

# Kapitel 5: Bildregistrierung

Prof. Dr.-Ing. Thomas Schultz

URL: <http://cg.cs.uni-bonn.de/schultz/>

E-Mail: [schultz@cs.uni-bonn.de](mailto:schultz@cs.uni-bonn.de)

Büro: Friedrich-Hirzebruch-Allee 6, Raum 2.117

16./23. Dezember 2024

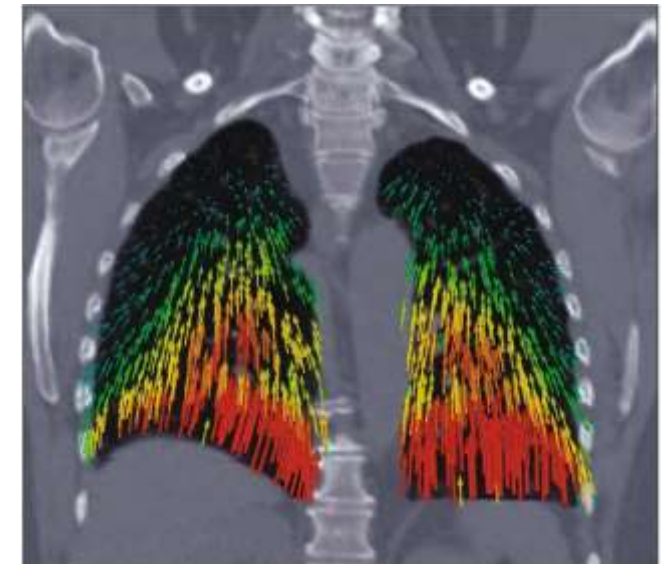
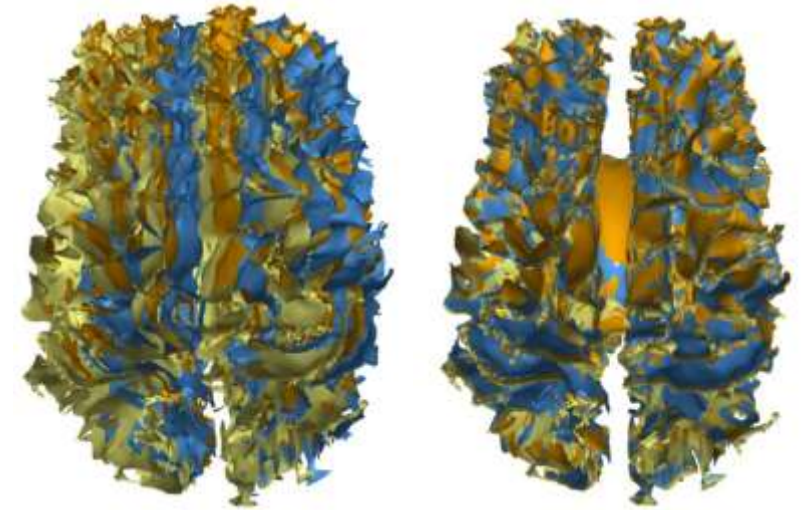
## **5.1 Problemstellung und Evaluierung**

# Anwendungen der Bildregistrierung (Teil 1)

- **Registrierung** bringt bewegliche *Objektbilder* (*Templatebilder*) in Korrespondenz mit einem festen *Referenzbild*.

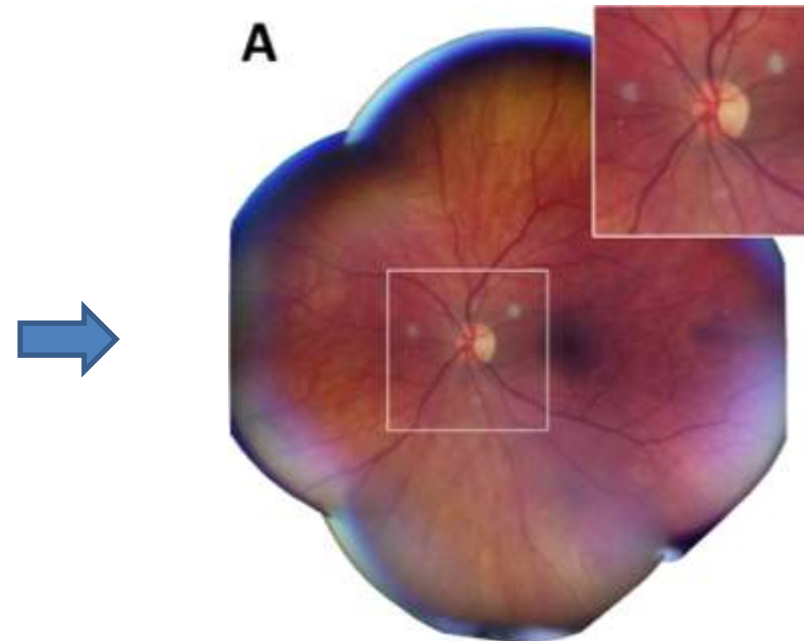
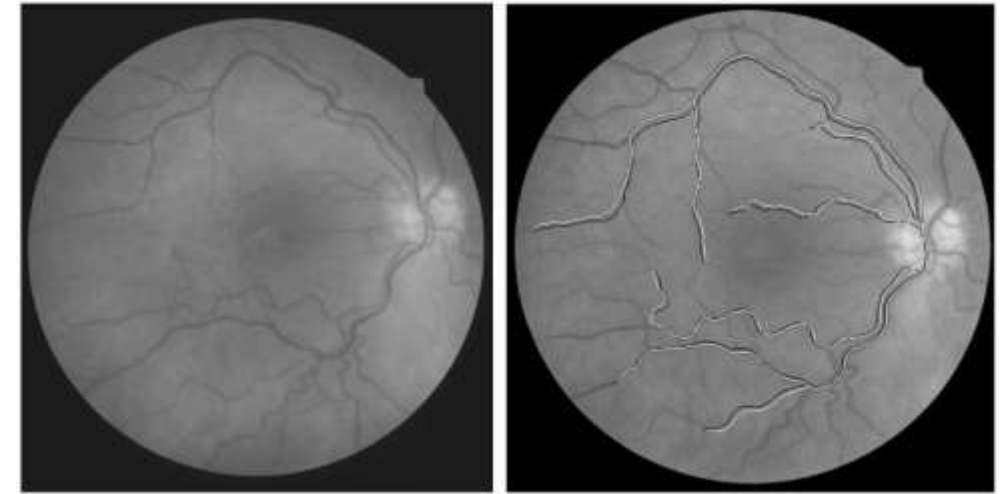
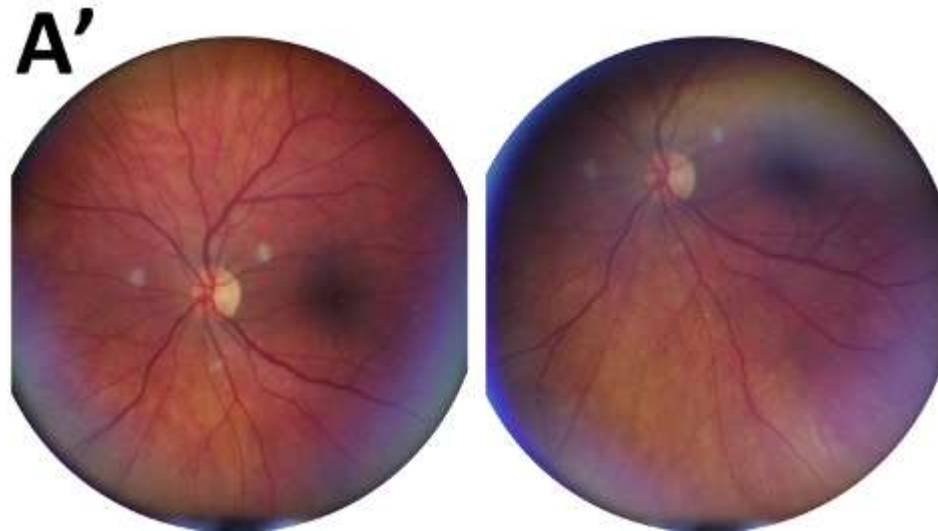
Einsatzbereiche:

- **Korrektur** von **Bewegungen** oder Lagerungsunterschieden bei wiederholten Aufnahmen
- Direkter **Vergleich** verschiedener Patienten oder Zeitpunkte
  - Auf Basis der registrierten Bilder oder der benötigten Deformation



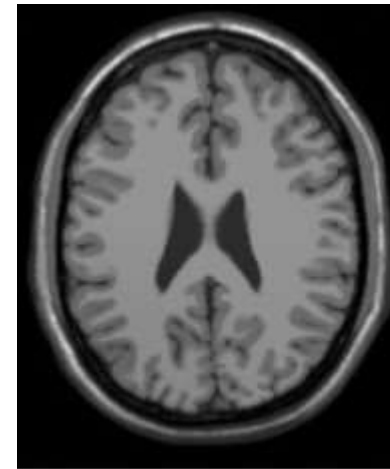
# Anwendungen der Bildregistrierung (Teil 2)

- Weitere Einsatzbereiche:
  - **Fusionierung** von Bildinhalten verschiedener Zeitpunkte oder komplementärer Modalitäten
  - **Montage** von Einzelbildern in Panoramas

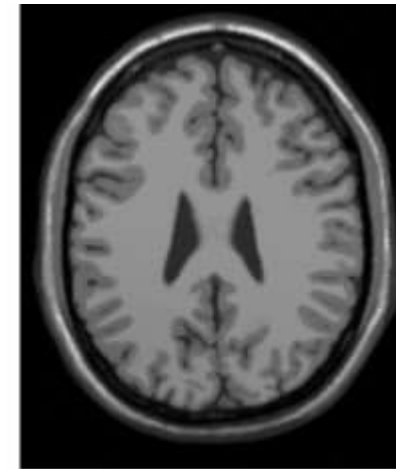


# Anwendungen der Bildregistrierung (Teil 3)

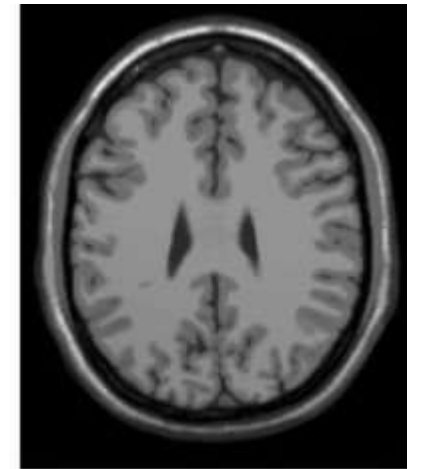
- Weiterer Einsatzbereich:
  - Strukturerhaltende **Interpolation** zwischen Bildern



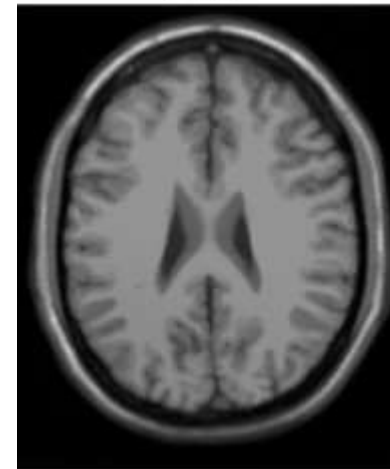
slice 96



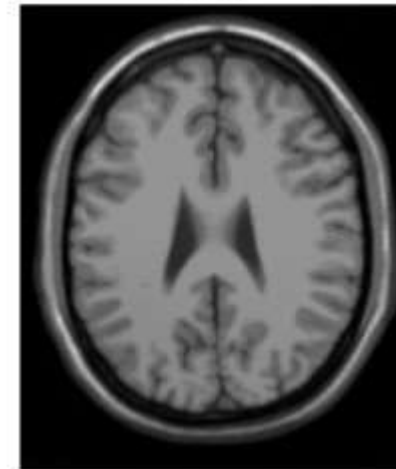
slice 98



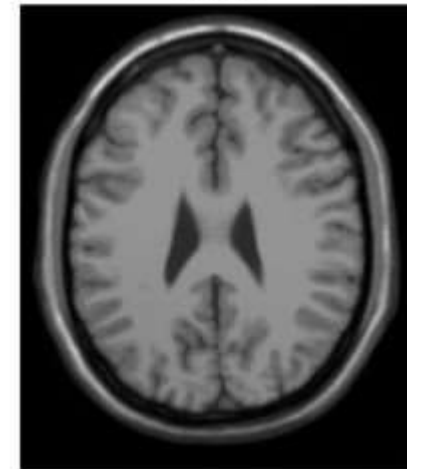
slice 100



Linear, MSD=71.7



Optical flow, MSD=45.3

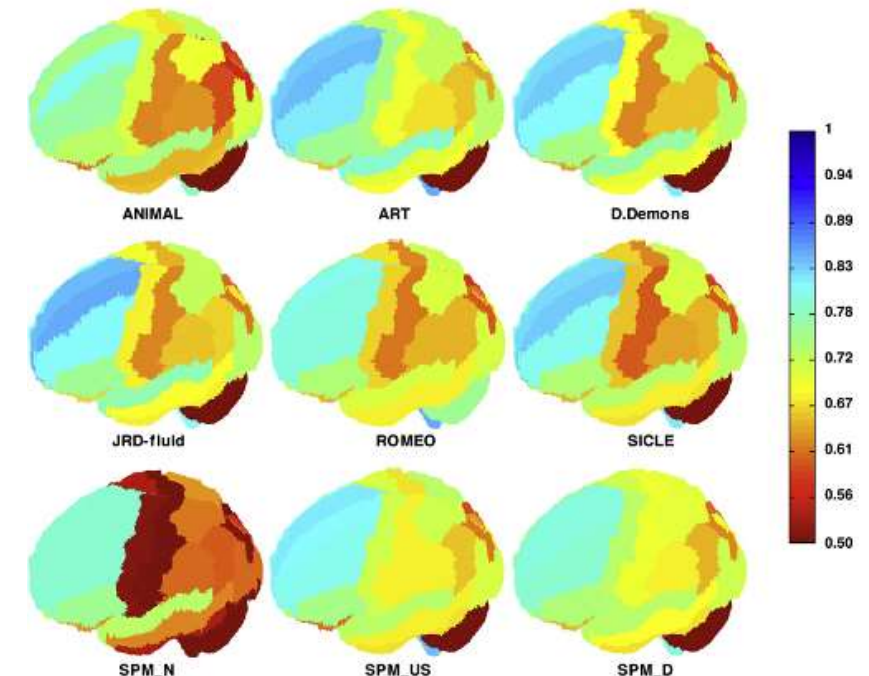


Proposed, MSD=40.7

# Registrierung vs. Segmentierung

In bestimmten Fällen besteht eine Verwandtschaft zwischen Registrierungs- und Segmentierungsproblemen:

- **Registrierung via Segmentierung**
  - Segmentierung durch statistische Formmodelle bringt Stützpunkte in Korrespondenz
- **Segmentierung via Registrierung**
  - Registrierung auf ein korrekt segmentiertes Beispiel ermöglicht eine Übertragung der Labels



# Mono- vs Multimodale Registrierung

- **Monomodale (intramodale) Registrierung**

- *Annahme*: Die korrekt registrierten Bilder haben im pixelweisen Vergleich ähnliche Intensitäten
- Setzt i.d.R. voraus, dass die beteiligten Bilder vom selben bildgebenden Verfahren stammen und mit ähnlichen Parametern (z.B. T1 oder T2-Wichtung im MRT) aufgenommen wurden

- **Multimodale (intermodale) Registrierung**

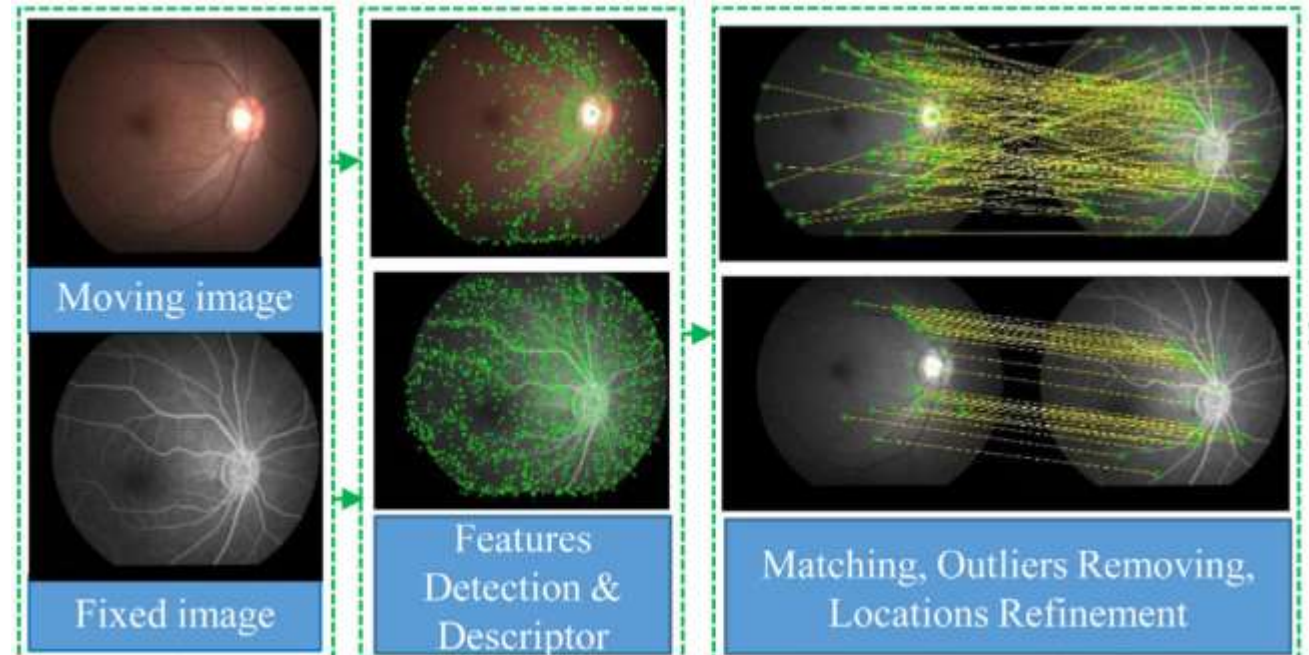
- Anatomische Strukturen sollen in Korrespondenz gebracht werden, obwohl sie i.d.R. unterschiedliche Kontraste erzeugen



# Zur Registrierung verwendete Information

Registrierungsverfahren unterscheiden sich in der **Art der verwendeten Bildinformation**:

- Manuell selektierte, durch Marker gegebene oder im Bild erkannte **Landmarken** (Punkte)
- **Kurven** oder **Oberflächen**
- **Voxel- bzw. intensitäts-**  
**basiert**





# Evaluierung mit vorgegebenen Landmarken

Sind für bestimmte **Landmarken** Positionen

- $\mathbf{r}_i$  ( $i = 1, \dots, m$ ) im Referenzbild
- $\mathbf{p}_i$  ( $i = 1, \dots, m$ ) im Objektbild

bekannt, quantifizieren der mittlere bzw. maximale *Target Registration Error*

$$\text{TRE}_{\text{mean}} = \frac{1}{m} \sum_{i=1}^m \|\mathbf{r}_i - T(\mathbf{p}_i)\|$$

$$\text{TRE}_{\text{max}} = \max\{\|\mathbf{r}_i - T(\mathbf{p}_i)\| \mid i = 1, \dots, m\}$$

deren verbleibende Abweichung nach Anwendung der von der Registrierung bestimmten Transformation  $T$

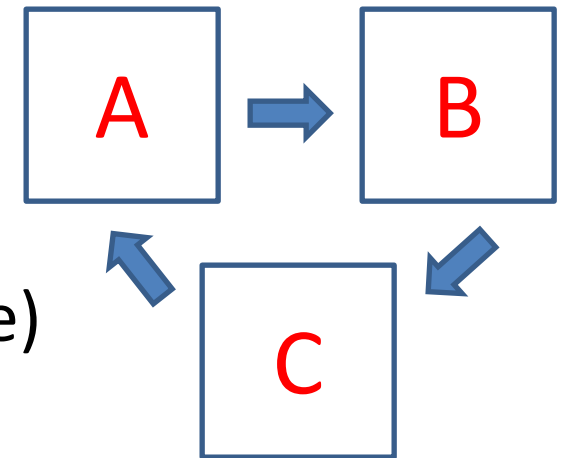
- Mögliche Herkunft der Landmarken: Annotation durch Experten oder Anwendung einer bekannten Transformation zu Testzwecken

# Weitere wünschenswerte Eigenschaften

- **Vertauschbarkeit** von Referenz- und Objektbild
  - Sollte ungefähr die inverse Transformation ergeben



- **Zyklische Transformationen**
  - Sollten ungefähr die Identität ergeben
- **Robustheit**
  - Bildstörungen (z.B. künstliches Rauschen / Artefakte) sollten die geschätzte Transformation möglichst wenig verändern



# Evaluierung durch Visualisierung

- Zur **Einschätzung von Registrierungen** visualisiert man häufig
  - Differenz- / mittlere Bilder
  - Schachbrett-Muster
  - Ausrichtung von Landmarken
  - Separate Farbkanäle

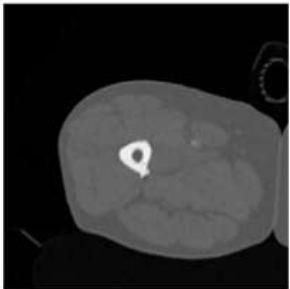


Fig 3.102: CT of a leg

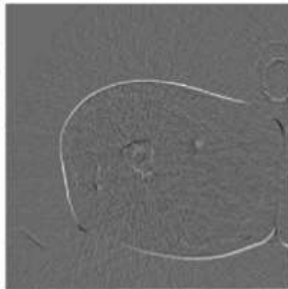


Fig 3.103: Pre-registration image subtraction

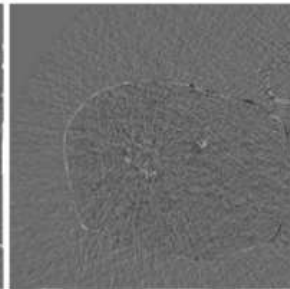
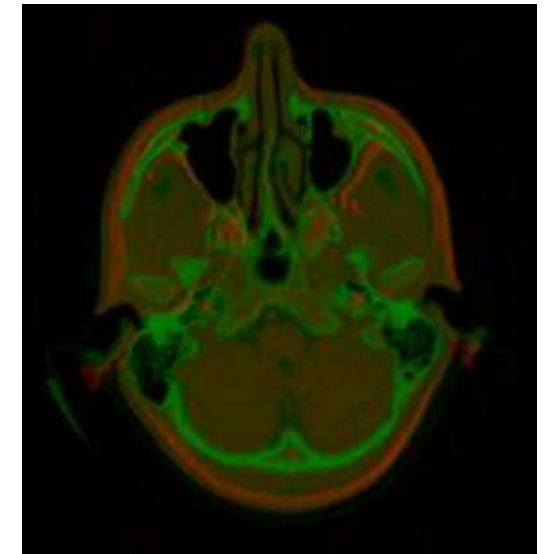
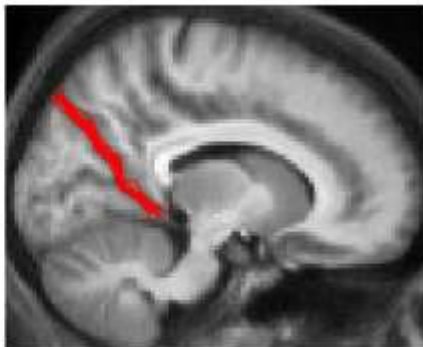
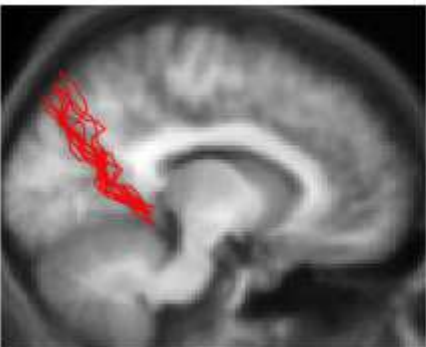


Fig 3.104: Post-registration image subtraction



# Bausteine eines typischen Registrierungsalgorithmus'

Registrierung erfordert in der Regel die Auswahl

- eines **Suchraums** erlaubter Transformationen
- eines **Interpolationsverfahren** um die Transformation anzuwenden
  - Siehe Kapitel 2
- einer **Kostenfunktion**, die Abweichungen der beiden Bilder oder ihrer relevanten Merkmale (z.B. Landmarken) quantifiziert
- eines **Optimierungsalgorithmus**, der die Kostenfunktion minimiert

## **5.2 Lineare Bildtransformationen**

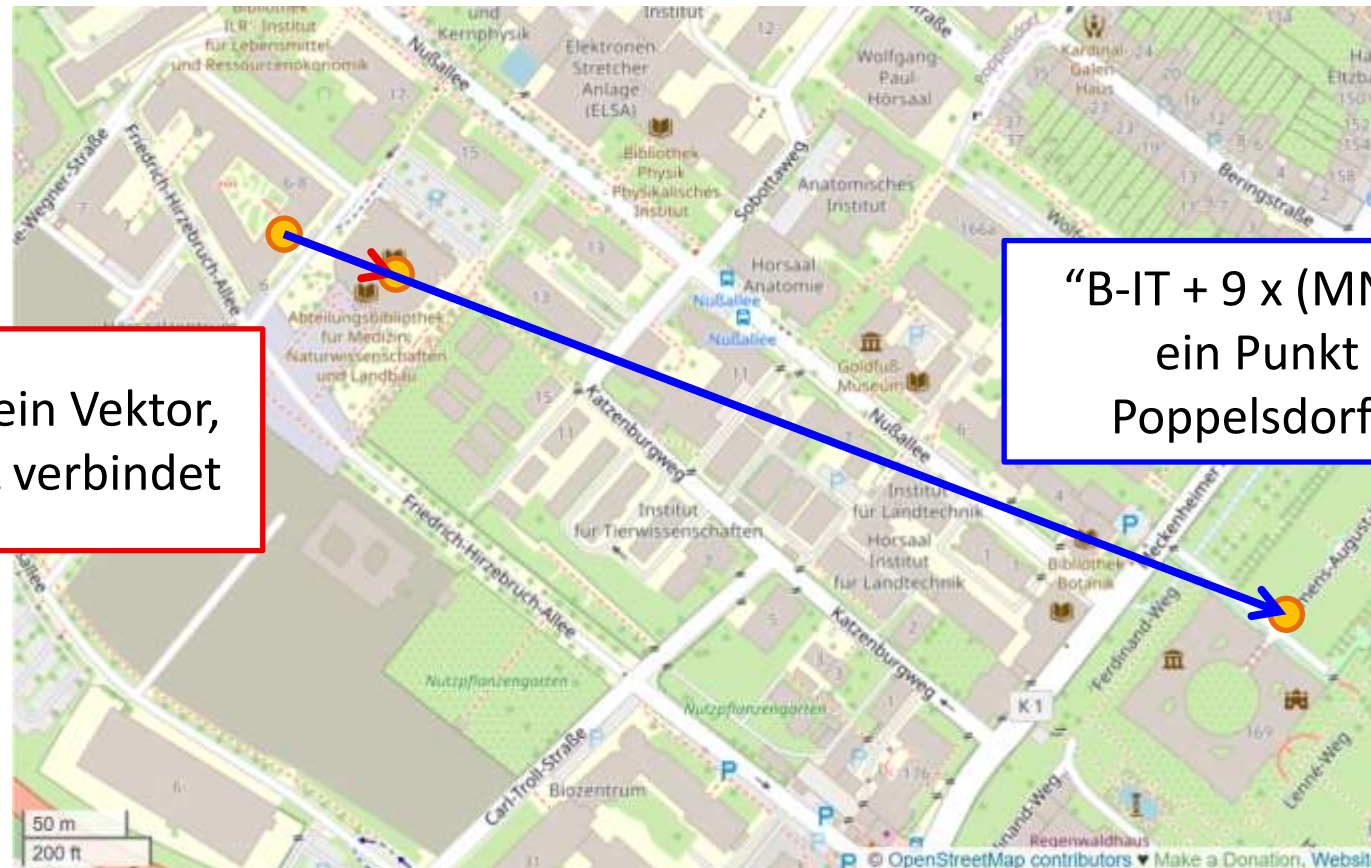
# In der Registrierung übliche Transformationen

- Je nach Art der angewandten Transformation unterscheiden wir
  - **Starre Registrierung** (*engl.* rigid registration) erlaubt nur Verschiebung und Rotation
  - **Affine Registrierung** (*engl.* affine registration) bildet parallele Linien auf parallele Linien ab
  - **Deformierbare Registrierung** (*engl.* deformable registration) erlaubt im Prinzip beliebige Deformationen, die jedoch in der Regel durch Regularisierung wieder eingeschränkt werden
    - z.B. keine gegenseitige Durchdringung von Organen, Begrenzung von Verzerrungen auf ein plausibles Maß



# Punkte

- **Punkte** in einem Bild bilden keinen sinnvollen Vektorraum
  - Was sollte “2 x B-IT” oder “MNL + B-IT” sein?



“MNL – B-IT” ist ein Vektor,  
der B-IT mit MNL verbindet

“B-IT + 9 x (MNL – B-IT)” ist  
ein Punkt (Eingang  
Poppelsdorfer Schloss)

# Punkte als Vektoren

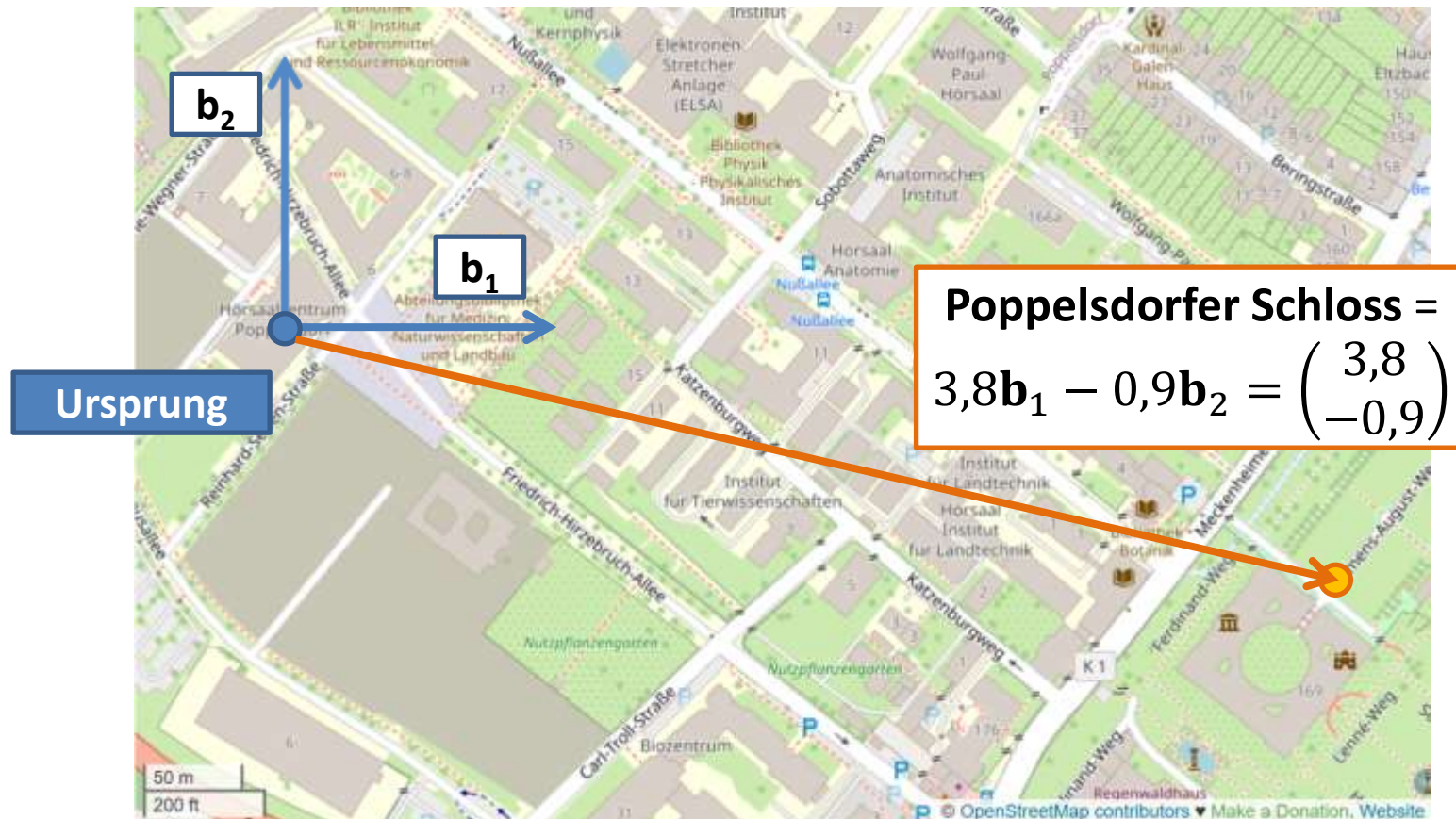
- Ein Referenzpunkt als **Ursprung** ermöglicht es mit Punkten zu arbeiten, als seien es Vektoren





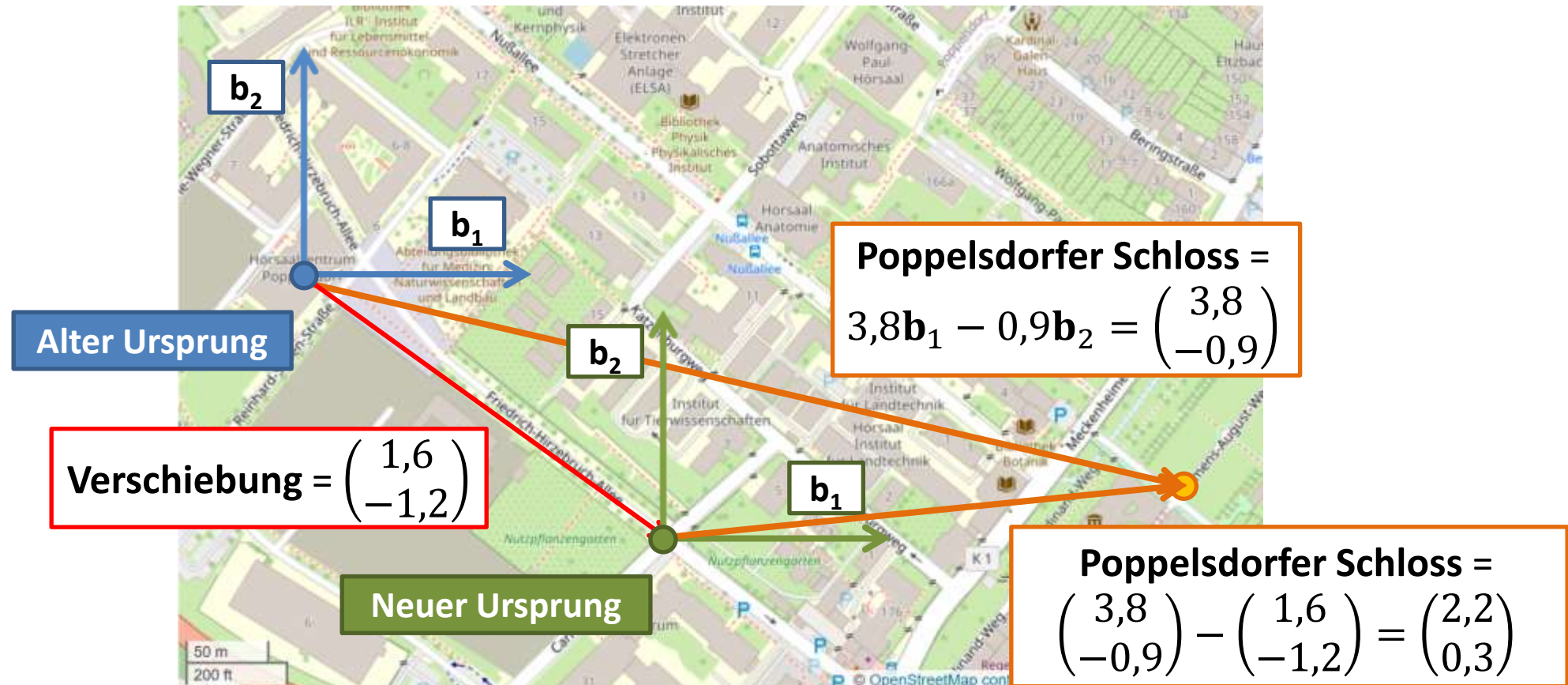
# Basen und Koordinaten

- Die Definition einer **Basis** ermöglicht die Darstellung von Vektoren durch Koeffizienten



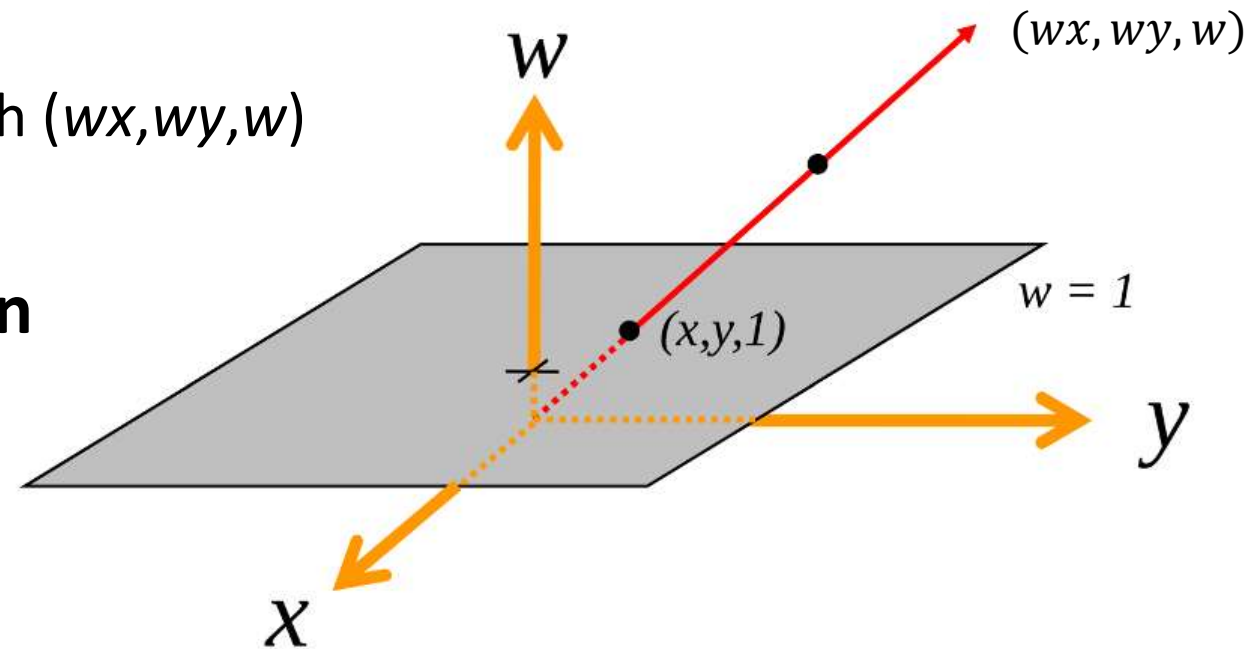
# Verschiebung des Ursprungs

- Eine **Verschiebung des Ursprungs** ändert die Koeffizienten von Punkten, nicht aber die von Vektoren zwischen Punkten



# Homogene Koordinaten

- **Homogene Koordinaten** ermöglichen uns die Unterscheidung von Punkten und Vektoren:
  - Repräsentation von  $n$ -dimensionalen **Punkten** durch Geraden in einem  $(n+1)$ -dimensionalen Raum:
    - Nenne die neue Koordinate  $w$
    - Repräsentiere den Punkt  $(x,y)$  durch  $(wx,wy,w)$
    - Kanonische Repräsentation:  $(x,y,1)$
  - Repräsentiere **Vektoren zwischen Punkten** mit  $w=0$



# Translation in Homogenen Koordinaten

- **Translation** (Parallelverschiebung)  $f(\mathbf{x}) = \mathbf{x} + \mathbf{t}$  ist eine der grundlegendsten Operationen bei der Registrierung
  - Im ursprünglichen  $n$ -D-Raum ist sie jedoch nichtlinear. Für  $\mathbf{t} \neq 0$  ist
    - $f(\mathbf{x} + \mathbf{y}) = \mathbf{x} + \mathbf{y} + \mathbf{t} \neq f(\mathbf{x}) + f(\mathbf{y}) = \mathbf{x} + \mathbf{y} + 2\mathbf{t}$
    - $f(\alpha\mathbf{x}) = \alpha\mathbf{x} + \mathbf{t} \neq \alpha f(\mathbf{x}) = \alpha\mathbf{x} + \alpha\mathbf{t}$
- **Homogene Koordinaten** ermöglichen Translationen mittels Matrix-Vektor-Multiplikation:

$$f(\mathbf{x}) = \mathbf{x} + \mathbf{t} = \begin{pmatrix} 1 & 0 & t_x \\ 0 & 1 & t_y \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x_x \\ x_y \\ 1 \end{pmatrix}$$

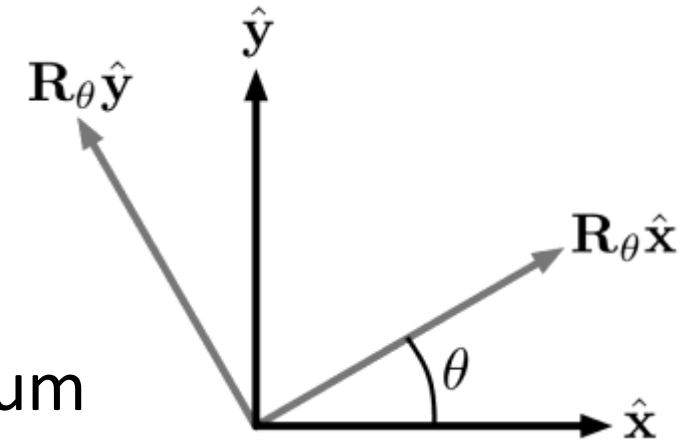
- *Hinweis:* In Übereinstimmung mit der Anschauung bleiben Vektoren ( $w = 0$ ) von Translationen unberührt



# Rotationen: Grundidee

- **Rotationen** wirken immer in einer Ebene, der Orthogonalraum bleibt unberührt

- in 2D: Rotation des ganzen Bildes
- in 3D: Rotation um eine Rotationsachse
- Rotation um den Ursprung. Vorgeschaltete Translation ermöglicht alternatives Rotationszentrum



- Skizze zeigt Auswirkung der Rotation um  $\theta$  in der von orthonormalen Vektoren  $\hat{x}$  and  $\hat{y}$  aufgespannten Ebene

- *Konvention*: Positive  $\theta$  rotieren  $\hat{x}$  auf  $\hat{y}$  zu

$$R_\theta \hat{x} = \cos \theta \hat{x} + \sin \theta \hat{y}$$

$$R_\theta \hat{y} = -\sin \theta \hat{x} + \cos \theta \hat{y}$$

$$R_\theta \mathbf{a} = \mathbf{a}.$$

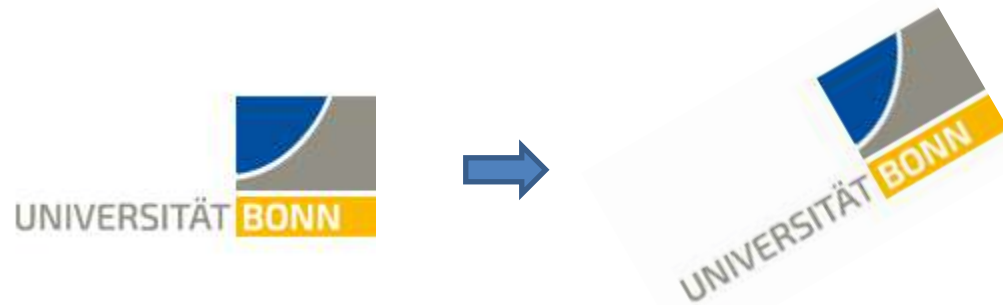
# Rotationen: Matrix-Schreibweise

- Aufgrund der Überlegungen der vorherigen Folie

$\mathbf{R}_\theta \hat{\mathbf{x}} = \cos \theta \hat{\mathbf{x}} + \sin \theta \hat{\mathbf{y}} ; \mathbf{R}_\theta \hat{\mathbf{y}} = -\sin \theta \hat{\mathbf{x}} + \cos \theta \hat{\mathbf{y}} ; \mathbf{R}_\theta \mathbf{a} = \mathbf{a}$   
erhalten wir:

$$\mathbf{R}_\theta = \mathbf{I} + (\cos \theta - 1)(\hat{\mathbf{x}}\hat{\mathbf{x}}^T + \hat{\mathbf{y}}\hat{\mathbf{y}}^T) + \sin \theta (\hat{\mathbf{y}}\hat{\mathbf{x}}^T - \hat{\mathbf{x}}\hat{\mathbf{y}}^T)$$

- Beispiel mit  $\hat{\mathbf{x}} = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}, \hat{\mathbf{y}} = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} : \mathbf{R}_\theta = \begin{bmatrix} \cos \theta & -\sin \theta & 0 \\ \sin \theta & \cos \theta & 0 \\ 0 & 0 & 1 \end{bmatrix}$



# Ähnlichkeitsabbildungen

- Die skalierte Einheitsmatrix  $\alpha \mathbf{I}$  entspricht einer **uniformen Skalierung** um den Faktor  $\alpha$ 
  - **Ähnlichkeitsabbildungen**  $f(\mathbf{x}) = \alpha \mathbf{R}\mathbf{x} + \mathbf{t}$  erhalten Formen, d.h. Winkel und relative Abstände
  - In der Registrierung ermöglichen sie z.B. den Ausgleich von Auflösungsunterschieden



# Affine Abbildungen

- **Affine Abbildungen** haben die Form  $f(\mathbf{x})=\mathbf{M}\mathbf{x}+\mathbf{t}$ 
  - Die Matrix **M** darf beliebig sein
    - *Zusätzlich*: Projektion, nicht-uniforme Skalierung, Spiegelung, Scherung
  - Bildet Geraden auf Geraden oder Punkte ab, Parallelität bleibt erhalten, Teilverhältnisse auf Geraden bleiben erhalten
- Homogene Koordinaten ermöglichen die Schreibweise

$$\mathbf{M} \quad \begin{pmatrix} a & b & c & tx \\ d & e & f & ty \\ g & h & i & tz \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad \mathbf{t}$$

The diagram illustrates the relationship between the affine transformation  $f(\mathbf{x}) = \mathbf{M}\mathbf{x} + \mathbf{t}$  and its representation in homogeneous coordinates. A bracket labeled **M** spans the first three columns of the 4x4 matrix, which are labeled  $a, b, c$  in the first row. Another bracket labeled **t** spans the fourth column, which contains  $tx, ty, tz$  in the first three rows. The bottom row of the matrix is  $0, 0, 0, 1$ .

- „Affine“ und „lineare“ Registrierung werden häufig synonym verwendet

# Orthogonalprojektionen

- **Projektionen** spielen in der Registrierung in der Regel nur eine mittelbare Rolle als Bausteine anderer Transformationen
- **Orthogonalprojektion** auf den von  $\mathbf{x}$  aufgespannten Unterraum:

$$\mathbf{P}_{\mathbf{x}} = \frac{\mathbf{x}\mathbf{x}^T}{\mathbf{x}^T\mathbf{x}}$$

$$\mathbf{x} = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} \Rightarrow \mathbf{P}_{\mathbf{x}} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

- Orthogonalprojektion auf den **Komplementärraum**:  $\mathbf{P}_{\mathbf{x}}^{\perp} = \mathbf{I} - \mathbf{P}_{\mathbf{x}}$

$$\mathbf{x} = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} \Rightarrow \mathbf{P}_{\mathbf{x}}^{\perp} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

# Nicht-uniforme Skalierung

- **Nicht-uniforme Skalierung** um den Faktor  $\alpha$  entlang der Richtung  $\mathbf{x}$

- basiert auf der Zerlegung

$$\mathbf{v} = \mathbf{I}\mathbf{v} = \mathbf{P}_{\mathbf{x}}\mathbf{v} + \mathbf{P}_{\mathbf{x}}^{\perp}\mathbf{v}$$

- Resultierende Matrix:

$$\mathbf{S}_{\alpha,\mathbf{x}} = \alpha\mathbf{P}_{\mathbf{x}} + \mathbf{P}_{\mathbf{x}}^{\perp}$$

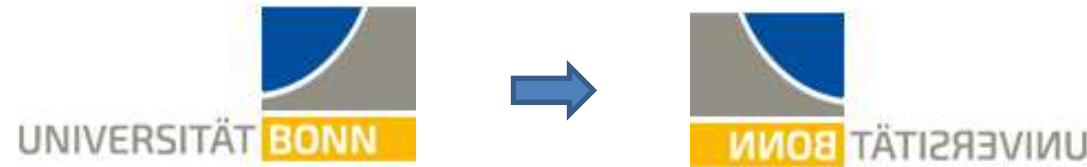
- In der Registrierung ermöglicht sie z.B. den Ausgleich von richtungsabhängiger (anisotroper) Auflösung





# Spiegelungen und Scherungen

- **Spiegelung** entspricht einer nicht-uniformen Skalierung mit Faktor  $\alpha = -1$ :  $\mathbf{M}_{\hat{\mathbf{x}}} = \mathbf{P}_{\hat{\mathbf{x}}}^{\perp} - \mathbf{P}_{\hat{\mathbf{x}}}$



- **Scherung** (Transvektion): Verschiebung parallel zu einer festen Achse, proportional (mit Scherfaktor  $\beta$ ) zur Position entlang einer orthogonalen Achse

$$\mathbf{T} = \mathbf{I} + \beta \hat{\mathbf{x}} \hat{\mathbf{y}}^T$$



# Berechnung der Transformationen aus Korrespondenzen

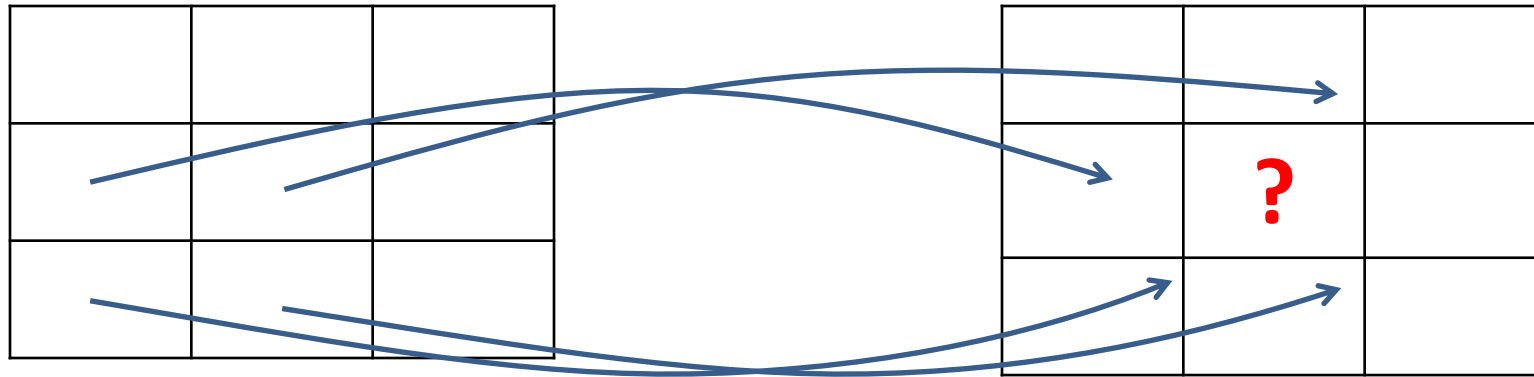
- Seien wieder  $\mathbf{p}_i$  ( $i = 1, \dots, m$ ) Punkte im Objektbild,  $\mathbf{r}_i$  ( $i = 1, \dots, m$ ) korrespondierende Punkte im Referenzbild
- Gesucht sei Transformation  $T$  die zu kleinsten Quadraten führt:

$$T = \arg \min_{T'} \sum_{i=1}^m \|T'(\mathbf{p}_i) - \mathbf{r}_i\|^2$$

- **Starre** oder **Ähnlichkeits-Transformationen**  $T$  erhalten wir mittels Prokrustes-Analyse (Kapitel 4.5)
- **Affine Transformationen**  $T$  ergeben sich durch lineares Ausgleichsproblem mit den Koeffizienten der Transformationsmatrix als Unbekannten (Übung)

# Nachteil der Vorwärts-Transformation

- Anwendung der **Vorwärts-Transformation**  $T$  erzeugt das transformierte Bild durch Iteration über das Ursprungsbild
  - Manche Pixel des erzeugten Bildes erhalten mehrere Farbwerte, andere keins
  - Hinterlässt ohne weiteres Auffüllen „Löcher“ im Resultat



# Lösung: Nutzen der Rückwärts-Transformation

- Berechnung der inversen **(Rückwärts)-Transformation**  $T^{-1}$  ermöglicht Berechnung des transformierten Bildes durch Iteration über die Ausgabepixel
  - Jedes Pixel erhält genau einen Farbwert
  - Interpolation im Ursprungsbild mit üblichen Verfahren
  - Bei affinen Abbildungen: Berechnung von  $T^{-1}$  als inverse Matrix



# Bild- vs. Weltkoordinaten

Bei der Arbeit mit medizinischen Bildern unterscheiden wir zwischen

- **Bildkoordinaten:** Indizieren Pixel/Voxel
  - Bild als Matrix: rc-Koordinaten (row/column)
    - Vertikale Position zuerst, Ursprung links oben, größere Zeilen weiter unten
  - Bild als Funktion: xy-Koordinaten
    - Horizontale Position zuerst, Ursprung links unten, größeres y weiter oben
- **Weltkoordinaten:** Physisches Koordinatensystem (XYZ), in dem der Patient verortet ist. Bei uniformen Voxelgittern:

$$\begin{pmatrix} X \\ Y \\ Z \end{pmatrix} = \begin{pmatrix} X_0 \\ Y_0 \\ Z_0 \end{pmatrix} + x \begin{pmatrix} X_x \\ Y_x \\ Z_x \end{pmatrix} + y \begin{pmatrix} X_y \\ Y_y \\ Z_y \end{pmatrix} + z \begin{pmatrix} X_z \\ Y_z \\ Z_z \end{pmatrix}$$

# Bild- zu Weltkoordinaten als Affine Abbildung

- In homogenen Koordinaten lässt sich der Zusammenhang zwischen Bild- und Weltkoordinaten durch eine Matrix beschreiben:

$$\begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix} = \begin{pmatrix} X_x & X_y & X_z & X_0 \\ Y_x & Y_y & Y_z & Y_0 \\ Z_x & Z_y & Z_z & Z_0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix}$$

- Bilder aus Hybridgeräten (z.B. PET/CT) sind häufig intrinsisch registriert. Korrespondenzen sind durch die Bild-zu-Welt-Transformationen gegeben bzw. können durch Matrix-Invertierung und –Multiplikation leicht berechnet werden



# Zusammenfassung

- Anwendung „**linearer Bildtransformationen**“ geht von der Nutzung **homogener Koordinaten** aus
  - Hierdurch werden auch Translationen abgedeckt
  - **Starre Abbildungen**: Translation und Rotation
  - **Ähnlichkeits-Abbildungen**: Starr+uniforme Skalierung
  - **Affine Abbildungen**: Ähnlichkeit+nicht uniforme Skalierung, Scherung, Spiegelung
- Erzeugung registrierter Bilder durch **Rückwärts-Transformation**
- Unterscheidung von **Bild-** und **Weltkoordinaten**

## **5.3 Registrierung als Optimierungsproblem**

# Registrierung als Optimierungsproblem

- Gegeben seien
  - Referenzbild  $\mathbf{y}$
  - Objektbild  $\mathbf{x}$
  - Kostenfunktion  $C$
  - Suchraum  $S_T$  von Transformationen, parametrisiert durch  $\mathbf{w}$
- Folgendes Optimierungsproblem bestimmt die Parameter der optimalen Transformation  $T$ :

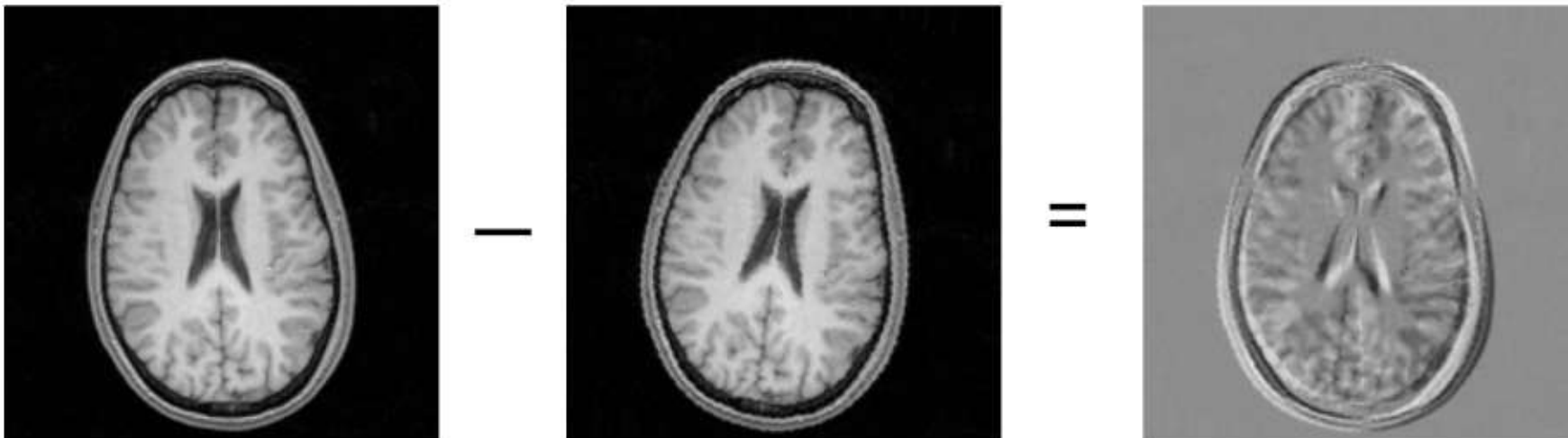
$$\mathbf{w} = \arg \min_{\mathbf{w}'} C(\mathbf{y}, T(\mathbf{x}|\mathbf{w}'))$$

# Kleinste Quadrate (L2-Norm) als Kostenfunktion

- Eine einfache Kostenfunktion zur voxelbasierten Registrierung ist durch die **mittlere quadratische Abweichung** über die  $N$  überlappenden Pixel gegeben:

$$C = \frac{1}{N} \sum_{i=1}^N (x_i - y_i)^2$$

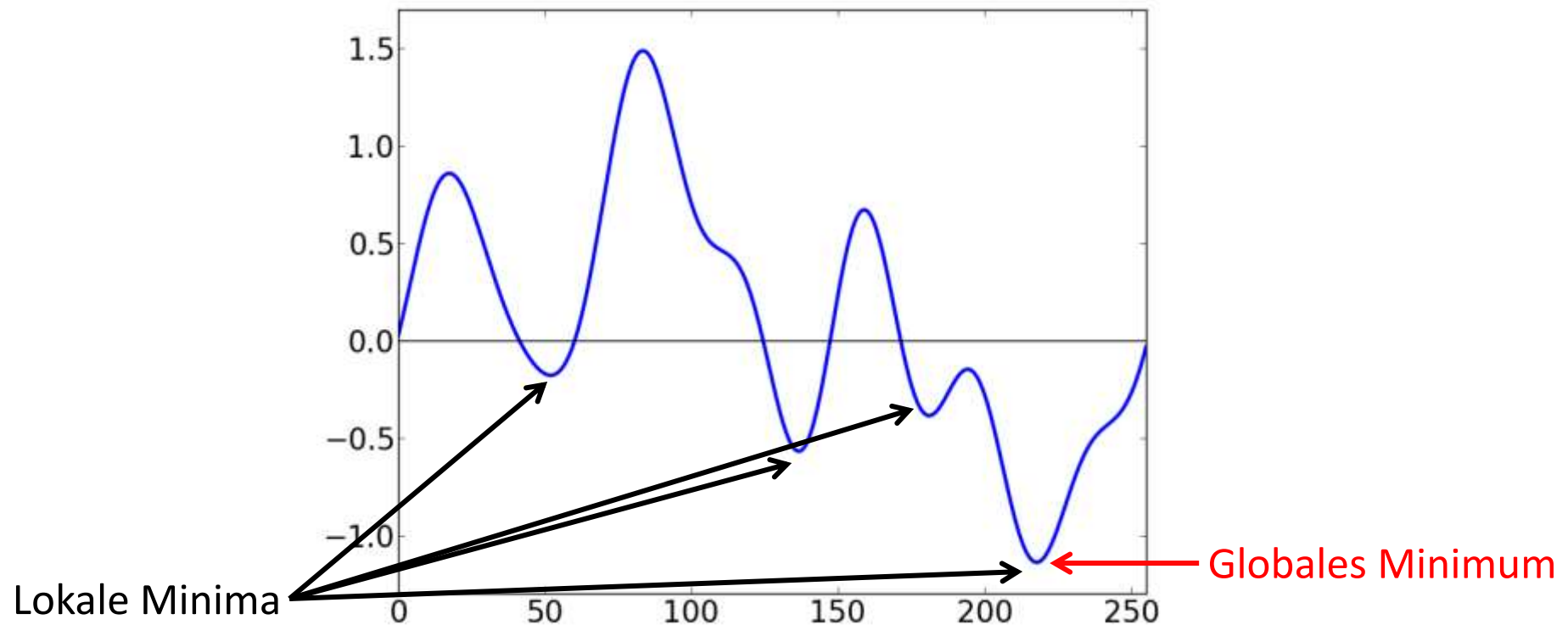
- *Quiz:* Was ist eine Einschränkung dieser Kostenfunktion?



Bildquelle: Oxford FMRIB

# Ziel von Optimierungsverfahren

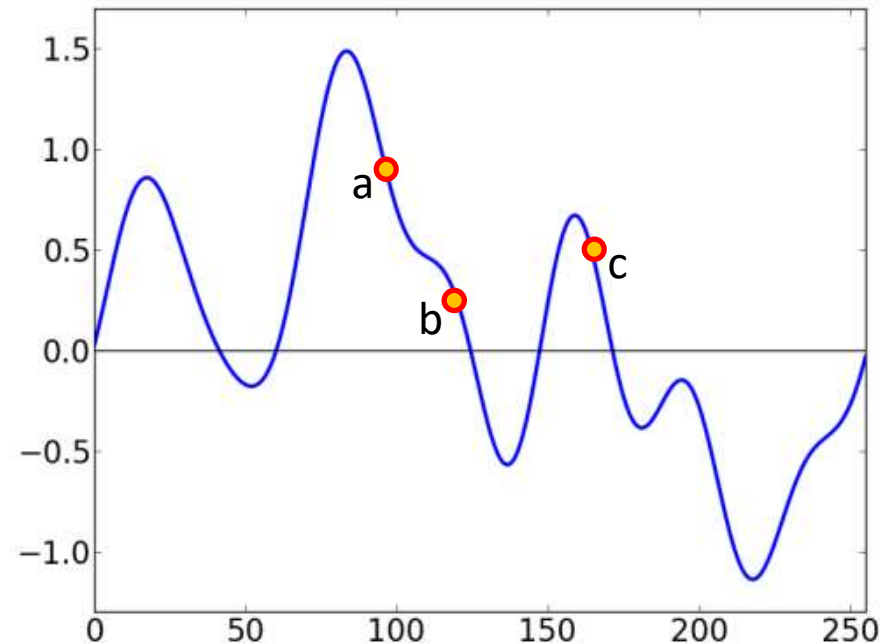
- *Angestrebt*: algorithmische Minimierung einer Funktion  $f(x)$ 
  - Annahme:  $f(x)$  ist stetig



# 1D-Fall: „Einfangen“ von Minima durch Intervalle

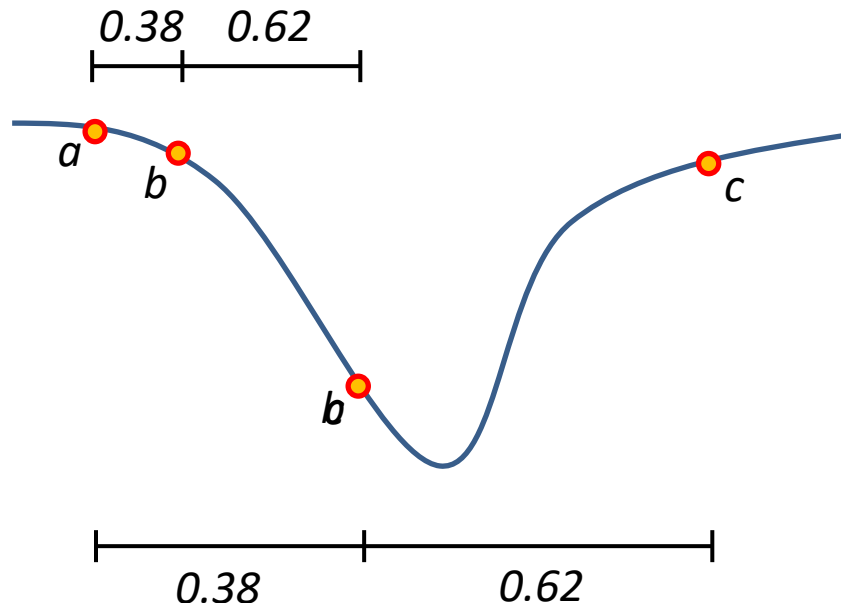
- Eine kontinuierliche Funktion  $f(x)$  hat im Intervall  $(a,c)$  mindestens ein lokales Minimum, falls es darin einen inneren Punkt  $b$  gibt, für den gilt:

$$f(a) > f(b) \text{ und } f(c) > f(b)$$



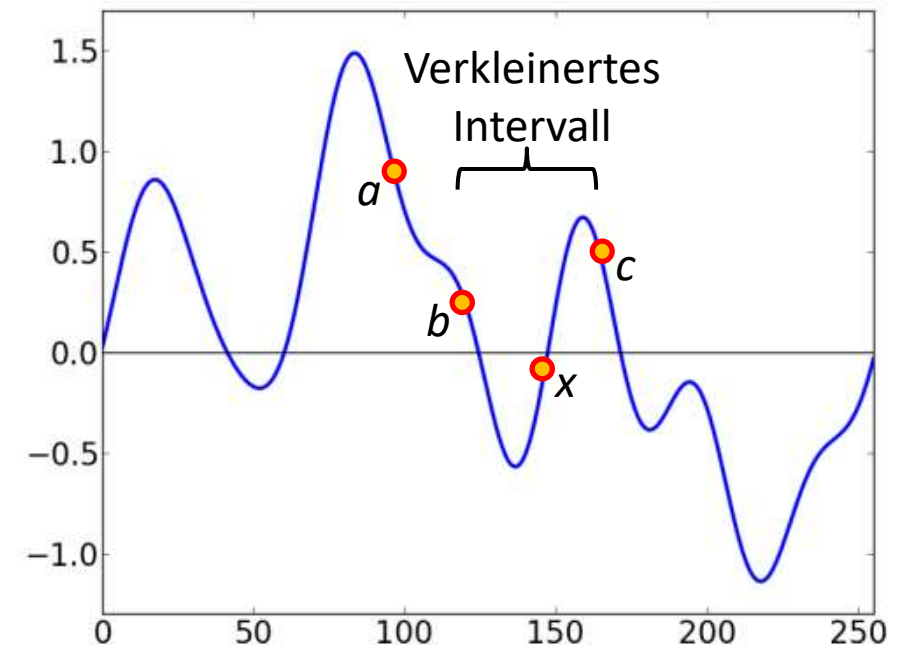
# Finden eines initialen Intervalls

- Ausgehend von einem ersten Intervall  $(a,b)$  mit  $f(a) > f(b)$  vergrößern wir es schrittweise über den kleineren Wert hinaus, bis die Bedingung der letzten Folie erfüllt ist
  - Minimierung scheitert, wenn das Intervall zu groß wird oder die Grenze der zulässigen Werte erreicht



# Iterative Verkleinerung des Intervalls

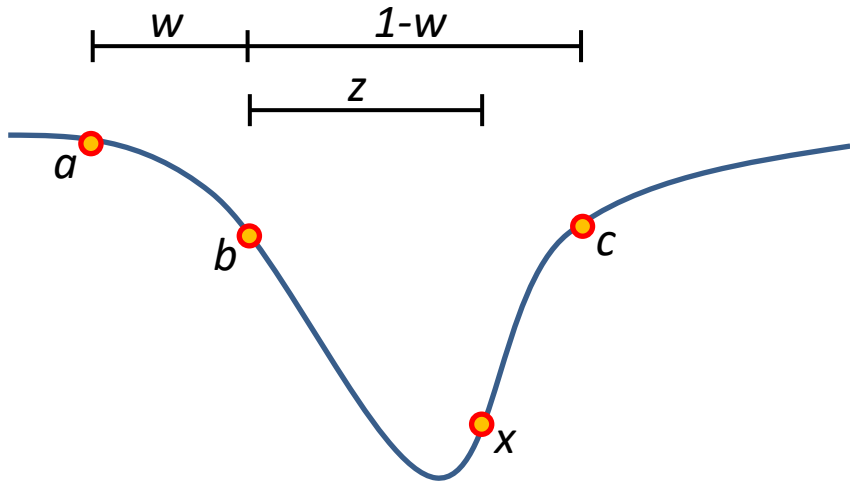
- **Grundidee:** Werte  $f(x)$  im längeren der beiden Intervalle  $(a, b)$  und  $(b, c)$  aus. O.B.d.A. sei dies  $(b, c)$ 
  - Falls  $f(x) < f(b)$  befindet sich mindestens ein lokales Minimum in  $(b, c)$
  - Falls  $f(x) > f(b)$  befindet sich mindestens ein lokales Minimum in  $(a, x)$
  - In jedem Fall haben wir das Intervall verkleinert. Benenne die neuen drei Punkte in  $(a, b, c)$  um und iteriere so lange, bis  $|c - a| < \theta$





# An welchem Punkt verfeinern?

- **Frage:** Welcher neue Punkt  $x$  führt für vorgegebene  $(a,b,c)$  im nächsten Schritt zum kürzesten Intervall?



$$w = \frac{b - a}{c - a} \quad z = \frac{x - b}{c - a}$$

Abhängig von  $f(x)$  ist das neue Intervall  $1-w$  oder  $w+z$  breit. Um in beiden Fällen denselben Fortschritt zu erzielen setzen wir:

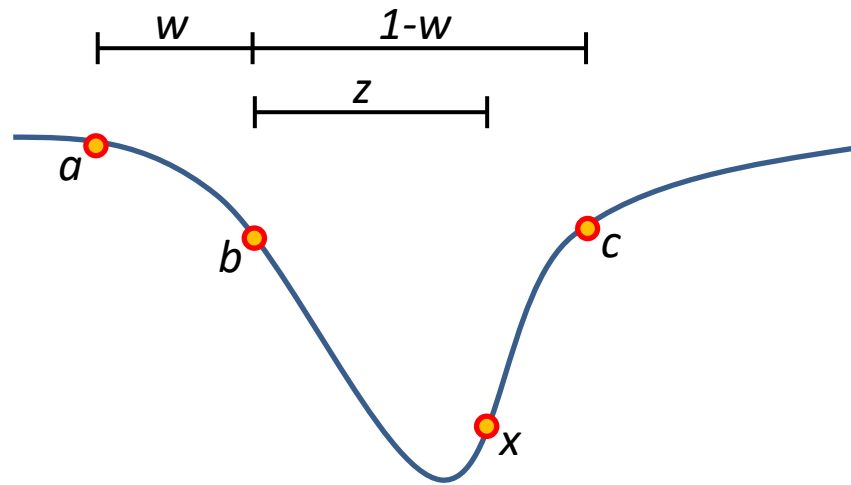
$$1 - w = w + z$$

$$\Rightarrow z = 1 - 2w \quad (\text{Gl. 1})$$

- **Antwort:**  $x$  sollte von  $c$  denselben Abstand haben wie  $b$  von  $a$ .

# Wo liegt der optimale innere Punkt?

- **Frage:** Mit welchem inneren Punkt  $b$  macht die Regel auf der letzten Folie für gegebenes  $(a,c)$  konstanten Fortschritt?



**Idee:** Um konstanten Fortschritt zu machen, muss die relative Position von  $x$  im neuen Intervall der von  $b$  im ursprünglichen entsprechen

$$\frac{z}{1-w} = w \quad (\text{Gl. 2})$$

- Kombination mit Gl. 1 ( $z = 1 - 2w$ ) ergibt:  $w^2 - 3w + 1 = 0$
- Die einzige Lösung in  $(0,1)$  ist  $w = \frac{3-\sqrt{5}}{2} \approx 0.38197$
- **Antwort:**  $b$  sollte dem „goldenen Schnitt“ von  $(a,c)$  entsprechen

# Randbemerkung: Der goldene Schnitt in der Kunst

In der Renaissance war der „goldene Schnitt“  $\phi := \frac{1}{1-w} = \frac{1-w}{w} = \frac{1+\sqrt{5}}{2}$  auch als „göttliche Proportion“ bekannt



# Algorithmus: Suche mit dem goldenen Schnitt

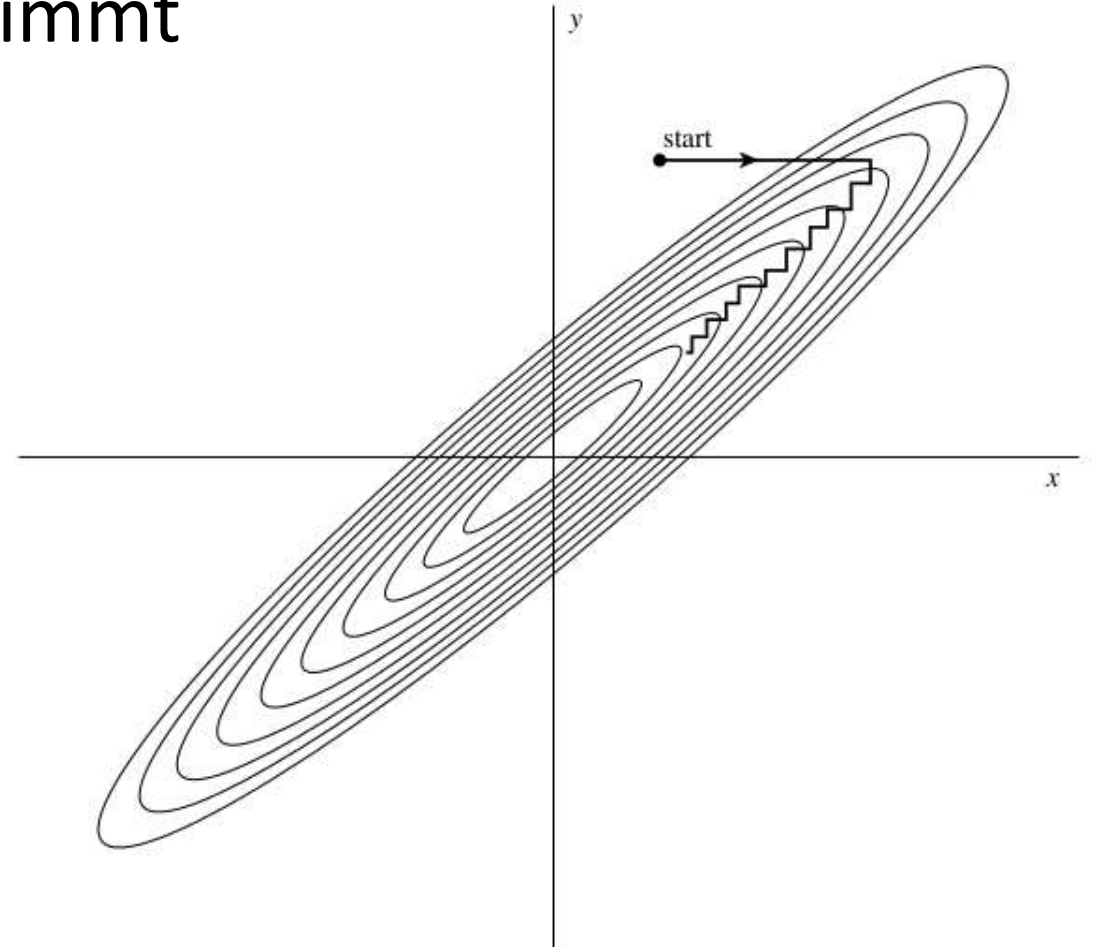
- **Gegeben:** Intervall  $(a,c)$  mit innerem Punkt  $b$
- **Iteration:**
  - Berechne  $f(x)$  für einen Punkt  $x$  der das größere der Intervalle  $(a,b)$  oder  $(b,c)$  im Verhältnis 0.38:0.62 teilt (zu  $b$  hin)
  - Wähle in Abhängigkeit von  $f(x)$  das linke oder rechte Intervall
    - Das, wo der innere Punkt unter den Rändern liegt
- **Beobachtung:** Selbst wenn  $b$  ursprünglich  $(a,c)$  nicht im goldenen Schnitt teilt, erzeugt diese Vorgehensweise i.d.R. innerhalb weniger Iterationen Tripel im goldenen Schnitt
  - Ab diesem Punkt verkürzt jede Iteration das Intervall um den Faktor 0.62
  - „Lineare Konvergenz“: Zahl der korrekten Nachkommastellen wächst linear mit der Zahl der Iterationen

# Optimierung in höheren Dimensionen

- Lineare Registrierung in 3D hat einen 6- (starr) bis 12-dimensionalen (affin) Suchraum
  - Die Suche mit dem goldenen Schnitt lässt sich nicht in höhere Dimensionen verallgemeinern
  - *Statt dessen:* Alternierende 1D-Optimierungen entlang einer Menge von Suchrichtungen
    - *Beispiel:*  $N$  Basis-Vektoren des  $N$ -D-Suchraums
    - *In der Registrierung:* Translation, Rotation, Skalierung, Scherung
  - Iteriere wiederholt über die  $N$  Richtungen, bis zur Konvergenz
  - Funktioniert dann gut, wenn Richtungen „entkoppelt“ sind

# Ein Problem für alternierende 1D-Ansätze

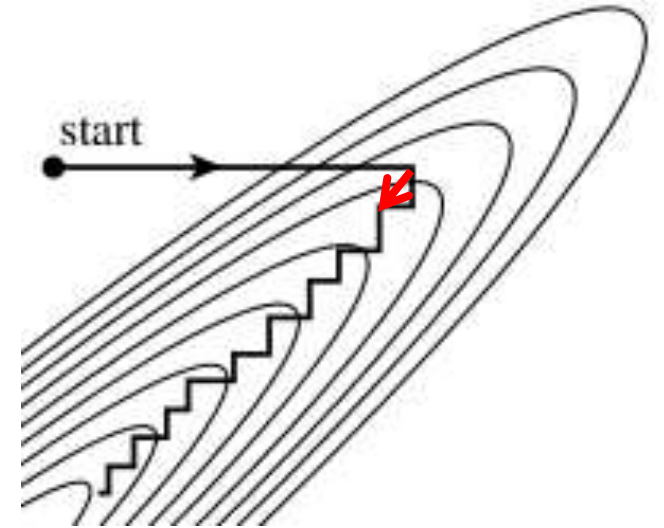
- **Problem:** Alternierende 1D-Optimierung wird sehr langsam, wenn die Zielfunktion ein schmales Tal aufweist, das mit den Suchrichtungen nicht übereinstimmt
  - Zwingt die Methode zu einer großen Zahl kleiner Schritte



# Verfahren von Powell

- **Grundidee:**

- Iteriere vom Startpunkt  $P_0$  aus  $N$  1D-Optimierungen
- Jede davon ende in einem neuen Punkt  $P_i$
- Nutze nach  $N$  1D-Optimierungen  $P_N - P_0$  als neue Suchrichtung
  - Richtung des Gesamtfortschritts
  - Hoffnung: „weist entlang des Tals“



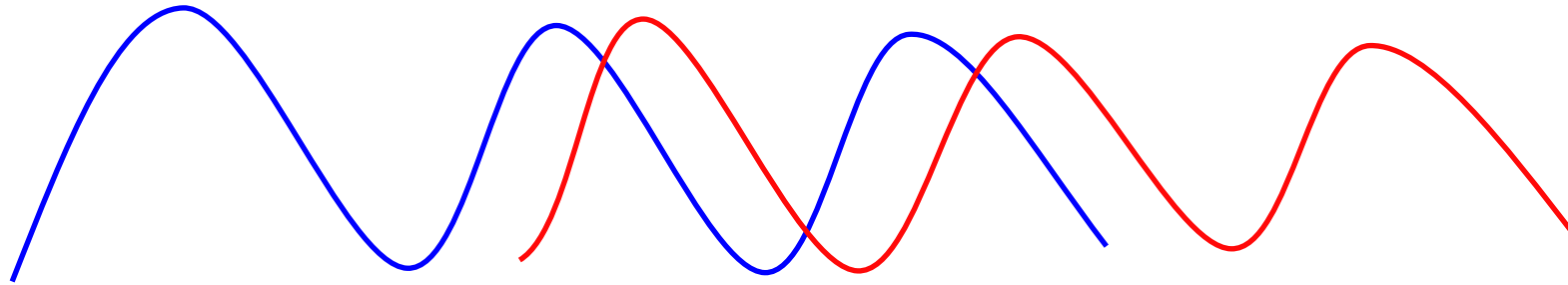
- **Varianten:**

- Ersetze die bislang erste Suchrichtung durch die neue Richtung
  - *Aber:* Richtungen können degenerieren (lineare Abhängigkeit)
- Ersetze die Suchrichtung mit dem größten Beitrag
  - *Idee:* Ersetzen einer möglichst kollinearen Richtung
- Periodische Orthogonalisierung der Suchrichtungen

# Lokale und Globale Optimierung

- **Lokale Optimierung:**

- Findet ein lokales Minimum in der Nachbarschaft des Punkts  $x$
- *Aber:* Suboptima sind aufgrund von Bildinhalten oft unvermeidlich



- **Globale Optimierung:**

- Ohne weitere Einschränkungen von  $f(x)$  können wir idR nicht darauf hoffen garantiert das globale Optimum zu finden
- Heuristiken sind oft ausreichend um sinnvolle Optima zu finden
- Manuelle Initialisierung / Eingreifen bleibt manchmal notwendig

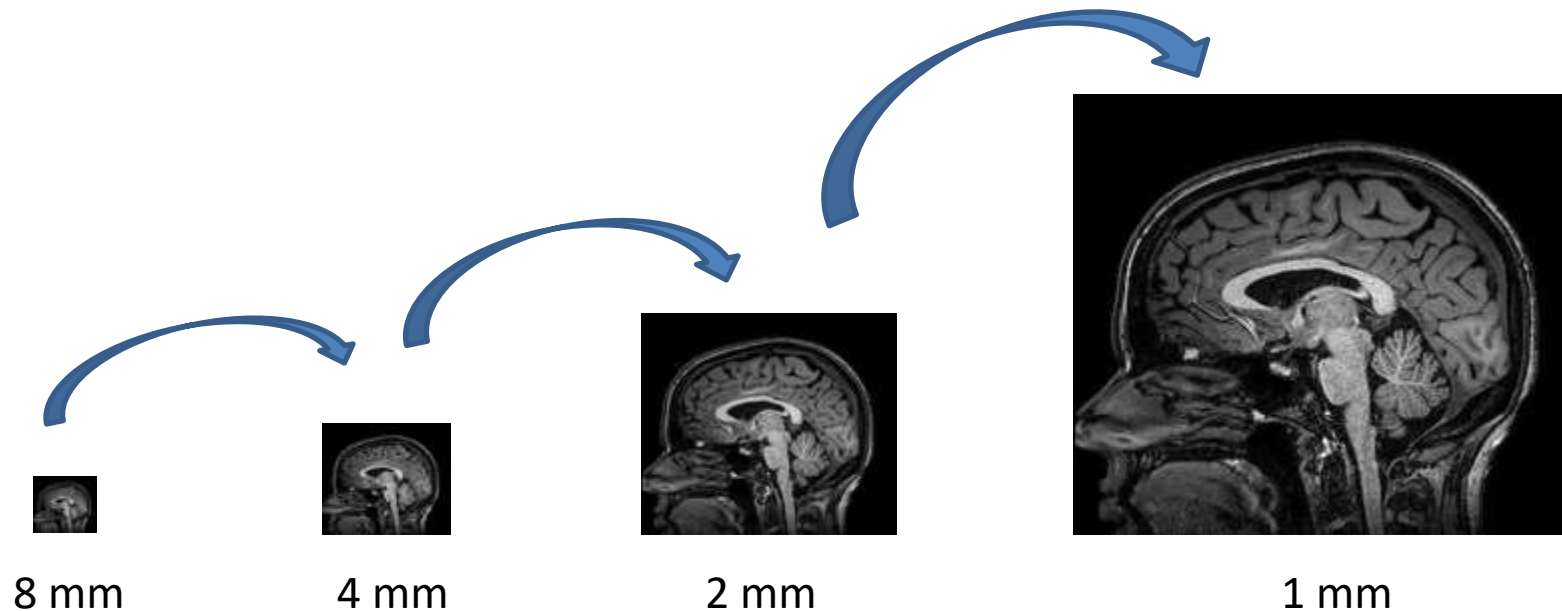


# Rastersuche

- **Idee:** Systematisches Ausprobieren verschiedener Initialisierungen
  - *Beispiel:* Je 10 Werte innerhalb eines sinnvollen Bereichs
- **Vorsicht:** Kombinatorische Explosion in höheren Dimensionen
  - *Beispiel:* Affine Transformation in 3D:  $10^{12}$  Kombinationen
  - Selbst wenn wir die Kostenfunktion in einer Zehntelsekunde auswerten können verbringen wir damit 3000 Jahre!
- **Dennoch:** Nützliche Idee für leicht zu berechnende Kostenfunktionen und niedrigdimensionale Suchräume

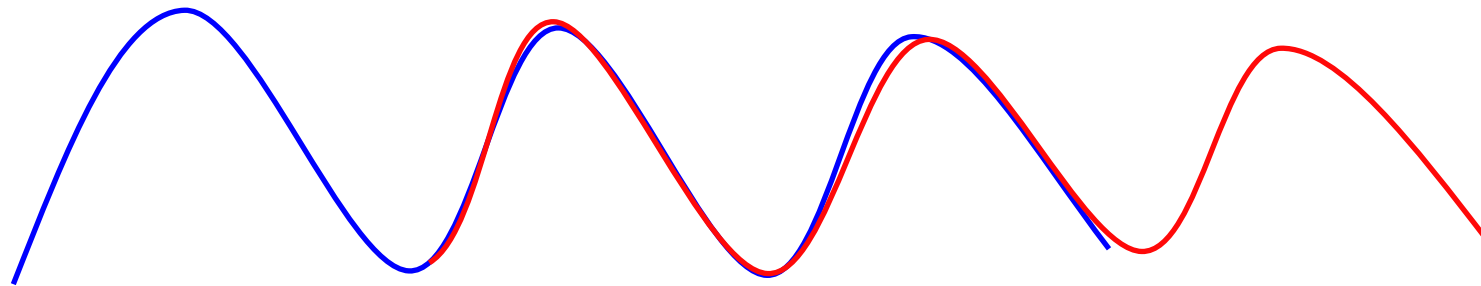
# Optimierung auf mehreren Skalen

- **Idee:** Registriere zunächst auf einer groben Skala, verfeinere dann iterativ auf detaillierteren Skalen. *Vorteile:*
  - Kostenfunktion schnell zu berechnen
  - Konvergenz in wenigen Schritten
  - Eliminierung feiner Strukturen reduziert Probleme mit Suboptima



# Perturbation und Neustart

- Zufällige Perturbation einer Lösung und Neustart der Optimierung kann aus lokalen Optima heraushelfen
  - *Aber*: Könnte natürlich auch in einem noch schlechteren Suboptimum enden
  - *Daher*: Vergleich zu vorherigem Optimum, Auswahl des besseren



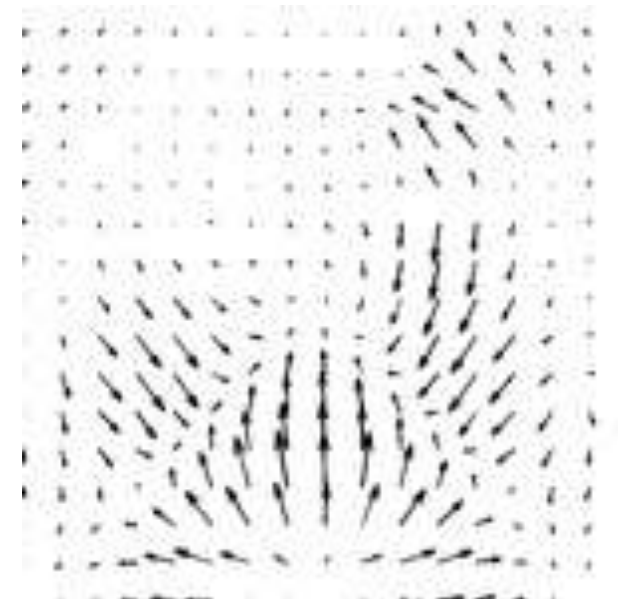
# Zusammenfassung: Registrierung per Optimierung

- Die Norm des Differenzbildes ist eine einfache **Kostenfunktion** zur intensitätsbasierten Bildregistrierung
- Optimierung stetiger 1D-Funktionen per **Suche mit dem goldenen Schnitt**
  - *Grundidee*: Iterative Verkürzung eines Suchintervalls
- **Alternierende 1D-Optimierung** als Grundidee für höhere Dimensionen
  - Verfahren von Powell findet geeignete Suchrichtungen
- Bessere Ergebnisse in der Praxis durch **Heuristiken** wie Rastersuche, Mehrskalen-Optimierung, Perturbation und Neustart

## **5.4 Nichtlineare Registrierung**

# Deformierbare Registrierung

- **Affine Transformationen** sind nicht flexibel genug, um
  - Bewegungen der meisten Organe auszugleichen (z.B. Atmung, Herzschlag, Brain Shift)
  - Anatomie verschiedener Patienten aufeinander abzubilden
- **Nichtlineare Registrierung** ermöglicht die hierzu benötigten lokalen Deformationen
  - *Grundidee:* Darstellung der Deformation durch ein Verschiebungsvektorfeld
  - Dieses muss **regularisiert** werden, da sonst
    - nicht genug Bildinformation zur Verfügung steht, um es zu schätzen
    - unplausible Deformationen erzeugt werden



# Verwandtes Problem: Optischer Fluss

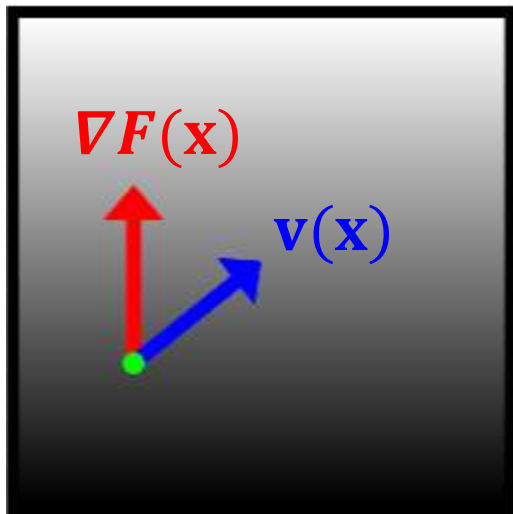


*Bildquelle:* Blender Foundation / [bigbuckbunny.org](http://bigbuckbunny.org)

# Optischer Fluss

- **Ziel:** Finde in einer Bildsequenz die Verschiebungsvektoren  $\mathbf{v}(\mathbf{x})$  jeden Pixels  $\mathbf{x}$  von einem Bild ins nächste
- **Zugrundeliegende Annahme:** Pixel bewegen sich, ändern jedoch nicht ihre Intensität
- Approximation erster Ordnung:

$$M(\mathbf{x}) = F(\mathbf{x}) - \langle \mathbf{v}(\mathbf{x}), \nabla F(\mathbf{x}) \rangle$$



**Hinweis:** Nur die Bewegung entlang des Gradienten wirkt sich auf die Intensität aus. Dies bezeichnet man als *Aperturproblem*.



# Lösung des Aperturproblems

- Um das **Aperturproblem** zu lösen, wählen wir den Verschiebungsvektor  $\mathbf{v}(\mathbf{x})$  mit der kleinsten möglichen Norm:  $\mathbf{v}(\mathbf{x}) = \lambda \nabla F(\mathbf{x})$

- Aus  $M(\mathbf{x}) = F(\mathbf{x}) - \langle \mathbf{v}(\mathbf{x}), \nabla F(\mathbf{x}) \rangle$  folgt dann

$$M(\mathbf{x}) = F(\mathbf{x}) - \lambda \|\nabla F(\mathbf{x})\|_2^2$$

$$\Rightarrow \lambda = \frac{F(\mathbf{x}) - M(\mathbf{x})}{\|\nabla F(\mathbf{x})\|_2^2}$$

$$\Rightarrow \mathbf{v}(\mathbf{x}) = \frac{F(\mathbf{x}) - M(\mathbf{x})}{\|\nabla F(\mathbf{x})\|_2^2} \nabla F(\mathbf{x})$$

– *Quiz*: Gibt es mit dieser Formel noch ein Problem?

# Dämonen-Regularisierung

- **Dämonen-Regularisierung** für  $\|\nabla F(\mathbf{x})\|^2 \approx 0$ :

$$\mathbf{v}_D(\mathbf{x}) = \frac{F(\mathbf{x}) - M(\mathbf{x})}{\|\nabla F(\mathbf{x})\|_2^2 + (F(\mathbf{x}) - M(\mathbf{x}))^2} \nabla F(\mathbf{x})$$

– wenn  $\|\nabla F(\mathbf{x})\|_2^2 + (F(\mathbf{x}) - M(\mathbf{x}))^2 < \epsilon : \mathbf{v}_D(\mathbf{x}) := \mathbf{0}$

- Hält Korrespondenzen **in der Nachbarschaft** ( $\|\mathbf{v}_D\| \leq 0.5$ )

$$\begin{aligned} \|\mathbf{v}_D(\mathbf{x})\| &= \frac{|F(\mathbf{x}) - M(\mathbf{x})| \|\nabla F(\mathbf{x})\|}{\|\nabla F(\mathbf{x})\|^2 + (F(\mathbf{x}) - M(\mathbf{x}))^2} \\ &= \frac{\text{GM}^2(|F(\mathbf{x}) - M(\mathbf{x})|, \|\nabla F(\mathbf{x})\|)}{2 \text{QM}^2(|F(\mathbf{x}) - M(\mathbf{x})|, \|\nabla F(\mathbf{x})\|)} \end{aligned}$$

– mit GM=geometrisches Mittel, QM=quadratisches Mittel

–  $\|\mathbf{v}_D\| \leq 0.5$  folgt aus der Ungleichung  $\text{GM} \leq \text{QM}$

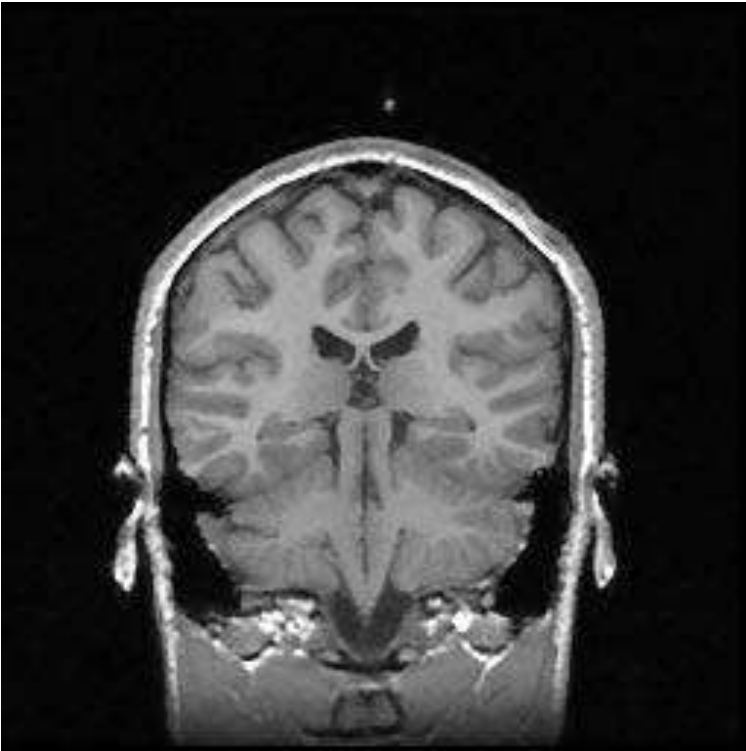
# Dämonen-Algorithmus zur Bildregistrierung

- Der **Dämonen-Algorithmus** zur Bildregistrierung (Thirion 1998) iteriert folgende Schritte:
  1. Berechne den regularisierten optischen Fluss  $\bar{\psi}(x)$  zwischen dem festen Bild  $F(x)$  und dem deformierten Objektbild  $M(\psi(x))$ .
  2. Addiere die Flussvektoren auf das aktuelle Verschiebungsfeld:
$$\hat{\psi}(x) = \psi(x) + \bar{\psi}(x)$$
  3. Regularisiere durch Gauss-Glättung:  $\varphi(x) = G_{\sigma} * \hat{\psi}(x)$ .  
Das ergibt das Verschiebungsfeld  $\psi(x)$  der nächsten Iteration

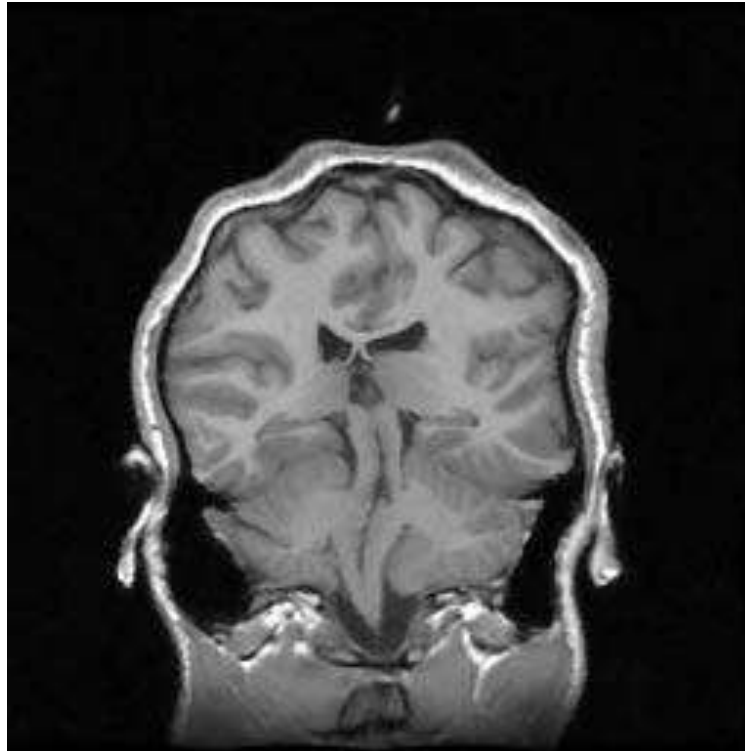
**Vorstellung:** “Maxwellsche Dämonen” besetzen die Pixel und verschieben das Bild aufgrund der Intensitätsunterschiede.

# Beispiel-Ergebnis des Dämonen-Algorithmus

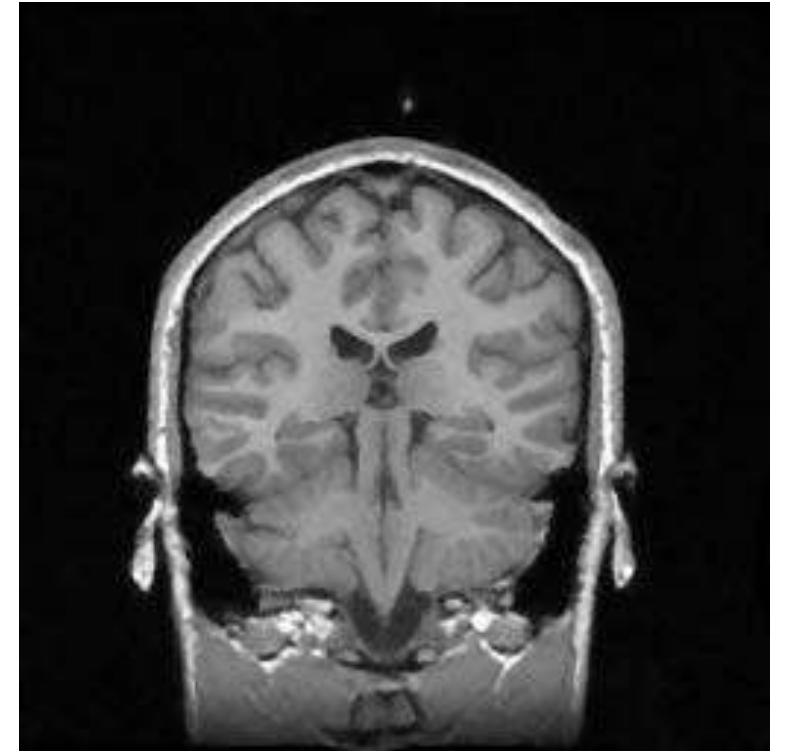
- Ergebnis von (Thirion 1998)



(a) Ursprüngliches MRT-Bild



(b) Deformierte Variante von (a)



(c) Registrierung von (b) auf (a)

# Zusammenfassung: Nichtlineare Registrierung

- **Grundidee** der nichtlinearen Registrierung:
  - Transformation durch ein **Verschiebungsvektorfeld** bietet genügend Flexibilität für lokale Deformationen
  - **Regularisierung** beschränkt Flexibilität auf ein sinnvolles Maß
- Klassischer Ansatz: **Dämonen-Algorithmus**
  - Basiert auf iterativer Berechnung des **optischen Flusses** zwischen Referenz und deformiertem Objektbild
    - Regularisiert optischen Fluss um sehr lange Vektoren zu vermeiden
  - Regularisiert das Verschiebungsvektorfeld zusätzlich durch **Gauss-Glättung** in jeder Iteration

## Zum Nach- und Weiterlesen

- Heinz Handels: *Medizinische Bildverarbeitung*. Vieweg+Teubner, 2. Auflage, 2009
- Isaac N. Bankman: *Handbook of Medical Imaging. Processing and Analysis*. Academic Press, 2000
- W.H. Press, S.A. Teukolsky, W.T. Vetterling, B.P. Flannery: *Numerical Recipes: The Art of Scientific Computing*. Cambridge University Press, 3<sup>rd</sup> edition, 2007
- J.-P. Thirion: *Image Matching as a Diffusion Process: An Analogy with Maxwell's Demons*. Medical Image Analysis 2(3):243-260, 1998