Funktionen zur Schätzung der Jacobimatrix von φJ_{ij} aus (3.13) folgt

$$\begin{aligned} \nabla_{\varphi} p_1(\bar{\varphi}, \bar{B}) &= \nabla J_{11}(\bar{\varphi}) \cdot \text{diag}(B_{11}) + \nabla J_{21}(\bar{\varphi}) \cdot \text{diag}(B_{12}) \in \mathbb{R}^{2m_1 m_2 \times m_1 m_2} \\ \nabla_{\varphi} p_2(\bar{\varphi}, \bar{B}) &= \nabla J_{11}(\bar{\varphi}) \cdot \text{diag}(B_{21}) + \nabla J_{21}(\bar{\varphi}) \cdot \text{diag}(B_{22}) \\ \nabla_{\varphi} p_3(\bar{\varphi}, \bar{B}) &= \nabla J_{12}(\bar{\varphi}) \cdot \text{diag}(B_{11}) + \nabla J_{22}(\bar{\varphi}) \cdot \text{diag}(B_{12}) \\ \nabla_{\varphi} p_4(\bar{\varphi}, \bar{B}) &= \nabla J_{12}(\bar{\varphi}) \cdot \text{diag}(B_{21}) + \nabla J_{22}(\bar{\varphi}) \cdot \text{diag}(B_{22}) \\ \Rightarrow \nabla_{\varphi} p(\bar{\varphi}, \bar{B}) &= (\nabla_{\varphi} p_1(\bar{\varphi}, \bar{B}) \quad \nabla_{\varphi} p_2(\bar{\varphi}, \bar{B}) \quad \nabla_{\varphi} p_3(\bar{\varphi}, \bar{B}) \quad \nabla_{\varphi} p_4(\bar{\varphi}, \bar{B})) \in \mathbb{R}^{2m_1 m_2 \times 4m_1 m_2} \end{aligned}$$

Mit der Nullmatrix $Z := Z_{m_1 m_2}$ lauten die Ableitungen der J_{ij} nach φ

$$\begin{aligned} \nabla J_{11} &= (D_x \quad Z)^T \\ \nabla J_{21} &= (Z \quad D_x)^T \\ \nabla J_{12} &= (D_y \quad Z)^T \\ \nabla J_{22} &= (Z \quad D_y)^T \end{aligned}$$

Die Ableitung von p nach der Basistransformation B lautet

$$\nabla_B p(\bar{\varphi}, \bar{B}) = \begin{pmatrix} \text{diag}(J_{11}(\bar{\varphi})) & Z & \text{diag}(J_{12}(\bar{\varphi})) & Z \\ Z & \text{diag}(J_{11}(\bar{\varphi})) & Z & \text{diag}(J_{12}(\bar{\varphi})) \\ \text{diag}(J_{21}(\bar{\varphi})) & Z & \text{diag}(J_{22}(\bar{\varphi})) & Z \\ Z & \text{diag}(J_{21}(\bar{\varphi})) & Z & \text{diag}(J_{22}(\bar{\varphi})) \end{pmatrix} \in \mathbb{R}^{4m_1 m_2 \times 4m_1 m_2}$$

Durch Konkatenation der einzelnen Ableitungen ergibt sich die Ableitung zu

$$\nabla p(\bar{\varphi}, \bar{B}) = \begin{pmatrix} \nabla_{\varphi} p(\bar{\varphi}, \bar{B}) \\ \nabla_B p(\bar{\varphi}, \bar{B}) \end{pmatrix} \in \mathbb{R}^{6m_1 m_2 \times 4m_1 m_2}$$

Die Ableitung des Strafterms ist durch die Ableitung der Hilfsfunktion vollständig beschrieben wie folgt

$$\nabla \mathcal{P}^h(\bar{\varphi}, \bar{B}) = 2 (\nabla p(\bar{\varphi}, \bar{B})) p(\bar{\varphi}, \bar{B}) \in \mathbb{R}^{6m_1 m_2}$$

Die Ableitung der Zielfunktion \mathcal{J}_{npb} ist durch die Ableitungen der Summanden voll bestimmt und lautet

$$\nabla \mathcal{J}_{\text{npb}}^h(\bar{\varphi}, B) = \nabla \mathcal{D}^h(\bar{\varphi}) + \alpha \nabla \mathcal{S}^h(\bar{u}) + \beta \nabla \mathcal{P}^h(\bar{\varphi}, \bar{B})$$

Die Beschreibung der Zielfunktionen ist unvollständig, da bisher noch keine Bildfunktion festgelegt wurde. Die bisherige Forderung, dass es sich dabei um eine stetig, differenzierbare Funktion handeln soll, wird im Folgenden spezifiziert.

3.4 Interpolation

Bilder sind diskrete Darstellungen von Ausschnitten der stetigen Realität, die durch punktuelle Messungen von bestimmten Sensoren verschiedenster Art entstehen.

Für die Deformation von Bildern wurde die stetige Abbildung φ eingeführt, welche einen Ort an einen anderen Ort transformiert. Da Bilder auf dem entstehenden deformierten Bildbereich ausgewertet werden müssen, wurden skalar- und vektorwertige Bilder im vorangegangenen Kapitel 2 als eine kontinuierliche Funktion definiert.

Im Gegensatz dazu, führen die üblichen Messverfahren, auch in der medizinischen Bildgebung, zu digitalen Bildern, die aus einer begrenzten Anzahl von Pixel oder Voxel bestehen und somit diskret sind. Um aus diesen vorliegenden diskreten Messungen stetige Abbildungen zu gewinnen, wurden sie mit kubische B-Splines interpoliert. Diese können dann an beliebigen Stellen ausgewertet werden, so dass eine Auswertung auf einem regelmäßigem Gitter ebenfalls möglich ist.

Zudem handelt es sich hierbei um Polynome dritten Grades, so dass die Bestimmung der Ableitung problemlos erfolgen kann.

Nachdem alle Zielfunktionen diskretisiert und differenziert vorliegen, werden im folgenden Kapitel die verwendeten numerischen Optimierungsverfahren vorgestellt um die Werte der Parameter zu approximieren, welche die Funktionen minimieren, und somit das zu Beginn erläuterte Bildregistrierungsproblem zu lösen.

4 Numerische Optimierungsverfahren

Das Bildregistrierungsproblem besteht darin, eine Funktion φ zu finden, welche eine Zielfunktion \mathcal{J} minimiert. Wie bereits erläutert, ist die Lösung mit analytischen Methoden nicht berechenbar.

Um die Anwendung von numerischen Optimierungsverfahren zu ermöglichen, wurden die in Kapitel 2 vorgestellten Zielfunktionen diskretisiert.

In diesem Kapitel werden problemangepasste Verfahren aus [6] vorgestellt, um eine numerische Lösung des Minimierungsproblems zu ermitteln.

Die diskretisierten Zielfunktionen lauten:

$$\begin{aligned}\mathcal{J}_p^h(\theta) &= D^h(A_\theta^{-1} \cdot \bar{T}(\bar{\varphi}_\theta), \bar{R}) \\ \mathcal{J}_{np}^h(\bar{\varphi}) &= D^h(B \cdot \bar{T}(\bar{\varphi}), \bar{R}) + \alpha \mathcal{S}^h(\bar{u}) \\ \mathcal{J}_{npb}^h(\bar{\varphi}, \bar{B}) &= D^h\left(\begin{pmatrix} B_{11} \odot \bar{T}_1(\bar{\varphi}) + B_{12} \odot \bar{T}_2(\bar{\varphi}) \\ B_{21} \odot \bar{T}_1(\bar{\varphi}) + B_{22} \odot \bar{T}_2(\bar{\varphi}) \end{pmatrix}, \bar{R}\right) + \alpha \mathcal{S}^h(\bar{u}) + \beta \mathcal{P}^h(\bar{\varphi}, \bar{B})\end{aligned}$$

Zur numerischen Bestimmung der Minimierer der Zielfunktionen wird ein allgemeines Abstiegsverfahren mit dem Gauß-Newton-Verfahren verwendet.

Dabei wird ausgehend von einem fest gewähltem Startwert x_0 in jedem Iterationsschritt eine Suchrichtung p_k und eine Schrittweite α_k berechnet, um die neue Iterierte wie folgt zu bestimmen

$$\begin{aligned}x_{k+1} &= x_k + \alpha_k p_k \\ x_k, p_k &\in \mathbb{R}^n, \quad \alpha_k \in \mathbb{R}^+\end{aligned}$$

Im Folgenden werden die Suchrichtung und die Schrittweite bestimmt.

4.1 Bestimmung der Suchrichtung

Bei einer gegebenen Schrittweite α wird eine Suchrichtung p gesucht, für die gilt, dass der neue Funktionswert kleiner sein soll als der vorherige:

$$f(x + \alpha p) < f(x)$$

Mit Approximation durch die Taylornetwicklung, lautet der Funktionswert der neuen Iterierten

$$f(x + \alpha p) = f(x) + \alpha \nabla f(x)^T p + \mathcal{O}(\alpha^2)$$

eingesetzt in die erste Ungleichung folgt

$$f(x) + \alpha \nabla f(x)^T p + \mathcal{O}(\alpha^2) < f(x)$$

$$\Leftrightarrow \alpha \nabla f(x)^T p + \mathcal{O}(\alpha^2) < 0$$

Für kleine α kann der Fehlerterm vernachlässigt werden, so dass folgt

$$\nabla f(x)^T p < 0$$

Jede Suchrichtung p , welche diese Eigenschaft erfüllt, heißt *Abstiegsrichtung*.

Das bestehende Minimierungsproblem kann als ein „least-squares“ Problem formuliert werden, da jede Teilfunktion D^h , S^h und P^h als eine Summe von quadrierten Teilfunktionen wie folgt aufgefasst werden kann:

$$r : \mathbb{R}^n \rightarrow \mathbb{R}^m, \quad r_i : \mathbb{R}^n \rightarrow \mathbb{R}$$

$$f(x) = \frac{1}{2} \sum_{i=1}^m (r_i(x))^2 = \frac{1}{2} \|r(x)\|_2^2 = \frac{1}{2} r(x)^T r(x)$$

Mit gegebener Funktion

$$f(x) = \|r(x)\|_2^2$$

wird eine Abstiegsrichtung p gesucht, die den Funktionswert minimiert

$$\min_p f(x + p)$$

Der Funktionswert kann durch die Taylorentwicklung der Hilfsfunktion r approximiert werden als

$$f(x + p) = \|r(x + p)\|_2^2$$

$$\approx \|r(x) + \nabla r(x)^T p\|_2^2 =: \hat{f}(p)$$

Diese Funktion soll in p minimiert werden. Gesucht ist demzufolge ein p , für das der zugehörige Gradient verschwindet

$$\nabla \hat{f}(p) = \nabla r(x) (r(x) + (\nabla r(x))^T p) \stackrel{!}{=} 0$$

$$\Leftrightarrow \nabla r(x) r(x) + \nabla r(x) (\nabla r(x))^T p \stackrel{!}{=} 0$$

Dies führt zu folgendem Gleichungssystem

$$(\nabla r(x) \nabla r(x)^T) p = -\nabla r(x) r(x)$$

Dieses lineare Gleichungssystem ist lösbar, da die Matrix $(\nabla r(x) \nabla r(x)^T)$ symmetrisch positiv definit (s.p.d.) ist.

$$p = -(\nabla r(x) \nabla r(x)^T)^{-1} \nabla r(x) r(x)$$

Wegen

$$\nabla^2 \hat{f}(p) = (\nabla r(x) \nabla r(x)^T) \quad \text{s.p.d.}$$

folgt, dass p ein Minimum von \hat{f} ist.

Diese Vorgehensweise zur Bestimmung der Suchrichtung p nennt sich *Gauß-Newton-Verfahren*. Es ist dem Newton-Verfahren darin überlegen, dass die zweite Ableitung der Funktion f nicht in jedem Iterationsschritt bestimmt werden muss, da sie durch erste Ableitungen approximiert wird.

4.2 Schrittweitenbestimmung

Angenommen eine Suchrichtung p sei fest gegeben, dann wird eine Schrittweite α^* gesucht, so dass der Funktionswert der neuen Iterierten in Richtung p minimiert wird, so dass gilt

$$\alpha^* = \operatorname{argmin}_{\alpha} f(x + \alpha p)$$

Die *Wolfe-Bedingung* (manchmal auch Armijo-Bedingung) sorgt dafür, dass die Zielfunktionen einen hinreichenden Abstieg erfährt und lautet

$$f(x_{k+1}) \leq f(x_k) + c \cdot \alpha \nabla f(x_k)^T p_k, \quad c \in]0, 1[$$

Armijo-Verfahren

Zur Bestimmung des optimalen α^* , wird ausgehend von $\alpha = 1$ die aktuelle Schrittweite so lange halbiert, bis die Wolfe-Bedingung erfüllt ist. Falls stattdessen eine maximale Anzahl an Iterationen überschritten wurde, ist die Suche in die Richtung p fehlgeschlagen.

Wenn t die Anzahl der Iterationen bezeichnet, ergibt sich die zugehörige Schrittweite wie folgt

$$\alpha = 2^{-t}$$

Da die Schrittweite α bereits bei relativ geringen t so klein wird, dass nur noch eine unwesentliche Veränderungen der Iterierten bewirkt werden kann, sollte die Iteration bereits nach wenigen Schritten (z.B. 10) abgebrochen werden.

Mit den Verfahren zur Bestimmung der Suchrichtung und der Schrittweite kann die zu Beginn vorgestellte Iteration durchgeführt werden. Zur Vermeidung von Endlosschleifen sind sinnvolle Bedingungen notwendig, die gewährleisten, dass es in jedem Fall zu einem Abbruch kommt.

4.3 Abbruchkriterien

Es muss dafür gesorgt werden, dass die Iteration in jedem Fall stoppt, daher sind Abbruchbedingungen unerlässlich. Die iterative Suche ist beendet, wenn keine Lösung gefunden werden konnte oder wenn eine hinreichend gute Lösung gefunden wurde.

Es werden die fünf *Abbruchkriterien nach Gill, Murray und Wright* [2] verwendet. Diese lauten wie folgt:

- (K1) $|f(x_{k+1}) - f(x_k)| < \tau (1 + |f(x_{k+1})|)$
ist erfüllt, wenn es keinen Abstieg in den Funktionswerten mehr gibt.
- (K2) $\|x_{k+1} - x_k\|_2^2 < \sqrt{\tau} (1 + \|x_{k+1}\|)$
Es gibt keine (großen) Änderungen zwischen den Iterationswerten.
- (K3) $\|\nabla f(x_{k+1})\|_2^2 < \sqrt[3]{\tau} (1 + |f(x_{k+1})|)$
gilt, wenn die Gradientennorm zu klein wird.
- (K4) $\|\nabla f(x_{k+1})\|_2^2 < \varepsilon$
Der Gradient ist kleiner als die Maschinengenauigkeit.
- (K5) $k > k_{\max}$
Die maximale Anzahl an Iterationen ist erreicht.

Der Abbruch der Iteration erfolgt, wenn $(K1 \wedge K2 \wedge K3) \vee K4 \vee K5$.

Die Ergebnisse im folgenden Kapitel sind das Resultat einer numerischen Optimierung auf θ , $\bar{\varphi}$ und ggf. \bar{B} , unter Verwendung der hier vorgestellten Verfahren.

5 Ergebnisse

Im Folgenden wird gezeigt, dass die Registrierung von Flussdaten mit den vorgestellten drei Zielfunktionen erfolgreich durchgeführt werden kann. Dazu werden die Verfahren an konstruierten Beispieldaten getestet, welche grundsätzlich an die Struktur von Blutgefäßen angelehnt sind. Dabei handelt es sich um Vektorfelder, die den Fluss in einer Röhrenstruktur simulieren. Durch Anwendung der verschiedenen Verfahren auf die synthetischen Daten werden die Gemeinsamkeiten und Unterschiede zwischen den Ansätzen basierend auf den drei Zielfunktionen erläutert.

5.1 Validierung für parametrische Deformationen

Zu Beginn wird die parametrische Registrierung mit \mathcal{J}_p^h evaluiert. Dazu wird gezeigt, dass die Deformationen, welche auf einer affin-linearen Abbildung basieren, erfolgreich rekonstruiert werden können.

Für die Experimente wird als Template T eine gerade Röhrenstruktur mit einem gleichmäßigem Fluss verwendet. Das Referenzbild R ist jeweils aus T durch eine parametrische Deformation entstanden und wird durch den Ansatz mit der parametrischen Zielfunktion \mathcal{J}_p^h registriert.

5.1.1 Translation

Um zu verifizieren, dass der vorgestellte parametrische Ansatz Translationen rekonstruieren kann, wird das Referenzbild R aus dem Templatebild T durch eine Verschiebung erstellt. Diese sind in den ersten zwei Spalten der Abbildung 5.1 dargestellt. Das Ergebnis der Registrierung ist durch das deformierte Template T_φ in der letzten Spalte abgebildet. Ein Vergleich zwischen T und T_φ zeigt, dass die Länge und Orientierung der Bildvektoren erhalten bleiben. Zusätzlich lässt sich feststellen, dass das deformierte Template T_φ dem Referenzbild R entspricht, bis auf kleine Abweichungen am Rand der Röhrenstruktur, die hier bedingt durch die Interpolationsmethode auftreten.

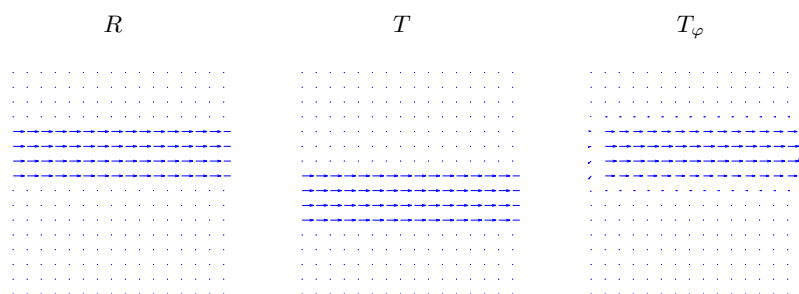


Abbildung 5.1: Registrierungsergebnis der Translation eines gleichmäßigen Vektorfeldes mit \mathcal{J}_p^h .

5.1.2 Skalierung

Für dieses Experiment wurde R aus T durch einer Veränderung der Breite des Flusses und der Länge der Vektoren erstellt.

In der Abbildung 5.2 sind in den ersten beiden Spalten R und T dargestellt. In der ersten Zeile wurden die Vektoren in R relativ zu denen von T verlängert, in der zweiten Zeile entsprechend gekürzt. Die dritte Spalte dieser Abbildung zeigt, dass im deformierten Template T_φ jeweils die erwartete erhöhte bzw. verkürzte Länge der Vektoren aus R erreicht wurde. Die Verbreiterung des Flusses wurde für beide Varianten ebenfalls erfolgreich erzielt.

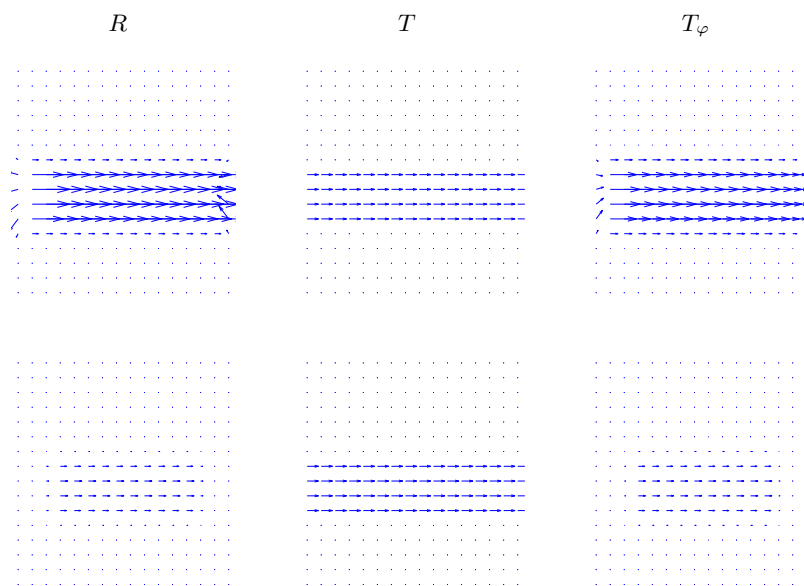


Abbildung 5.2: Registrierungsergebnis der Skalierung eines gleichmäßigen Vektorfeldes mit \mathcal{J}_p^h .

5.1.3 Scherung

Durch die Scherung eines gleichförmigen Flusses T entsteht das Referenzbild R . Für eine erfolgreiche Registrierung wird erwartet, dass R aus T so rekonstruiert wird, dass die Länge der Vektoren, sowie die vom Vektorfeld eingenommene Fläche, erhalten bleibt. Tatsächlich wird die erwartete Struktur in dem deformierten Template T_φ erreicht. Dies wird in der Abbildung 5.3 veranschaulicht, in der wie bisher in den ersten drei Spalten jeweils das Referenz- und Templatebild, sowie das deformierte Template dargestellt werden. Das deformierte Gitter in der vierten Spalte zeigt, dass die Deformation die angestrebte Gestalt eines Parallelogramms aufweist, welche durch die Scherung eines Rechtecks entsteht.

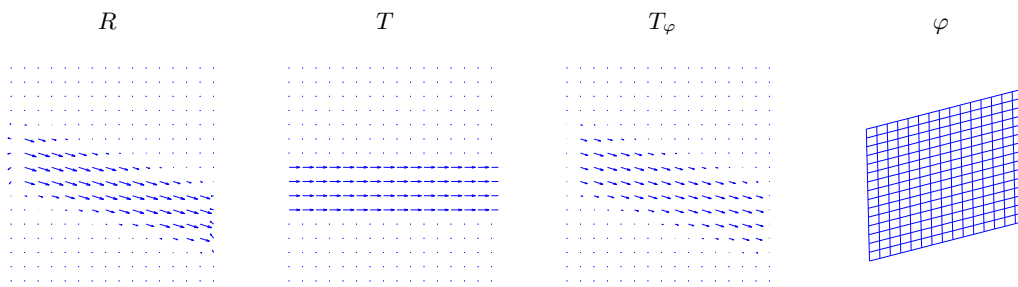


Abbildung 5.3: Registrierungsergebnis der Scherung eines Vektorfeldes mit \mathcal{J}_p^h .

5.1.4 Rotation

Bei der Rotation eines Vektorfeldes wird erwartet, dass die Länge der Vektoren erhalten bleibt, während sich ihre Richtung entsprechend dem gewählten Drehwinkel ändert. Für das Registrierungsproblem wird das Referenzbild R daher als eine rotierte Version von T gewählt. Diese sind in der Abbildung 5.4 in den ersten beiden Spalten dargestellt. Das durch die geschätzten Parameter der Registrierung deformierte Template T_φ befindet sich in der dritten Spalte. Der Vergleich zeigt, dass T_φ eine treffend gedrehte Variante von T ist, die R entspricht. Die zu Grunde liegende geschätzte Deformation ist gleichförmig und längen-, sowie winkelerhaltend, was auch in der Darstellung des deformierten Gitters in der vierten Spalte der Abbildung 5.4 zu erkennen ist.

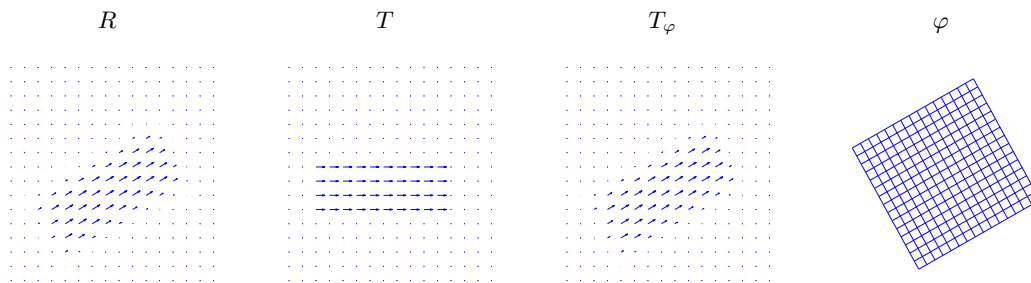


Abbildung 5.4: Registrierungsergebnis einer Rotation eines Vektorfeldes mit \mathcal{J}_p^h .

5.1.5 Nicht affin-lineare Deformation

Liegt eine nicht affin-lineare Deformationen vor, wie beispielsweise eine lokale Veränderung des Flusses, liefert dieser Ansatz unzureichende Ergebnisse.

Für das Registrierungsproblem aus Abbildung 5.5, wurde das bisher verwendete gerade Gefäß lokal verengt um ein Referenzbild zu erhalten. Das abgebildete deformierte Template und das deformierte Gitter zeigen, dass der parametrische Ansatz in diesem Beispiel fälschlicherweise zur Schätzung der Identität als Deformation führt. Dies veranschaulicht, dass derartige Veränderungen nicht mehr kompensiert werden können.

Zusammenfassend lässt sich feststellen, dass zu Grunde liegende affin-lineare Deformationen mit der parametrischen Registrierung durch \mathcal{J}_p^h korrekt rekonstruiert werden.

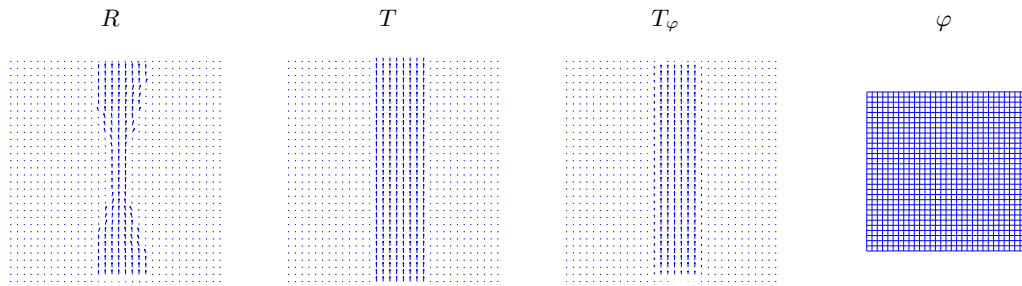


Abbildung 5.5: Registrierungsergebnis der parametrischen Registrierung für eine Gefäßverengung.

5.2 Validierung für nichtparametrische Deformationen

In diesem Abschnitt wird die Registrierung von Vektorfeldern mit den nichtparametrischen Ansätzen basierend auf den zwei Zielfunktionen \mathcal{J}_{np}^h und \mathcal{J}_{npb}^h für verschiedene Beispiele durchgeführt und bewertet. Dazu werden zunächst Beispiele betrachtet, die an eine Stenose und Dilatation von Blutgefäßen angelehnt sind.

Anschließend werden globale Veränderungen eines Bildausschnittes durch jeweils ein Beispiel für eine einfache und zweifache Biegung eines Blutgefäßes betrachtet, wobei dann zusätzlich unterschiedliche Flussgeschwindigkeiten berücksichtigt werden.

Die präsentierten Ergebnisse wurden durch manuelle Festlegung der Parameter erzeugt und stellen eine Repräsentation der Gesamtheit der getesteten Parameter da.

5.2.1 Lokale Deformationen

Als Beispiele für lokale Veränderungen von Blutgefäßen werden jeweils eine Vasokonstriktion (Gefäßverengung) und eine Vasodilatation (Gefäßerweiterung) betrachtet. Es wird erwartet, dass die Strukturen im Referenzbild hauptsächlich durch lokale Adaption des Templatebildes mit der geschätzten Deformation erreicht werden.

Zur Feststellung ob die nichtparametrische Registrierung lokale Verengungen in Gefäßen erfolgreich registrieren kann, wurde eine räumlich begrenzte Verringerung des Gefäßdurchmessers simuliert, die durch Transformation eines gleichförmigen Flusses mit einer nichtlinearen Deformation erwirkt wurde. Die Richtung der Vektoren wird durch die umgebene Gefäßstruktur beeinflusst, so dass die Flussvektoren vor der Verengung zur Mitte des Gefäßes weisen, und danach auseinanderstreben, wie es physiologisch erwartet wird. Die Strömung vor und nach der Verengung bleibt homogen und unidirektional.

Das zugehörige Registrierungsproblem wird in der ersten Zeile der Abbildung 5.6 dargestellt. Es hat sich gezeigt, dass beide nichtparametrische Ansätze in der Lage sind die neue Flusstruktur erfolgreich durch die geschätzten Parameter zu rekonstruieren.

In der zweiten und dritten Zeile der Abbildung 5.6 sind jeweils die Registrierungsergebnisse mit den beiden nichtlinearen Ansätzen basierend auf den Zielfunktionen \mathcal{J}_{np}^h und \mathcal{J}_{npb}^h dargestellt. Diese zeigen, dass das deformierte Template T_φ jeweils dem Referenzbild R entspricht, bis auf kleine, vernachlässigbare Interpolationsfehler am Gefäßrand, die in der dritten Spalte dargestellt sind. Die Betrachtung der deformierten Gitter in der zweiten Spalte zeigt, dass die Deformationen lokal wirken.

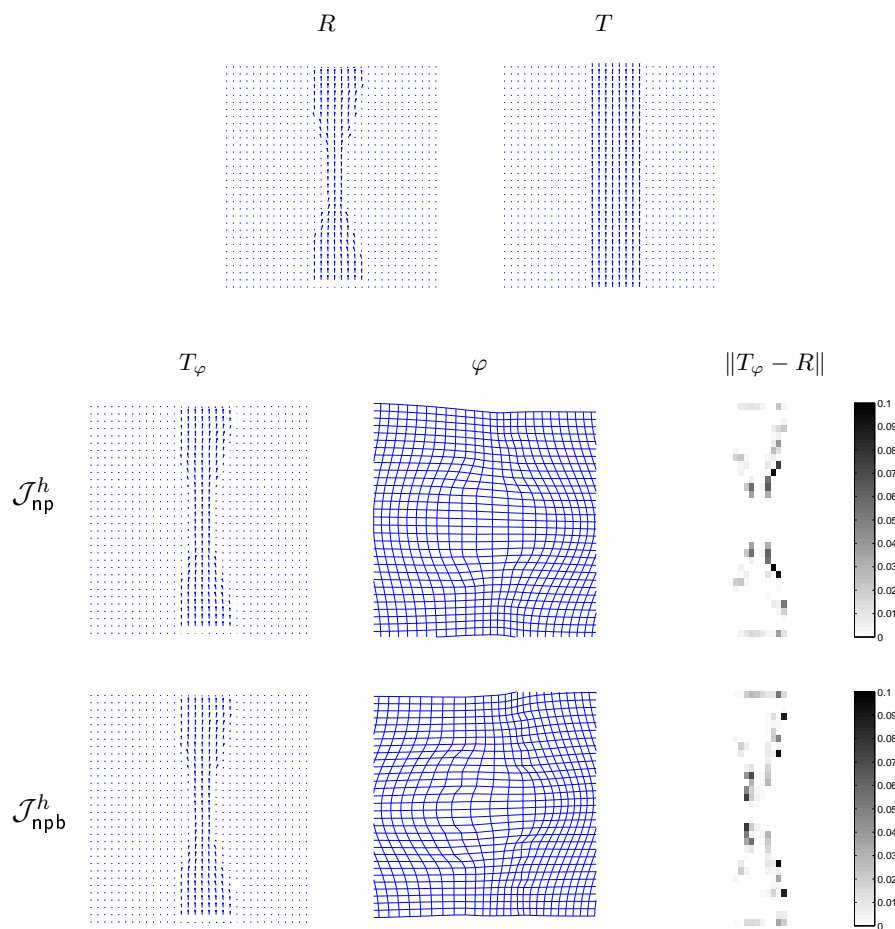


Abbildung 5.6: Ergebnis der nichtparametrischen Registrierung für lokal verengte Blutgefäße.

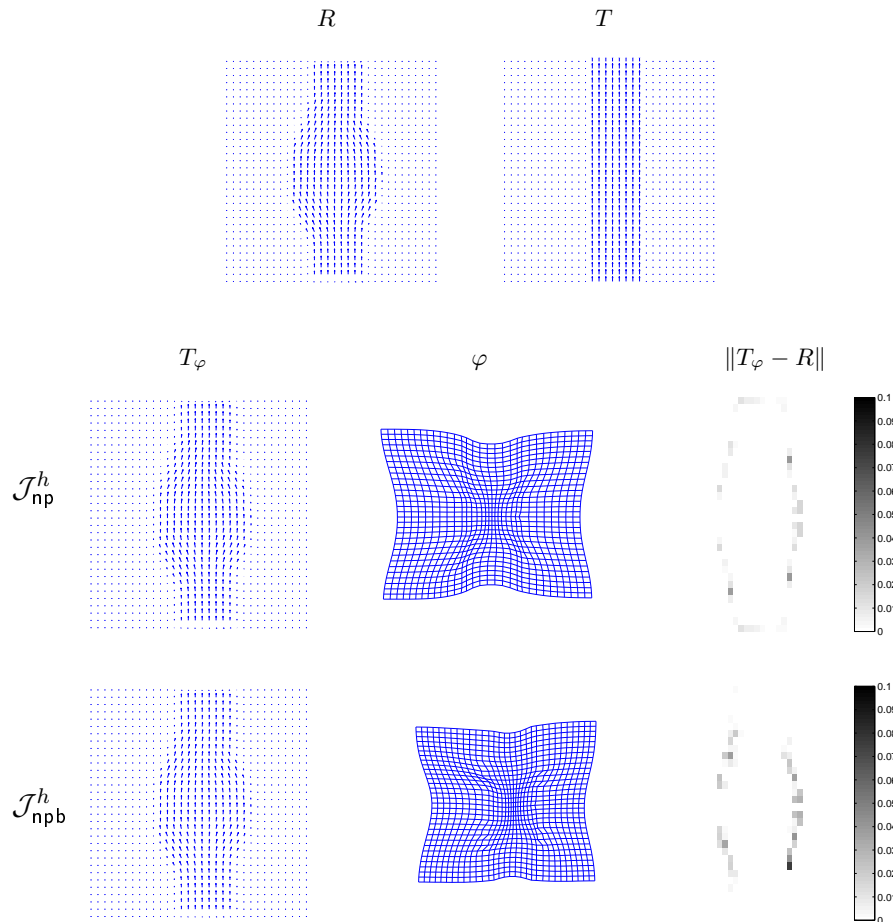


Abbildung 5.7: Ergebnis der nichtparametrischen Registrierung für lokal ausgedehnte Blutgefäße.

Als weiteres Beispiel wird die lokale Dilatation eines Gefäßes betrachtet, die hier ebenfalls durch die Anwendung einer nichtparametrischen Deformation auf einen gleichförmigen Fluss herbeigeführt wurde. Wie im vorangegangenen Beispiel, passen sich die enthaltenen Flussvektoren der umliegenden simulierten Gefäßstruktur durch eine angemessene Adaption ihrer Orientierung an den entsprechenden Stellen an, während ihre Länge weitgehend erhalten bleibt. Das dieser Beschreibung entsprechende Registrierungsproblem bestehend aus Template- und Referenzbild wird in der ersten Zeile der Abbildung 5.7 dargestellt.

Wie die erste Spalte dieser Abbildung veranschaulicht, wird durch die Registrierung mit beiden nichtparametrischen Zielfunktionen eine zufriedenstellende Ähnlichkeit zwischen dem deformierten Template und dem Referenzbild erreicht. Das Differenzbild in der dritten Spalte zeigt, dass die verbleibenden Fehler gering sind und sich wie bereits im vorangegangenen Beispiel, ausschließlich im Randbereich befinden. Das zugehörige Gitter in der zweiten Spalte belegt, dass die größte Deformation in dem Bereich vorliegt, in dem die Unterschiede zwischen Referenz- und Templatebild am größten sind.

5.2.2 Globale Deformationen

Neben den bisher ausgewählten lokalen Veränderungen eines Gefäßes innerhalb eines Bildausschnittes, werden im Folgenden globale Veränderungen am physiologisch naheliegendem

Beispiel von gebogenen Blutgefäßstrukturen betrachtet.

Das erste Beispiel ist durch die einfache, einseitige Biegung eines homogenen Flusses erzeugt worden, wobei die Flussgeschwindigkeiten am Rand jeweils leicht beschleunigt wurden, so dass die Vektoren am oberen und unteren Ende jeweils vergleichsweise länger sind als in der Mitte des Gefäßes. Die Orientierung der Flussvektoren wird durch die umgebende Gefäßstruktur bestimmt. Wird der Verlauf von unten nach oben in Flussrichtung betrachtet, weisen die Vektoren am unteren Rand nach links, richten sich dann bis zur Gefäßmitte zunehmend nach oben aus, um anschließend die Richtung des Stromes nach rechts einzunehmen.

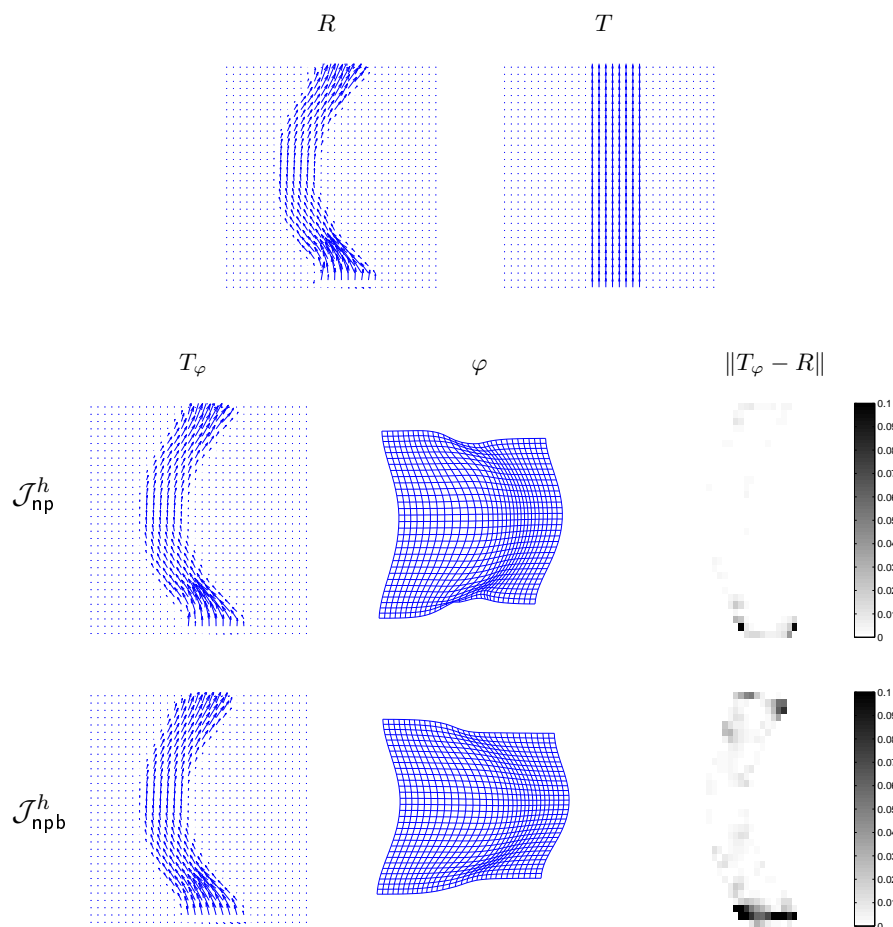


Abbildung 5.8: Ergebnis der nichtparametrischen Registrierung für ein einseitig gebogenes Blutgefäß.

Die Abbildung 5.8 zeigt das beschriebene Beispiel, repräsentiert in Template- und Referenzbild in der ersten Zeile. Die Registrierung unter Verwendung der Ansätze mit den zwei vorgestellten nichtparametrischen Zielfunktionen liefern eine fast fehlerlose Rekonstruktion der erstellten Krümmung inklusive der partiell verschiedenen Flussgeschwindigkeiten, wie die zweite und dritte Zeile der Abbildung veranschaulicht. Die geschätzten Deformationen geben die erwartete Strukturveränderung wieder und entsprechen einander qualitativ.

Zur Erhöhung des Schwierigkeitsgrades und somit zur Erkundung der Grenzen der Algorithmen, wird im Folgenden ein zweifach gebogenes Blutgefäß simuliert, dessen Enden sich fast diagonal gegenüberliegen. Das Gefäß wird leicht schmaler konstruiert als das im Template gewählte. Zusätzlich wird die Flussgeschwindigkeit an einem Gefäßende erneut erhöht, so dass

keine vollständige Rekonstruktion, sondern vermehrtes Auftreten von Randfehlern erwartet wird.

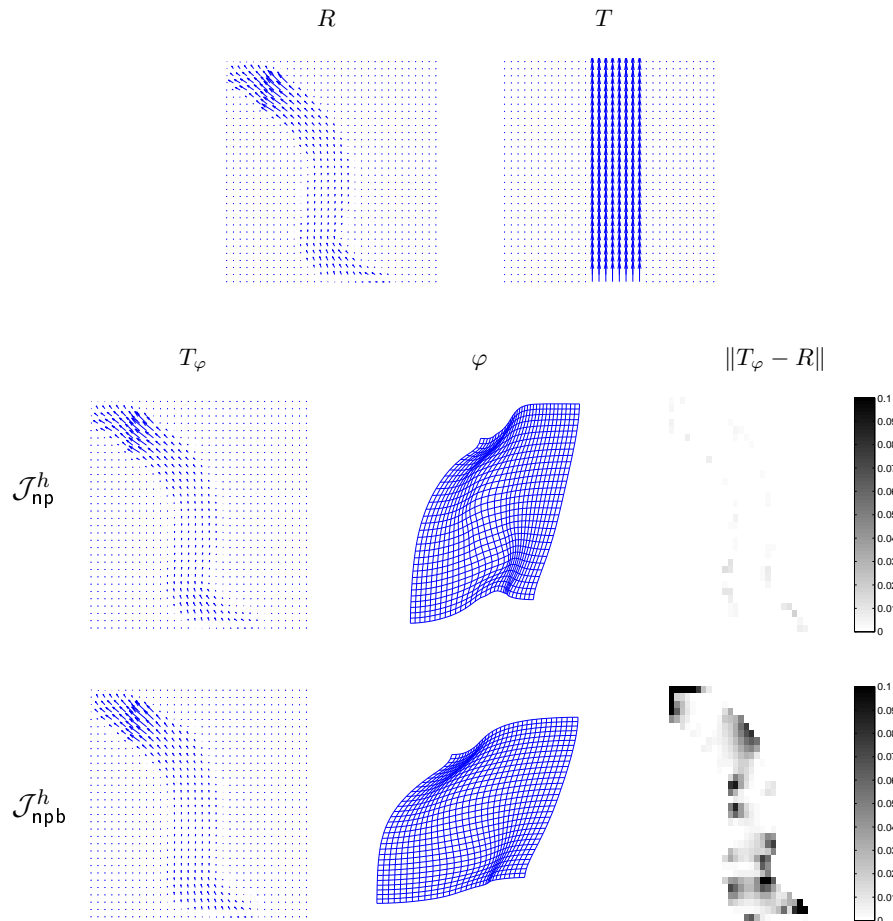


Abbildung 5.9: Ergebnis der nichtparametrischen Registrierung für ein zweifach gebogenes Blutgefäß.

Das erläuterte Registrierungsproblem wird in der ersten Zeile der Abbildung 5.9 dargestellt und basiert wie bisher auf einer nichtparametrischen Modifikation eines gleichförmigen, unidirektionalen Flusses, der erneut als Template dient. Das Ergebnis der Registrierung mit beiden Zielfunktionen liefert eine hinreichende Ähnlichkeit zwischen dem deformierten Template und dem Referenzbild, wobei vergleichsweise mehr Fehler auftreten als bei dem vorangegangenen Beispiel. Wie die zweite und dritte Zeile der Abbildung 5.9 zeigt, weist das deformierte Gitter für beiden Ansätze eine Faltung in dem Bereich auf, in dem die größten Unterschiede zwischen dem Referenz- und Templatebild bestehen.

Insgesamt belegen die durchgeführten Experimente, dass die nichtparametrische Registrierung von Flussdaten mit beiden Ansätzen der Zielfunktionen \mathcal{J}_{np}^h und \mathcal{J}_{npb}^h bei angemessener Parameterwahl erfolgreich durchgeführt werden kann, wobei die Ergebnisse qualitativ gleichwertig sind.

Bisher sind die Ergebnisse ohne quantitative Informationen zu den gewählten Parametern präsentiert worden. Im folgenden Abschnitt werden die Gemeinsamkeiten und Unterschiede der zwei vorgestellten nichtparametrischen Ansätze vorgestellt und diskutiert.

5.2.3 Verhalten bei Über- und Unterregularisierung

Die vorgestellten Resultate der Experimente, entsprechen bisher denen, die von allen Parameterwahlen zu den besten Ergebnissen geführt haben. Da diese für die betrachteten Beispiele grundsätzlich für alle Kombinationen von der Wahl des Regularisierers und der Randbedingung erzielt wurden, ist eine ausführliche Diskussion über die qualitativ nicht vorhandenen Unterschiede der Settings weder notwendig noch angebracht.

Obwohl die Resultate mit verschiedenen Regularisierungsparametern α erzeugt worden sind, wird im Folgenden im Allgemeinen von dem optimalen Parameter als α_{opt} gesprochen, wobei dieser dann auf das jeweils betrachtet Beispiel bezogen ist. Diese Festlegung wird getroffen, um die Begriffe Über- und Unterregularisierung anschaulich beschreiben zu können.

Überregularisierung

Wird der Regularisierer bei der Registrierung durch einen verhältnismäßig großen Regularisierungsparameter α gewichtet, der über dem Wert des optimalen α_{opt} liegt, so wird dies als *Überregularisierung* bezeichnet. Da mit steigendem α die Glattheitsanforderungen an die zu schätzende Deformation φ steigen, nähert sich das Ergebnis der Identität, je höher α gewählt wird.

In Abbildung 5.10 wird in der letzten Spalte das Registrierungsergebnis aus 5.8 mit der Zielfunktion $\mathcal{J}_{\text{np}}^h$ für $\alpha = 10$ präsentiert. Die geschätzte Deformation für $\alpha = 10$ entspricht näherungsweise der Identität. Bei der Registrierung unter Verwendung der Zielfunktion $\mathcal{J}_{\text{npb}}^h$, wird auch mit extremen Werten für den Parameter β keine Anpassung erreicht, was in den ersten drei Spalten der Abbildung 5.10 veranschaulicht wird.

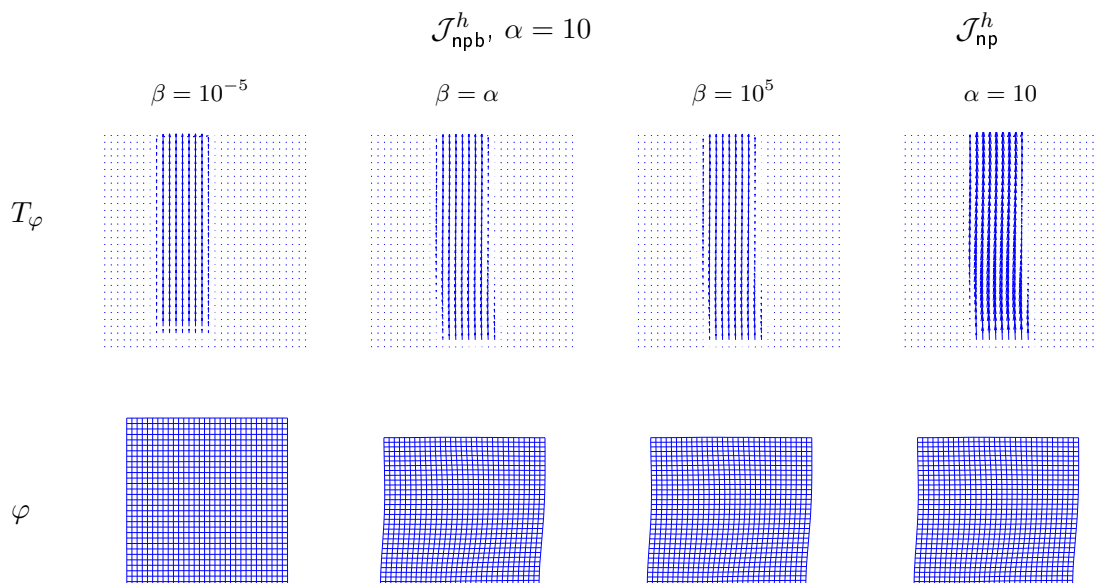


Abbildung 5.10: Veranschaulichung des Einflusses eines großen Regularisierungsparameters α auf die geschätzte Deformation im Vergleich für beide nichtparametrischen Ansätze.

Die Auswertung der bereits gezeigten Beispielprobleme belegt ebenfalls, dass eine Überregularisierung nicht durch eine geschickte Wahl des Straftermparameters β zu kompensieren ist.

Unterregularisierung

Bei der *Unterregularisierung* wird die Registrierung mit einem Regularisierungsparameter α durchgeführt, der deutlich unter dem optimalen Parameterwert α_{opt} liegt. Durch die entsprechend geringeren Glattheitsanforderungen kommt es dann verstärkt zu Unregelmäßigkeiten im deformierten Gitter. Die dadurch herbeigeführten Resultate weisen teilweise eine chaotische Struktur auf und entsprechen in der Regel nicht annähernd dem gewünschten Ergebnis.

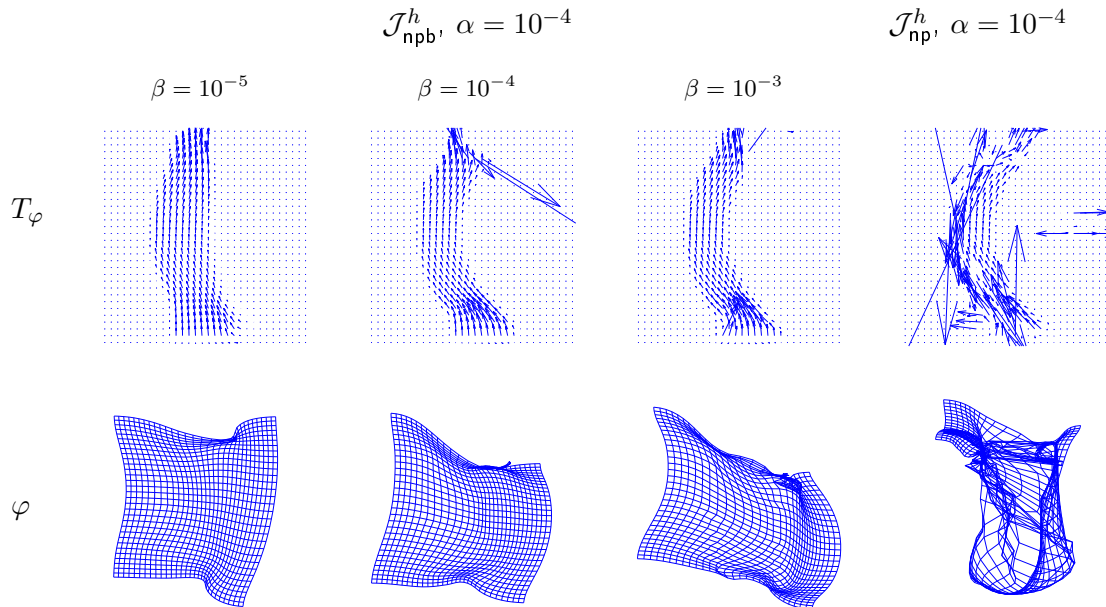


Abbildung 5.11: Veranschaulichung des Einflusses eines kleinen Regularisierungsparameters α auf die geschätzte Deformation im Vergleich für beide nichtparametrischen Ansätze.

Die Registrierung unter Verwendung von zu kleinen Regularisierungsparametern (Werte unter α_{opt}) mit der Zielfunktion $\mathcal{J}_{\text{np}}^h$ führt allgemein zur Schätzung von nicht glatten Deformationen, die durch starke Faltungen im deformierten Gitter gekennzeichnet sind. In der letzten Spalte der Abbildung 5.11 wird anhand des Registrierungsbeispiels aus der Abbildung 5.8 eine Unterregularisierung für $\alpha = 10^{-4}$ veranschaulicht. Das zugehörige deformierte Gitter in der zweiten Zeile ist hier stark in sich selbst gefaltet. Das korrespondierende deformierte Template enthält Flussvektoren verschiedenster Länge und Orientierungen, die vereinzelt an Stellen auftreten, an denen weder Template- noch Referenzbild Flüsse aufweist.

Die ersten drei Spalten zeigen die Ergebnisse einer Registrierung mit der nichtparametrischen Zielfunktion $\mathcal{J}_{\text{npb}}^h$ jeweils für denselben Parameterwert von α und variierenden Straftermparametern β . Im Vergleich zu den Resultaten der Zielfunktion $\mathcal{J}_{\text{npb}}^h$, weisen die deformierten Gitter hier eine deutlich glattere Struktur auf, die sich durch deutlich weniger Faltungen auszeichnet. Entsprechend weisen die deformierten Templates jeweils eine vergleichsweise höhere Ähnlichkeit zum Referenzbild auf.

Wenn das Verhältnis von α und β um mehr als zwei Zehnerpotenzen voneinander abweicht, treten erneut verstärkt Unregelmäßigkeiten in der geschätzten Deformation auf.

Insgesamt belegt die Auswertung der Experimente, dass eine Unterregularisierung bei Verwendung der Zielfunktion $\mathcal{J}_{\text{npb}}^h$ teilweise durch eine geeignet Wahl des Straftermparameters β kompensiert werden kann. Der genaue Wert für den Parameter ist problemabhängig. In den

Beispielen haben sich besonders solche Werte bewährt, die bis zu einer Zehnerpotenz von α nach oben abweichen, so dass die Qualität der Ergebnisse wiederholt durch die Festlegung $\beta \approx \alpha$ bzw. $\beta \approx 10 \cdot \alpha$ gesteigert werden konnte. Diese Feststellung wurde in den Beispielen für lokale Deformationen sogar für solche Regularisierungsparameter α verifiziert, die bis zu fünf Zehnerpotenzen von α_{opt} nach unten abweichen.

Einfluss des Straftermparameters β

Die Auswirkungen des Straftermparameters β auf die Ergebnisse der nichtparametrischen Registrierung wurden bisher für solche Regularisierungsparameter α betrachtet, die stark von α_{opt} abweichen.

Wird der Parameter α in der näheren Umgebung von α_{opt} gewählt, so zeigt sich, dass die Registrierung unter Verwendung der Zielfunktion $\mathcal{J}_{\text{npb}}^h$ dann die besten Ergebnisse liefert, wenn der Straftermparameter β im Bereich $\alpha \leq \beta \leq 10 \cdot \alpha$ liegt.

In den Abbildungen 5.12 und 5.13 werden jeweils die Registrierungsergebnisse der nichtparametrischen Zielfunktionen für $\alpha = \alpha_{\text{opt}}$ anhand von zwei bekannten Beispielen dargestellt. Die letzte Spalte enthält das deformierte Template und Gitter als Ergebnis der Registrierung mit $\mathcal{J}_{\text{np}}^h$ für den optimalen Registrierungsparameter α_{opt} . In den ersten drei Spalten sind jeweils die zugehörigen Ergebnisse von $\mathcal{J}_{\text{npb}}^h$ für verschiedene Straftermparameter abgebildet. Es zeigt sich, dass sich die geschätzte Deformation bei sinkendem β der Identität immer stärker annähert, während die Festlegung von sehr großen Werten für β nur bei extremen Werten zu Unregelmäßigkeiten im Gitter führt.

Wird der Regularisierungsparameter jeweils um eine Zehnerpotenz kleiner als α_{opt} gewählt, treten bei der Registrierung mit $\mathcal{J}_{\text{np}}^h$ in diesen Beispielen Faltungen im Gitter auf. Diese sind in der letzten Spalte der Abbildungen 5.14 und 5.15 zu sehen. Ein Vergleich mit den ersten drei Spalten veranschaulicht, dass die Anwendung der Zielfunktion $\mathcal{J}_{\text{npb}}^h$ die Ergebnisse für diese Fälle deutlich verbessern kann. Gilt $\beta \approx \alpha$ wird eine glattere Deformation erreicht als bei der Verwendung von $\mathcal{J}_{\text{np}}^h$. Für kleinere Werte nähert sich die geschätzte Deformation der Identität, während bei großen Werten von β mit $\beta > \alpha$, erneut Unregelmäßigkeiten auftreten.

Abschließend wird die Auswirkung des Straftermparameters für solche Regularisierungsparameter α betrachtet, die über α_{opt} liegen. Für die Registrierung mit $\mathcal{J}_{\text{np}}^h$ wurde bereits festgestellt, dass sich die geschätzte Deformation der Identität nähert, wenn α sehr groß gewählt wird und entsprechend eine Überregularisierung vorliegt. Diese Tendenz ist bereits für weniger extreme α abzusehen, wenn $\alpha > \alpha_{\text{opt}}$ gewählt wird. In den Abbildungen 5.16 und 5.17 ist in der letzten Spalte wie bisher das deformierte Template und Gitter der Registrierung mit $\mathcal{J}_{\text{np}}^h$ dargestellt. Ein Vergleich mit den Abbildungen 5.12 und 5.13 zeigt, dass die Deformation bereits weniger ausgeprägt ist. Entsprechend der dadurch bedingten höheren Glattheitsanforderung, hat die Variation des Straftermparameters β einen vergleichsweise geringeren Einfluss. Mit steigenden Werten des Straftermparameters werden nur noch geringfügige Veränderungen der Ergebnisse erwirkt, bis für sehr große Werte von β schließlich kein Einfluss auf die Lösung mehr quantifizierbar ist. Wie die Betrachtung der zweiten und dritten Spalten der Abbildungen 5.16 zeigt, ist beim Vergleich der Werte $\beta = \alpha$ und $\beta = 10^5$ kaum mehr ein Unterschied zwischen den T_φ und φ feststellbar, obwohl sich die Straftermparameter um mehrere Zehnerpotenzen unterscheiden. Wird β hingegen deutlich kleiner als α gewählt, nähert sich die geschätzte Deformation mit sinkendem Straftermparameter der Identität an.

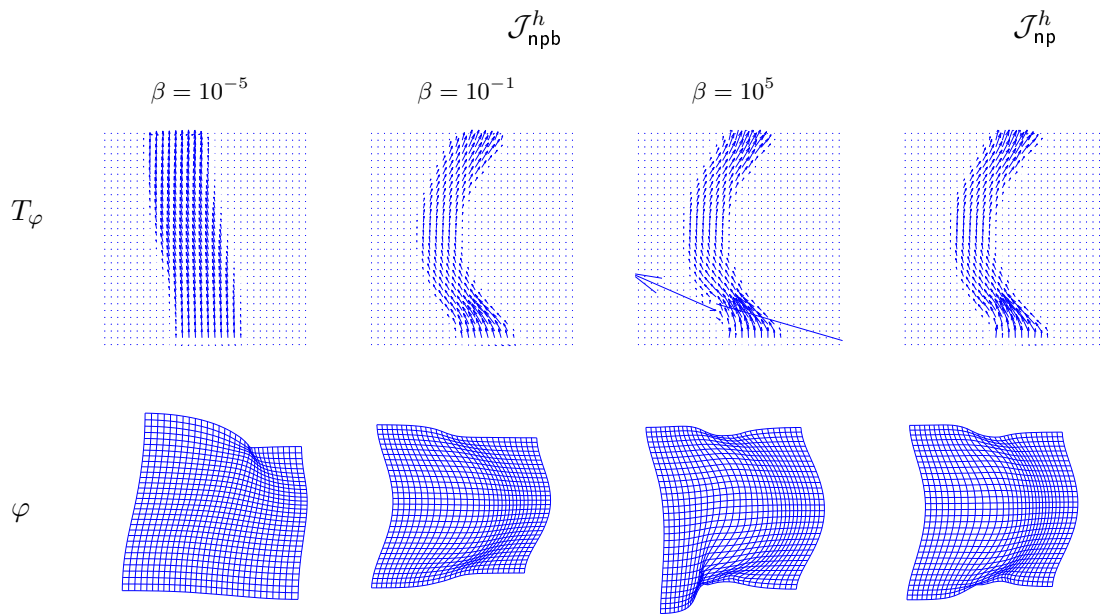


Abbildung 5.12: Einfluss des Straftermparameters β für $\alpha = 10^{-2} = \alpha_{\text{opt}}$

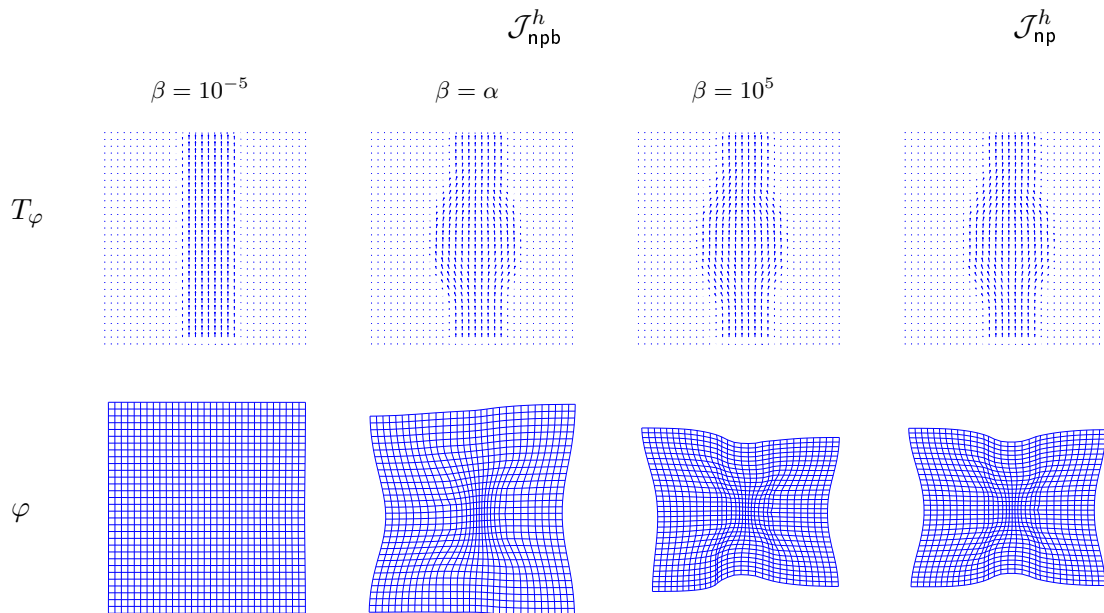


Abbildung 5.13: Einfluss des Straftermparameters β für $\alpha = 100 = \alpha_{\text{opt}}$

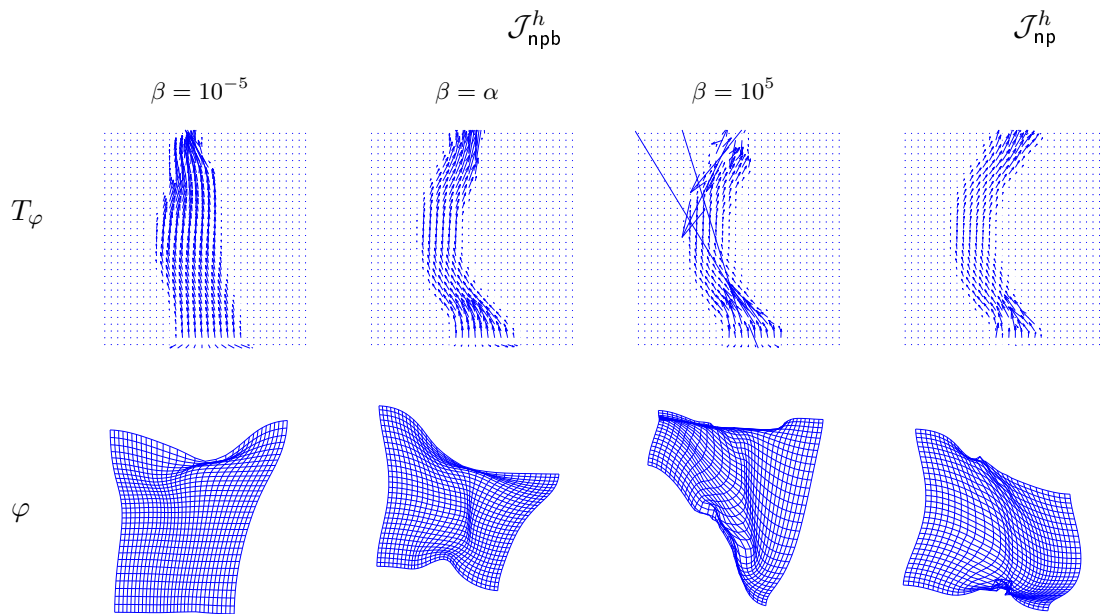


Abbildung 5.14: Einfluss des Straftermparameters β für $\alpha = 10^{-3} < \alpha_{\text{opt}} = 10^{-2}$

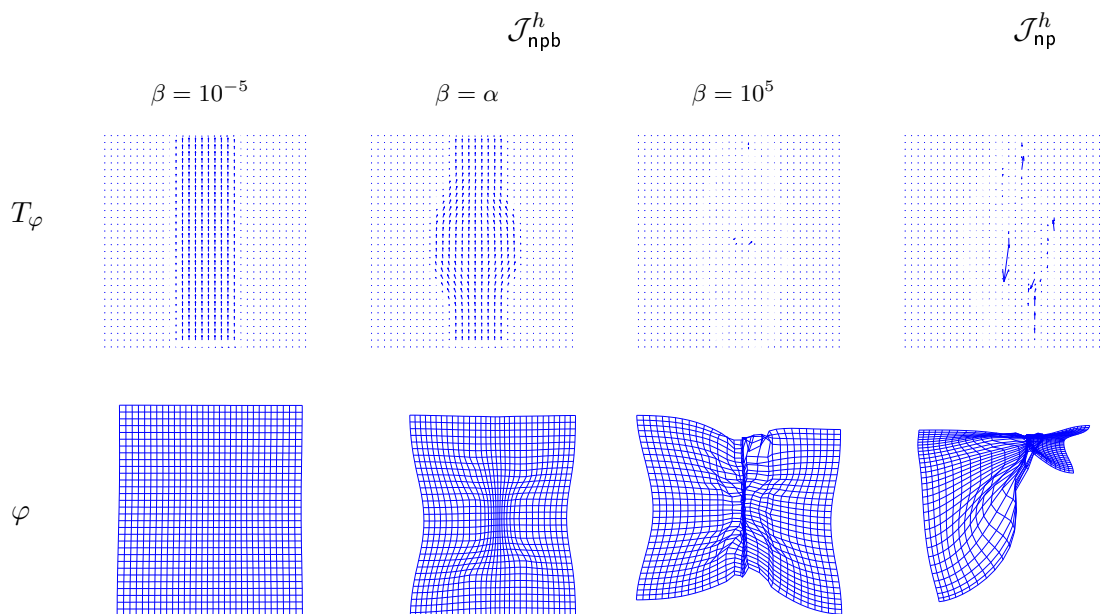


Abbildung 5.15: Einfluss des Straftermparameters β für $\alpha = 10 < \alpha_{\text{opt}} = 100$

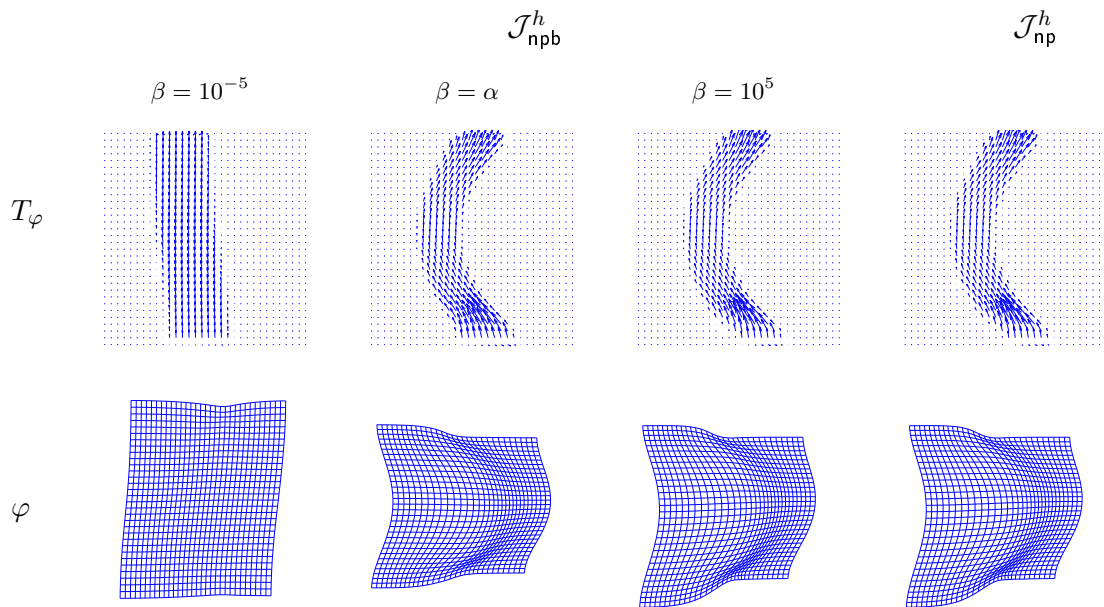


Abbildung 5.16: Einfluss des Straftermparameters β für $\alpha = 10^{-1} > \alpha_{\text{opt}} = 10^{-2}$

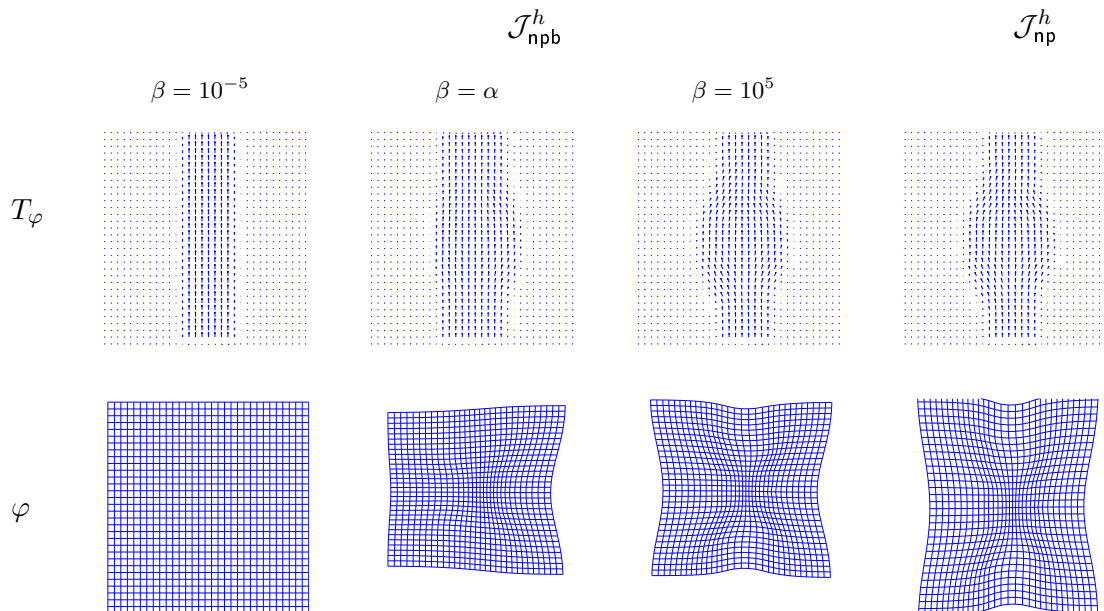


Abbildung 5.17: Einfluss des Straftermparameters β für $\alpha = 10^3 > \alpha_{\text{opt}} = 100$

5.3 Validierung für dreidimensionale Daten

Bisher wurden die Verfahren explizit für den zweidimensionalen Fall im Detail hergeleitet und evaluiert, mit dem Hinweis, dass die Methoden analog auf beliebige Dimensionen übertragen werden können. In diesem Abschnitt soll abschließend exemplarisch an einfachen Beispielen gezeigt werden, dass die vorgestellten Methoden auch für die dreidimensionale Registrierung geeignet sind.

Wie bereits für die zweidimensionalen Problemstellungen, wurden die Algorithmen für dreidimensionale Anwendung anhand von Daten evaluiert, die angelehnt an echte Blutgefäßstrukturen erstellt wurden.

Exemplarisch wird zu Beginn die Rotation eines gleichmäßigen Flusses in einer dreidimensionalen Röhrenstruktur betrachtet. In der Abbildung 5.18 wird das Ergebnis der zugehörigen parametrischen Registrierung anhand des deformierten Templates dargestellt. Es lässt sich feststellen, dass die gesuchte Reorientierung gefunden wird.

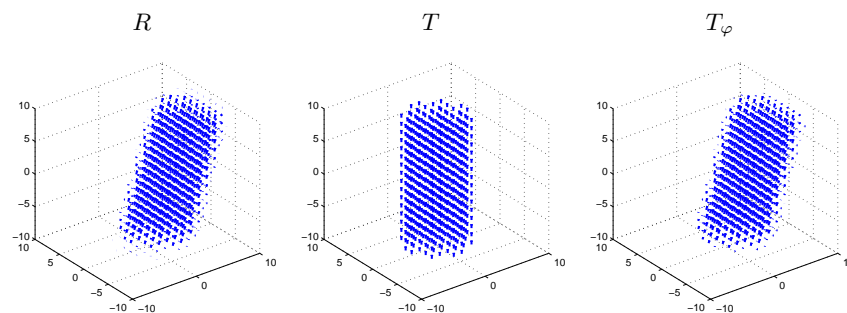


Abbildung 5.18: Ergebnis der parametrischen Registrierung anhand einer dreidimensionalen Rotation

Als Beispiel einer nichtparametrischen Registrierung wird ein dreidimensionales, einseitig gebogenes Blutgefäß simuliert, wobei die Krümmung entlang der vertikalen z -Achse konstruiert wird. Das nichtparametrische Registrierungsproblem besteht darin, die gebogene Gefäßstruktur aus einer geraden Röhrenstruktur mit gleichförmigen Fluss zu rekonstruieren, was in den ersten zwei Spalten der Abbildung 5.19 illustriert wird. Die zugehörigen Ergebnisse der zwei nichtparametrischen Zielfunktionen sind in den letzten zwei Spalten dargestellt.

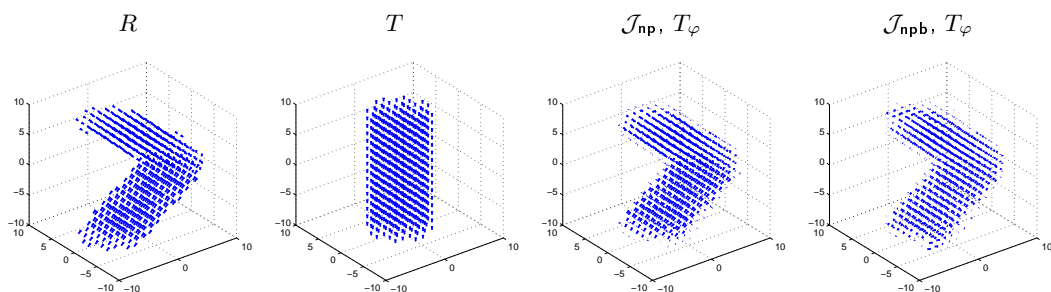


Abbildung 5.19: Registrierung eines dreidimensionalen Flussbildes mit beiden nichtparametrischen Zielfunktionen

Grundsätzlich sind vielversprechende Formanpassungen für beide Verfahren zu erkennen, dennoch fällt auch bei diesem Beispiel eine Beurteilung der Qualität schwer. Daher wird ergän-

zend der Registrierungsfehler ebenenweise betrachtet. Die Abbildung 5.20 zeigt die Norm der Differenz zwischen den Vektoren von R und T_φ .

Unter gemeinsamer Berücksichtigung der Grafiken kann nun die Aussage getroffen werden, dass die Güte der Registrierungsergebnisse sehr gut ist. Es treten nur am äußersten Bildrand Fehler auf. Dies führt zu der abschließenden Feststellung, dass auch dreidimensionale Registrierungsprobleme mit den vorgestellten Algorithmen erfolgreich durchgeführt werden können.

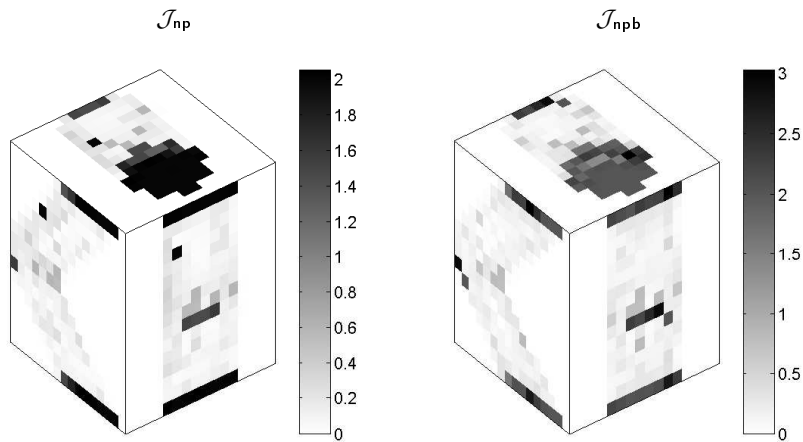


Abbildung 5.20: Fehler der dreidimensionalen Registrierung zu dem Beispiel aus Abbildung 5.19. Dargestellt wird die Norm der Differenz der Flussbilder, projiziert auf drei Seiten eines Würfels.

6 Diskussion

Das Ziel dieser Arbeit war die Registrierung von Flussdaten zu ermöglichen. Dazu wurde auf Verfahren für skalarwertige Bilder zurückgegriffen, die durch Anpassung des Distanzmaßes und des Deformationsmodells modifiziert wurden, um eine erfolgreiche Registrierung von vektorwertigen Bildern zu erreichen.

Der wesentliche Unterschied zur Registrierung mit skalarwertigen Daten besteht in der Basistransformation, die zur Reorientierung der Flussvektoren notwendig ist. Ergänzend wurde ein weiterer Strafterm motiviert, der die Basistransformation als Optimierungsparameter berücksichtigt, so dass insgesamt drei Ansätze mit jeweils verschiedenen Zielfunktionen vorgestellt wurden, davon eine für parametrische und zwei für nichtparametrische Registrierung.

Zur Evaluation wurden die Experimente angelehnt an realen Problemstellungen so konstruiert, dass Strukturveränderungen von Blutgefäßen simuliert wurden. Die Ergebnisse belegen, dass alle drei vorgestellten Ansätze zur Registrierung von Flussdaten geeignet sind. Bei den betrachteten Beispielen wurden durch unterschiedliche Festlegungen der Regularisierer und Randbedingungen keine qualitativen Unterschiede zwischen den Ergebnissen festgestellt.

Zusätzlich wurde ein Zusammenhang zwischen dem Regularisierungsparameter und dem Straftermparameter gefunden. Abhängig von dem Wert des Regularisierungsparameters α , der bei der Verwendung der nichtparametrischen Zielfunktion \mathcal{J}_{np}^h zu den besten Ergebnissen führt, konnte festgestellt werden, dass der Straftermparameter β für \mathcal{J}_{npb}^h zu gleichwertigen Ergebnissen führt, wenn $\beta \approx \alpha$ oder $\beta \approx 10 \cdot \alpha$ gewählt wird. Zudem konnte gezeigt werden, dass die geeignete Wahl des Straftermparameters eine Unterregularisierung durch zu kleine α teilweise kompensieren kann, so dass eine zusätzliche Regularisierungsmöglichkeit besteht.

Da die Parameter manuell gesetzt und getestet wurden, sind weitere Experimente nötig um die gefundenen Zusammenhänge zu evaluieren und zu manifestieren. Bisher ist die Auswertung der Ergebnisse sehr zeitaufwändig, da sie visuell durchgeführt wird und demzufolge schlecht automatisierbar ist.

Auch für dreidimensionale Anwendungen wurde gezeigt, dass eine Registrierung von Flussdaten grundsätzlich erfolgreich möglich ist. Dabei haben sich die eingeschränkten Visualisierungsmöglichkeiten für dreidimensionale Vektorfelder als eine große Hürde erwiesen, welche die Beurteilung der Qualität der Ergebnisse sehr erschwert hat.

Literaturverzeichnis

- [1] Sebastiano Barbieri, Martin Welk, and Joachim Weickert. A variational approach to the registration of tensor-valued images. In Santiago Aja-Fernandez, Rodrigo Luis Garca, Dacheng Tao, and Xuelong Li, editors, *Tensors in Image Processing and Computer Vision*, Advances in Pattern Recognition, pages 59–77. Springer London, 2009.
- [2] Philip E. Gill, Walter Murray, and Margret H. Wright. *Practical Optimization*. Academic Press, 1992.
- [3] Martin Hanke-Bourgeois. *Grundlagen der Numerischen Mathematik und des Wissenschaftlichen Rechnens*. Vieweg & Teubner, 3 edition, 2009.
- [4] Jan Modersitzki. *Numerical Methods for Image Registration*. Oxford University Press, 2004.
- [5] Jan Modersitzki. *FAIR: Flexible Algorithms for Image Registration*. SIAM, 2009.
- [6] Jorge Nocedal & Stephen J. Wright. *Numerical optimization*. Springer Series in Operations Research. Springer, 2 edition, 2006. Unibib Code: QE:6/2.