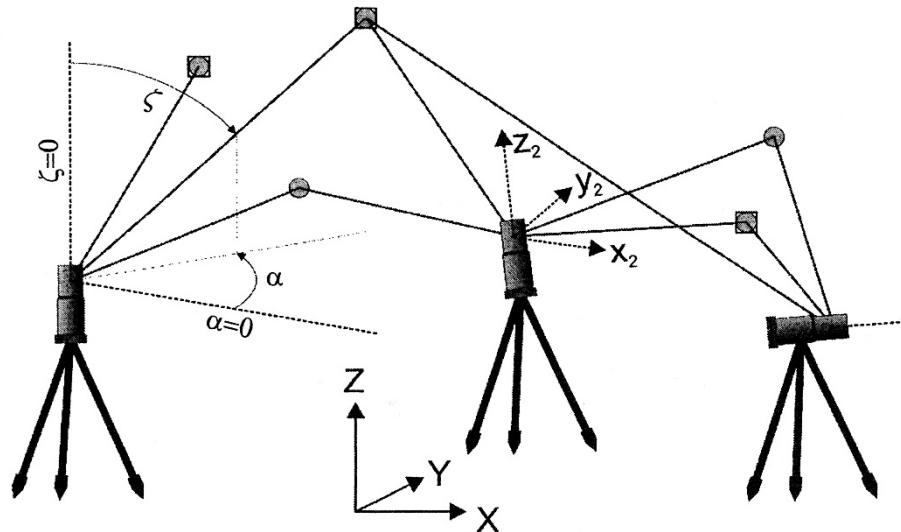


Die **Registrierung** beschreibt die Verknüpfung von Einzelscans verschiedener Standpunkte oder Epochen, sodass alle Koordinaten in einem einheitlichen Laserscannerkoordinatensystem vorliegen. Diese Transformation ist immer dann notwendig, wenn aufgrund von Abschattungen mehrere Standpunkte zur Aufnahme eines Objektes besetzt werden müssen. Dagegen gleicht die **Georeferenzierung** der Transformation der Scanpunkte in ein übergeordnetes oder absolutes Koordinatensystem.

Die Notwendigkeit der Georeferenzierung ist daher stark anwendungsbezogen.

Beide Arten der Koordinatentransformation sind eng miteinander verknüpft und aus Sicht der Mathematik identisch, verfolgen aber die erwähnten unterschiedlichen Ziele.

Da im Unterschied zur Tachymetrie keine signalisierten identischen Punkte zwischen zwei Koordinatensätzen oder Punktwolken vorhanden sind, können die Registrierparameter nur über Umwege gewonnen werden. Diese Registrierparameter folgen aus der Transformationsvorschrift einer **7-Parameter-Helmert-Transformation**.
(Quelle: Kuhlmann & Holst, 2017)

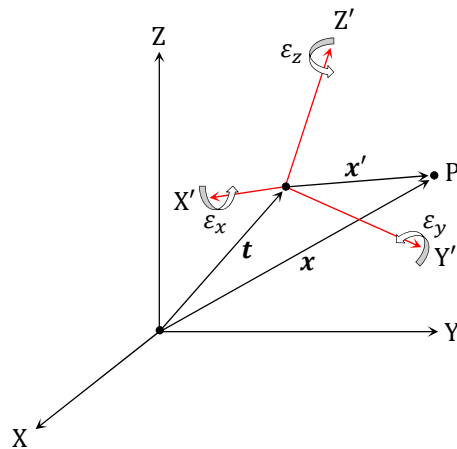


Verknüpfung mehrerer terrestrischer Laserscanner-Standpunkte (Scans)
(Quelle: Kraus, 2003)

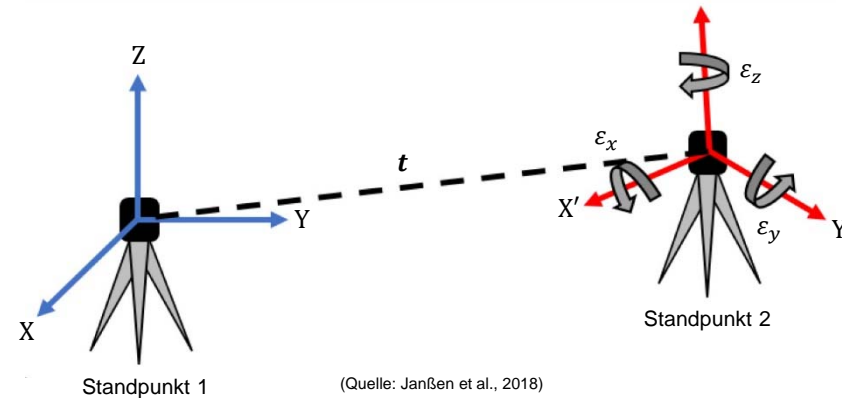
Die räumliche Zuordnung der Koordinaten x und x' eines Punktes in zwei Koordinatensystemen wird im Falle einer **konformen Transformation (Ähnlichkeitstransformation)** durch die Vorschrift

$$x = t + (1 + m) \cdot R(\varepsilon) \cdot x' \quad \text{vorgenommen.} \quad (\text{Formel 1})$$

→ einfache Lösung (vgl. absolute Orientierung der photogrammetrischen Stereomodelle)



Räumliche Ähnlichkeitstransformation
(Quelle: Welsch et al., 2000)



(Quelle: Janßen et al., 2018)

$t^T = (t_x \ t_y \ t_z)$... Translationsparameter, die den Unterschied in der Lage des Koordinatenursprungs des x' -Systems gegenüber dem x -System beschreiben.

$R(\varepsilon) = R_3(\varepsilon_z) \cdot R_2(\varepsilon_y) \cdot R_1(\varepsilon_x)$... Rotationsmatrix mit den Elementen $\varepsilon_x, \varepsilon_y, \varepsilon_z$, um die das x' -System gedreht werden muss, um in eine zum x -System achsparallele Orientierung gebracht zu werden.

m ... Maßstabsfaktor zur Anpassung des Maßstabs des x' -Systems an den des x -System

$$R_1(\varepsilon_x) = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos \varepsilon_x & \sin \varepsilon_x \\ 0 & -\sin \varepsilon_x & \cos \varepsilon_x \end{pmatrix}, \quad R_2(\varepsilon_y) = \begin{pmatrix} \cos \varepsilon_y & 0 & -\sin \varepsilon_y \\ 0 & 1 & 0 \\ \sin \varepsilon_y & 0 & \cos \varepsilon_y \end{pmatrix}, \quad R_3(\varepsilon_z) = \begin{pmatrix} \cos \varepsilon_z & \sin \varepsilon_z & 0 \\ -\sin \varepsilon_z & \cos \varepsilon_z & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

Drehsinn positiv entgegen dem Uhrzeigersinn.

Handelt es sich um kleine Drehwinkel, so kann $\cos \varepsilon_i = 1$, $\sin \varepsilon_i = \varepsilon_i$ und $\varepsilon_i \varepsilon_j$ für $i, j = x, y, z$ gesetzt werden. Dadurch vereinfacht sich $R(\varepsilon)$ zu

$$R(\varepsilon) = \begin{pmatrix} 1 & \varepsilon_z & -\varepsilon_y \\ -\varepsilon_z & 1 & \varepsilon_x \\ \varepsilon_y & -\varepsilon_x & 1 \end{pmatrix} \Rightarrow$$

$$\begin{pmatrix} x \\ y \\ z \end{pmatrix} = \begin{pmatrix} t_x \\ t_y \\ t_z \end{pmatrix} + (1 + m) \begin{pmatrix} 1 & \varepsilon_z & -\varepsilon_y \\ -\varepsilon_z & 1 & \varepsilon_x \\ \varepsilon_y & -\varepsilon_x & 1 \end{pmatrix} \begin{pmatrix} x' \\ y' \\ z' \end{pmatrix} \quad \dots \text{ differentielle Ähnlichkeitstransformation (Formel 2, HELMERT-Transformation)}$$

Eigenschaften von $R(\varepsilon)$: $R^{-1}(\varepsilon) = R^T(\varepsilon) \quad \dots \text{ orthogonale Drehmatrix}$
 $R^T(\varepsilon) \cdot R(\varepsilon) = R(\varepsilon) \cdot R^T(\varepsilon) = E \quad \Rightarrow$

$$x' = (1 - m) \cdot R^T(\varepsilon) \cdot (x - t) \quad \text{für } m \ll 1.$$

Kovarianzmatrizen der transformierten Punkte lauten $\Sigma_{xx} = (1 + 2m) \cdot R(\varepsilon) \cdot \Sigma_{x'x'} \cdot R^T(\varepsilon) \quad \text{und umgekehrt}$

$$\Sigma_{x'x'} = (1 - 2m) \cdot R^T(\varepsilon) \cdot \Sigma_{xx} \cdot R(\varepsilon).$$

Berechnung der Transformationsparameter $p^T = (t_x \quad t_y \quad t_z \quad \varepsilon_x \quad \varepsilon_y \quad \varepsilon_z \quad m)$

Die Parameter sind mit Hilfe der Koordinatenunterschiede einer hinreichenden Anzahl von in beiden Systemen identischen Punkten zu bestimmen.

$$\text{Aus } \begin{pmatrix} 0 & \varepsilon_z & -\varepsilon_y \\ -\varepsilon_z & 0 & \varepsilon_x \\ \varepsilon_y & -\varepsilon_x & 0 \end{pmatrix} \cdot \begin{pmatrix} x' \\ y' \\ z' \end{pmatrix} = \begin{pmatrix} 0 & -z' & y' \\ z' & 0 & -x' \\ -y' & x' & 0 \end{pmatrix} \cdot \begin{pmatrix} \varepsilon_x \\ \varepsilon_y \\ \varepsilon_z \end{pmatrix} \text{ folgt}$$

$$\begin{pmatrix} x \\ y \\ z \end{pmatrix} - \begin{pmatrix} x' \\ y' \\ z' \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 0 & -z' & y' & x' \\ 0 & 1 & 0 & z' & 0 & -x' & y' \\ 0 & 0 & 1 & -y' & x' & 0 & z' \end{pmatrix} \cdot p \quad (\text{hierbei sind kleine Größen } m \cdot \varepsilon_i \text{ vernachlässigt})$$

Kurznotation:

$$\Delta x = x - x' = G p$$

Wird die Gleichung für alle identischen Punkte (Passpunkte) angesetzt, kann sie nach den unbekannten Parametern p nach der Methode der kleinsten Quadrate aufgelöst werden.

Anmerkung: Für die eindeutige Bestimmung sind mindestens sieben Gleichungen notwendig. Die Passpunkte müssen dabei keine **Vollpasspunkte** (alle 3 Koordinaten gegeben) sein, es müssen lediglich insgesamt sieben Koordinaten, die im Raum ein dreidimensionales Gebilde beschreiben, gegeben sein.

HELMERT-Transformation

Werden mehr als die notwendige Zahl identischer Koordinaten zur Bestimmung der Parameter herangezogen, ergibt sich ein überbestimmtes System (**GAUSS-MARKOV-Modell**), das die Grundlage der bekannten **HELMERT-Transformation** (Formel 2) darstellt. Sie dient der *Auffelderung* der identischen Punkte zweier Netze/Scans aufeinander unter der Bedingung, die Quadratsumme der *Restklaffungen* zu einem Minimum zu machen. Als Restklaffungen bezeichnet man die nach der Transformation noch verbleibenden Unterschiede $v_{\Delta x}$ der Koordinaten der Vergleichspunkte. Sie haben die Bedeutung von Verbesserungen, die an den „Beobachtungen“ Δx anzubringen sind (Quelle: Welsch et al., 2000).

- Beobachtungsgleichungen

$$\Delta x + v_{\Delta x} = G \hat{p}, \quad P_{\Delta x} = E$$

- Normalgleichungen

$$G^T G \hat{p} = G^T \Delta x$$

- Lösung

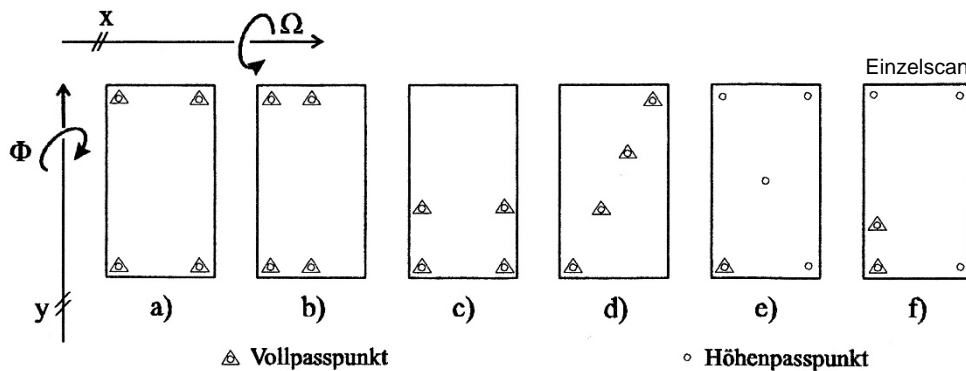
$$\hat{p} = (G^T G)^{-1} G^T \Delta x = Q_{pp} G^T \Delta x$$

Vergleiche dazu auch Kraus (2004): Absolute Orientierung, kleinste-Quadrate-Ausgleichung (Kap. 4.4).

- Falls die Scans mit identischen Laserscannern aufgenommen werden, wird der Maßstab als $\mu = 1$ angenommen. Dies gilt ferner für die Registrierung kleinräumiger Scans, bei denen ein Maßstabsunterschied kaum bestimmbar wäre.
- Falls die Laserscanner horizontalisiert sind, entfallen die Rotationen um die x - und y -Achse, sodass diese zusätzliche als $\varepsilon_x = \varepsilon_y = 0$ angenommen werden können. Die Parameteranzahl reduziert sich auf 4.

Punktverteilung

Vergleiche dazu die Fehlertheorie zur absoluten Orientierung in der Photogrammetrie.



- a) Standardfall
- b) Neigung Φ (zu) schlecht
- c) Neigung Ω (zu) schlecht
- d) Neigungen Ω und Φ unbestimmt, wenn Passpunkte exakt auf einer Geraden liegen.
- e) Kantung K und Maßstabszahl m unbestimmt
- f) Kantung K und Maßstabszahl m schlecht

Beurteilung verschiedener Passpunktanordnungen für die absolute Orientierung
(Quelle: Kraus, 2003)

Die strengere Lösung minimiert die Verbesserungen der **ursprünglichen** Beobachtungen α, ζ, s in der Quadratsumme.

$$\begin{pmatrix} x \\ y \\ z \end{pmatrix} = s \begin{pmatrix} \sin \zeta \cos \alpha \\ \sin \zeta \sin \alpha \\ \cos \zeta \end{pmatrix}$$

$$\begin{pmatrix} \bar{x} \\ \bar{y} \\ \bar{z} \end{pmatrix} + \begin{pmatrix} v_x \\ v_y \\ v_z \end{pmatrix} = (\bar{s} + v_s) \begin{pmatrix} \sin(\bar{\zeta} + v_\zeta) \cos(\bar{\alpha} + v_\alpha) \\ \sin(\bar{\zeta} + v_\zeta) \sin(\bar{\alpha} + v_\alpha) \\ \cos(\bar{\zeta} + v_\zeta) \end{pmatrix} = \bar{s} \begin{pmatrix} \sin \bar{\zeta} \cos \bar{\alpha} \\ \sin \bar{\zeta} \sin \bar{\alpha} \\ \cos \bar{\zeta} \end{pmatrix} + \overbrace{\bar{s} \begin{pmatrix} -\sin \bar{\zeta} \sin \bar{\alpha} \\ \cos \bar{\zeta} \cos \bar{\alpha} \\ -\sin \bar{\zeta} \end{pmatrix} v_\alpha + \bar{s} \begin{pmatrix} \cos \bar{\zeta} \cos \bar{\alpha} \\ \cos \bar{\zeta} \sin \bar{\alpha} \\ -\sin \bar{\zeta} \end{pmatrix} v_\zeta + \bar{s} \begin{pmatrix} \sin \bar{\zeta} \cos \bar{\alpha} \\ \sin \bar{\zeta} \sin \bar{\alpha} \\ \cos \bar{\zeta} \end{pmatrix} v_s}_{\text{nur Glieder erster Ordnung}}$$

$$\begin{pmatrix} v_x \\ v_y \\ v_z \end{pmatrix} = \begin{pmatrix} \sin \bar{\alpha} & \cos \bar{\zeta} \cos \bar{\alpha} & \sin \bar{\zeta} \cos \bar{\alpha} \\ -\cos \bar{\alpha} & \cos \bar{\zeta} \sin \bar{\alpha} & \sin \bar{\zeta} \sin \bar{\alpha} \\ 0 & -\sin \bar{\zeta} & \cos \bar{\zeta} \end{pmatrix} \begin{pmatrix} -(\sin \bar{\zeta}) \bar{s} \cdot v_\alpha \\ \bar{s} \cdot v_\zeta \\ v_s \end{pmatrix} = \mathbf{D}_{\bar{\zeta} \bar{\alpha}} \begin{pmatrix} -\bar{s} \cdot \sin \bar{\zeta} & & \\ & \bar{s} & \\ & & 1 \end{pmatrix} \begin{pmatrix} v_\alpha \\ v_\zeta \\ v_s \end{pmatrix}$$

Durch Einsetzen in $\mathbf{x} = \mathbf{t} + (1 + m) \cdot \mathbf{R}(\varepsilon) \cdot \mathbf{x}'$ erhält man für $m = 1$

$$\bar{s} \begin{pmatrix} \sin \bar{\zeta} \cos \bar{\alpha} \\ \sin \bar{\zeta} \sin \bar{\alpha} \\ \cos \bar{\zeta} \end{pmatrix} + \mathbf{D}_{\bar{\zeta} \bar{\alpha}} \begin{pmatrix} -\bar{s} \cdot \sin \bar{\zeta} & & \\ & \bar{s} & \\ & & 1 \end{pmatrix} \begin{pmatrix} v_\alpha \\ v_\zeta \\ v_s \end{pmatrix} = \mathbf{R}^T (\mathbf{x} - \mathbf{t}).$$

Multiplikation von links mit der Matrix $\mathbf{D}_{\bar{\zeta} \bar{\alpha}}^T$ ergibt:

$$\begin{pmatrix} \bar{0} \\ \bar{0} \\ \bar{s} \end{pmatrix} + \begin{pmatrix} -\bar{s} \cdot \sin \bar{\zeta} & & \\ & \bar{s} & \\ & & 1 \end{pmatrix} \begin{pmatrix} v_\alpha \\ v_\zeta \\ v_s \end{pmatrix} = \mathbf{D}_{\bar{\zeta} \bar{\alpha}}^T \mathbf{R}^T (\mathbf{x} - \mathbf{t})$$

Durch geringe Umstellung folgt die gesuchte Beziehung:

$$\begin{pmatrix} \bar{0} + v_\alpha \\ \bar{0} + v_\zeta \\ \bar{s} + v_s \end{pmatrix} = \begin{pmatrix} \frac{-1}{\bar{s} \cdot \sin \bar{\zeta}} & & \\ & \frac{1}{\bar{s}} & \\ & & 1 \end{pmatrix} \mathbf{D}_{\bar{\zeta} \bar{\alpha}}^T \mathbf{R}^T (\mathbf{x} - \mathbf{t}) \rightarrow \text{Freie Stationierung}$$

festе Parameter

freie Parameter: 3 DOF (\mathbf{R}), 3 DOF (\mathbf{t}), \mathbf{x}_i (Sie sind aus Beobachtungen entstanden!)

$$\begin{pmatrix} x \\ y \\ z \end{pmatrix} = \begin{pmatrix} \bar{x} \\ \bar{y} \\ \bar{z} \end{pmatrix} + \begin{pmatrix} v_x \\ v_y \\ v_z \end{pmatrix} = \begin{pmatrix} X \\ Y \\ Z \end{pmatrix}$$

Quelle: Kraus (1996)

Source: Lichti & Skaloud, 2010

Rotation representation using four algebraic parameters:

$$q = q_0 + q_1\mathbf{i} + q_2\mathbf{j} + q_3\mathbf{k} \quad \dots \text{quaternion}$$

$$= q_0 + \mathbf{q} \quad \dots \text{scalar part + vector part (imaginary part)}$$

$$q_0^2 + q_1^2 + q_2^2 + q_3^2 = 1 \quad \dots \text{unit quaternion}$$

$$\mathbf{R} = \begin{pmatrix} q_0^2 + q_1^2 - q_2^2 - q_3^2 & 2(q_1q_2 + q_0q_3) & 2(q_1q_3 - q_0q_2) \\ 2(q_1q_2 - q_0q_3) & q_0^2 - q_1^2 + q_2^2 - q_3^2 & 2(q_2q_3 + q_0q_1) \\ 2(q_1q_3 + q_0q_2) & 2(q_2q_3 - q_0q_1) & q_0^2 - q_1^2 - q_2^2 + q_3^2 \end{pmatrix}$$

$$\bar{q} = q_0 - q_1\mathbf{i} - q_2\mathbf{j} - q_3\mathbf{k} \quad \dots \text{conjugate of a quaternion}$$

$$\|q\| = \sqrt{q\bar{q}} = \sqrt{\bar{q}q} = \sqrt{q_0^2 + q_1^2 + q_2^2 + q_3^2} \quad \dots \text{norm}$$

The unit quaternion can be expressed in terms of the angle and axis parameters as follows:

$$q = \cos\left(\frac{\theta}{2}\right) + \sin\left(\frac{\theta}{2}\right)n_1\mathbf{i} + \sin\left(\frac{\theta}{2}\right)n_2\mathbf{j} + \sin\left(\frac{\theta}{2}\right)n_3\mathbf{k}$$

The angle-axis and quaternion representations are free from the singularity problem known as [gimbal lock](#).

Closed-form orientation estimation (cp. with Arun et al., 1987)

$$\mathbf{y}_i = \mathbf{R}\mathbf{x}_i + \mathbf{y}_0 \quad \dots \text{rigid body transformation (scale } s = 1)$$

\mathbf{R} ... rotation matrix, \mathbf{y}_0 ... translation vector
 \mathbf{y}_i ... master point cloud, \mathbf{x}_i ... slave point cloud

$$e^2 = \sum_i \|\mathbf{R}\mathbf{x}_i + \mathbf{y}_0 - \mathbf{y}_i\|^2 \rightarrow \min$$

Horn's (1987) method for estimating the transformation parameters

Each set of points is first reduced to their respective centroids so that they share a common origin.

$$\mathbf{x}_m = \frac{1}{n_x} \sum_{i=1}^{n_x} \mathbf{x}_i, \quad \mathbf{y}_m = \frac{1}{n_y} \sum_{i=1}^{n_y} \mathbf{y}_i$$

$$\mathbf{x}'_i = \mathbf{x}_i - \mathbf{x}_m, \quad \mathbf{y}'_i = \mathbf{y}_i - \mathbf{y}_m$$

⇒ This allows independent estimation of the rotation and translation elements of the transformation.

(1) Estimation of the **rotation**

$$e^2 = \sum_i \|\mathbf{R}\mathbf{x}'_i - \mathbf{y}'_i\|^2 = \sum_i (\mathbf{x}'_i{}^T \mathbf{x}'_i + \mathbf{y}'_i{}^T \mathbf{y}'_i - 2\mathbf{y}'_i{}^T \mathbf{R}\mathbf{x}'_i) \rightarrow \min$$

Since the first two terms are sums of squares of the point coordinates, e^2 is minimized when the third term is maximized. To do so, the rotation is parameterized in terms of a unit quaternion, $q = q_0 + q_1\mathbf{i} + q_2\mathbf{j} + q_3\mathbf{k}$, and the third term is manipulated into the form

$$\sum_i \mathbf{y}'_i{}^T \mathbf{R}\mathbf{x}'_i = q^T \mathbf{N} q \quad \text{where} \quad \mathbf{N} = \begin{pmatrix} S_{xx} + S_{yy} + S_{zz} & S_{yz} - S_{zy} & S_{zx} - S_{xz} & S_{xy} - S_{yx} \\ S_{xx} - S_{yy} - S_{zz} & S_{xy} + S_{yx} & S_{zx} + S_{xz} & S_{yz} + S_{zy} \\ -S_{xx} + S_{yy} - S_{zz} & -S_{xy} + S_{yx} & -S_{zx} + S_{xz} & -S_{yz} + S_{zy} \\ \text{sym.} & & & \end{pmatrix}$$

Whose entries are functions of the elements of the cross-covariance matrix of point coordinates, \mathbf{S}

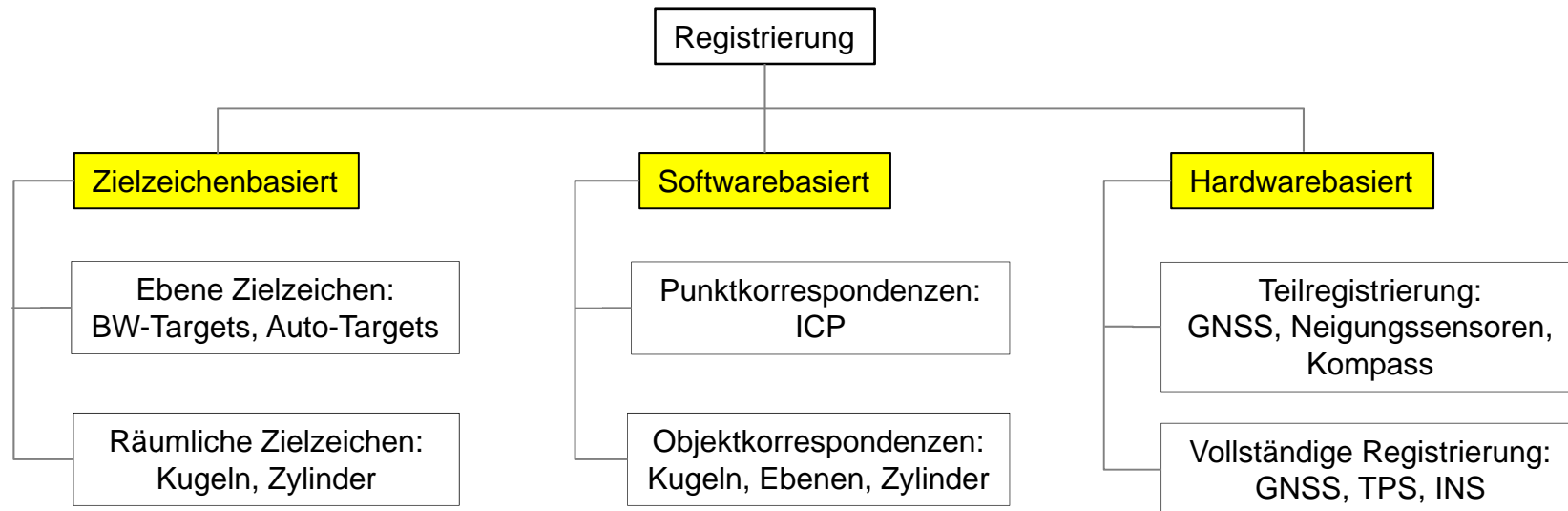
$$\mathbf{S} = \sum_{i=1} \mathbf{x}'_i \mathbf{y}'_i{}^T = \begin{pmatrix} S_{xx} & S_{xy} & S_{xz} \\ S_{yx} & S_{yy} & S_{yz} \\ S_{zx} & S_{zy} & S_{zz} \end{pmatrix}$$

The unit quaternion that maximises $q^T \mathbf{N} q$ is the eigenvector corresponding to the largest positive eigenvalue of \mathbf{N} . The coordinate centroids and estimated rotation matrix are then substituted into $\mathbf{y}_i = \mathbf{R}\mathbf{x}_i + \mathbf{y}_0$ to calculate

(2) the **translation vector** $\mathbf{y}_0 = \mathbf{y}_m - \mathbf{R}\mathbf{x}_m$.

The drawback of this method, which is a least-squares solution, is that weights cannot be introduced.

Für die Bestimmung der Parameterwerte der Ähnlichkeitstransformation haben sich verschiedene Strategien etabliert, die sich in drei Kategorien untergliedern lassen:



Möglichkeiten der Registrierung von Einzelscans
(Quelle: Kuhlmann & Holst, 2017)

Damit auch beim Scanning identische Punkte zwischen Einzelscans vorliegen, werden bei der zielzeichenbasierten Registrierung im einfachsten Fall ebene Zielzeichen (engl. Targets) aufgestellt, die der Laserscanner erkennt und denen er mit Hilfe spezieller Detektionsalgorithmen, die neben der geometrischen Messung auch die Intensität nutzen, eine 3D-Koordinate zuweisen kann. Die zielzeichenbasierte Registrierung lässt sich zu einer **Georeferenzierung** erweitern, falls die Zielzeichen bereits in einem übergeordneten Koordinatensystem gegeben sind.

Planare Zielzeichen

Schwarz-Weiß-Targets/ Sektorenmarken



Leica GZT21 4.5" circular B/W Tilt & Turn target



Target IGG Bonn

Reflexfolien (kreisförmige Marken)



Leica Geosystems HDS Target



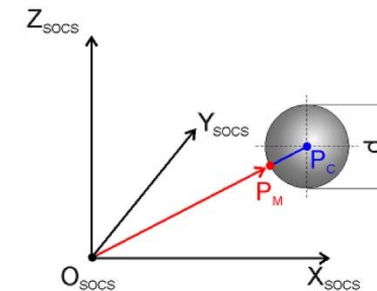
RiegI flat circular retroreflectors,
diameter 50/100 mm,
self-adhesive

3D-Zielzeichen

Kugeln



<https://www.faro.com/>

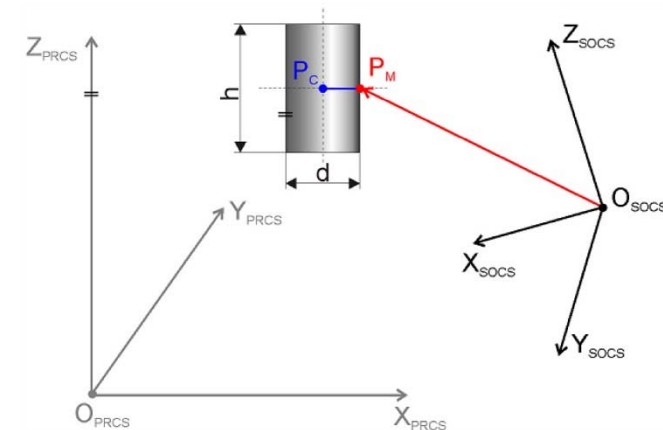


Bestimmung des Zentrums des Kugel
(Quelle: Handbuch RiSACN PRO, 2009)

Zylinder



RiegI cylindrical retroreflectors,
height 50 mm x diameter 50 mm

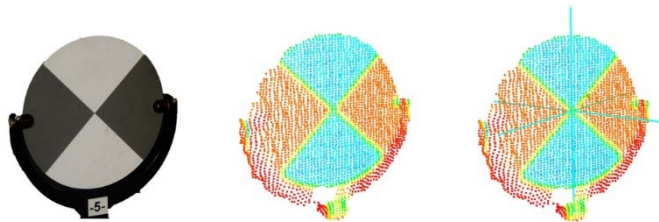


Bestimmung des Zentrums des Zylinders
Quelle: Handbuch RiSACN PRO, 2009)

Verfahren zur Koordinatenschätzung (ebener Sektorenmarken)

- Schätzung mittels Kantenextraktion

Der Algorithmus mittels Kantenextraktion nutzt die Gradienten zwischen den schwarzen und weißen Teilflächen des Zielzeichens zur Schätzung der Zielzeichenkoordinate. Im ersten Schritt wird eine Ebene durch die Punkte des Zielzeichens geschätzt. Die Ebenenparameter werden genutzt, um eine Verebnung der Punktwolke durchzuführen und so die dreidimensionale Aufgabe der Koordinatenschätzung auf eine zweidimensionale Aufgabe zu reduzieren. Als nächstes wird ein Raster über die 2D-Punktwolke gelegt und so ein Grauwertbild aus der Punktwolke abgeleitet. Jede Rasterzelle bildet ein Pixel, dem eine eindeutige x- und y-Koordinate zugeordnet werden kann. Der Grauwert des Pixels wird aus den Intensitäten der Punktwolke interpoliert. Mittels der Bildgradienten werden die **Kantenpixel** zwischen den schwarzen und weißen Teilflächen aus dem Grauwertbild extrahiert (Canny, 1986). Durch diese Kanten werden anschließend zwei Geraden (entsprechend dem Target-Muster, vgl. Abb.) geschätzt (Hough-Transformation: <https://de.wikipedia.org/wiki/Hough-Transformation>). Der Schnittpunkt der beiden (oder auch mehrerer) Geraden liefert den gesuchten Referenzpunkt des Zielzeichens im 2D-Raum. Mit Hilfe der Ebenenparameter kann der zweidimensionale Referenzpunkt auf den dreidimensionalen Referenzpunkt zurückgeführt werden (Janßen et al., 2018).



← Photo eines Zielzeichens der Fa. Leica Geosystems (links), Punktwolke des gescannten Zielzeichens (Mitte) und Punktwolke mit der von der Herstellersoftware geschätzten Koordinate (rechts). (Quelle: Janßen et al., 2018)

- Schätzung mittels Bildkorrelation

Dieses Verfahren nutzt Bildkorrelationen zur Schätzung der Koordinate eines Zielzeichens. Analog zum vorherigen Algorithmus wird zuerst eine Ebene durch die Punktwolke des Zielzeichens geschätzt und anschließend ein Grauwertbild des Zielbildes abgeleitet. Im nächsten Schritt wird ein ideales Schwarz-Weiß-Zielzeichen (Template) erzeugt und anschließend die Kreuzkorrelation zwischen dem Grauwertbild und dem Template berechnet. Die Stelle der höchsten Korrelation wird verwendet, um, zusammen mit den x- und y-Werten der Pixel, den 2D-Referenzpunkt des Zielzeichens zu bestimmen (vgl. die korrelationsbasierte Bildregistrierung in der Digitalphotogrammetrie). Mit Hilfe der Ebenenparameter kann die geschätzte 2D-Koordinate in eine 3D-Koordinate des Zielzeichens umgerechnet werden. (Janßen et al., 2018).

- Interaktive Messung am Bildschirm

Messung kreisförmiger Marken (Quelle: Luhmann, 2018)

- Bildschirmmessung
- Schwerpunktverfahren (Geeignet bei Bildmustern mit symmetrischer Verteilung der Grauwerte.)

$$x_M = \frac{\sum_{i=1}^n (x_i T g_i)}{\sum_{i=1}^n (T g_i)} \quad y_M = \frac{\sum_{i=1}^n (y_i T g_i)}{\sum_{i=1}^n (T g_i)} \quad T = \begin{cases} 0 & \text{für } g < t \\ 1 & \text{für } g \geq t \end{cases}$$

T ist eine Entscheidungsfunktion, die festlegt, ob ein Pixel mit in die Berechnung einbezogen wird. T kann z.B. mithilfe eines (adaptiven) Schwellwertes t definiert werden.

- Korrelationsverfahren

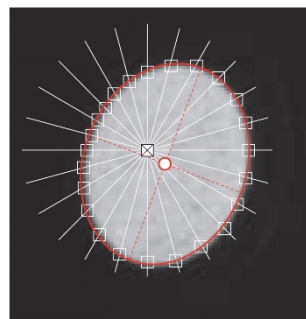
Unter Korrelationsverfahren werden hier solche Methoden verstanden, die ein Ähnlichkeitsmaß wie z.B. der [normierte Kreuzkorrelationskoeffizient](#) zwischen einem Musterbild und einem Suchbild berechnen. Die Stelle der besten Übereinstimmung wird mit der Position des zu suchenden Musters gleichgesetzt.

- Kleinste-Quadrate-Anpassung

Das Verfahren der Bildzuordnung nach kleinsten Quadraten ([least squares matching, LSM](#)) basiert auf einer iterativen geometrischen und radiometrischen Transformation zwischen Musterbild und Suchbild mit dem Ziel, die Quadratsumme der Grauwertdifferenzen zwischen Musterbild und transformierten Suchbild zu minimieren.

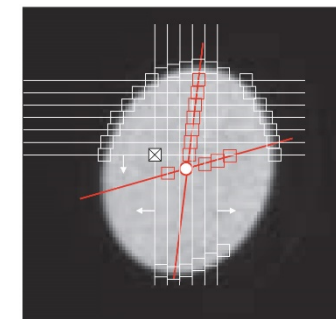
→ Bei bekannter Grauwertstruktur kann das Musterbild auch synthetisch erzeugt und als sogenanntes Template für alle entsprechenden Punkte im Suchbild verwendet werden ([least squares template matching](#)).

- Detektion kreis- und ellipsenförmiger Muster



a) Prinzip des Stern-Operators

- ☒ Näherungswert
- Kantenpunkt
- Ausgleichselement
- Mittelpunkt



b) Prinzip des Zhou-Ellipsenoperators

- ☒ Näherungswert
- Kantenpunkt
- Halbmesserpunkt
- Ausgleichselement
- Mittelpunkt

Die Idee der softwarebasierten Registrierung ist, die in den Einzelscans teilweise überlappenden Punktwolken selber zur Bestimmung der Transformationsparameter zu nutzen. Dabei werden wieder Punkte oder geometrische Objekte gesucht, die ihre Korrespondenzen in jeweils einer anderen Punktwolke haben. Diese Objekte sind dann nicht künstlich im Laserscan platziert worden, sondern werden automatisch aus dem gescannten Objekt selber entnommen. Hierbei ist dann nicht von identischen Punkten oder Ebenen die Rede, sondern von **korrespondierenden** oder **nächsten** Elementen, da die Zuordnung zunächst über die willkürliche Abtastung der Oberfläche erfolgt.

Die softwarebasierte Registrierung erweitert sich zu einer Georeferenzierung, falls diese im Scan abgetasteten Objekte (die in ihrer Gesamtheit auch ein komplettes 3D-Modell bilden können) bereits in einem übergeordneten Koordinatensystem, in die die Scans dann transformiert werden sollen, gegeben sind.

Die einfachste Form dieser Registrierung ist der von Besl & McMay (1992) eingeführte **Iterative Closest Point (ICP) Algorithmus**.

(Quelle: Kuhlmann & Host, 2017)

(Quelle: Lichti & Skaloud, 2010)

Iterative closest point objective function

$$\mathbf{y}_i = \mathbf{R}\mathbf{x}_i + \mathbf{y}_0 \quad (\text{rigid body transformation})$$

\mathbf{R} ... rotation matrix, \mathbf{y}_0 ... translation vector

\mathbf{y}_i ... master point cloud, \mathbf{x}_i ... slave point cloud

The basic premise of the ICP is that the slave point cloud or data shape is rigidly transformed to best fit the master point cloud or model shape such that the sum of squares of Euclidean distances between the nearest (transformed) slave point, \mathbf{x}_i , and the master point, \mathbf{y}_i , is minimized, i.e.

$$e^2 = \sum_i \|\mathbf{R}\mathbf{x}_i + \mathbf{y}_0 - \mathbf{y}_i\|^2 \rightarrow \min$$

Where the summation is taken over all data points in the overlap area.

Iterative closest point description

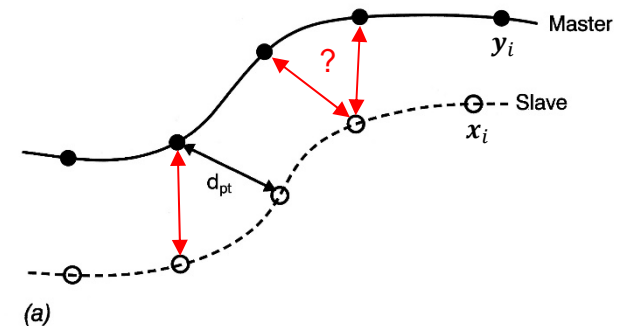
Besl & McKay (1992) proposed the ICP algorithm for shape registration.

Though nowadays it is most commonly used to register two or more point clouds, they present a general framework for the registration of points, line segments and triangulated surfaces as well as implicit and parametric curves and surfaces. Here the focus is on the registration of two point clouds. Besl & McKay (1992) also parameterise the rotation in terms of the [unit quaternion](#) and define the translation, \mathbf{y}_0 , as

$$\mathbf{y}_0 = (q_4 \quad q_5 \quad q_6)^T$$

Good **a priori alignment** of the two point clouds is required.

This is often done manually by selecting a small set of corresponding points and using them for estimation the initial transformation parameters. Recent research has tried to eliminate this step with the goal of completely automating the algorithm (→ see special topic). The estimated transformation is then applied to the slave point cloud.



ICP distance reckoning method: (a) point-point distance

The basic ICP algorithm then proceeds as follows (Besl & McKay, 1992):

1. Approximate values for the transformation parameters are all set to zero, except for q_0 , which is set to 1.
2. Nearest neighbour points are located either by k-d tree or other methods (vgl. 06_TLS_Modellierung.pdf).
3. The transformation parameters are determined by Horn's method and applied to the slave point cloud.
4. The RMS registration error (i.e. e) is calculated and compared with a convergence criterion.
If the test fails, then the process is repeated from step 1.

A high number of iterations is typically required and the method is prone to outliers.

Other iterative closest point approaches

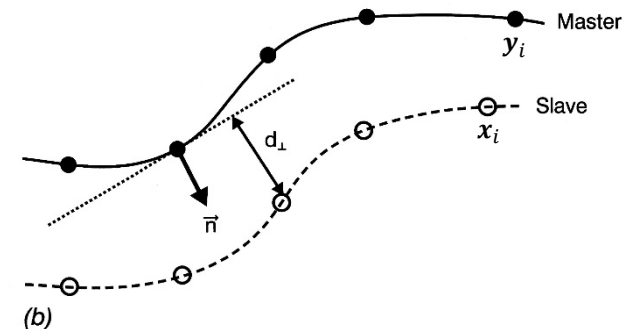
Chen & Medioni (1992) take a slightly different approach in their formulation. Among the differences are the rotation parameterisation in terms of three independent angles, the use of homogeneous coordinates and, most significantly, how the distance measure is reckoned. Their objective function is to minimise the sum of squares of Euclidian distances between slave points and the master point cloud [surface](#). The surface of the master point cloud is locally approximated by a tangent plane and the distance is taken along the surface normal from the slave point – so-called [normal shooting](#). → The direction in which the distance can be reduced is constrained to the surface normal vector. The single point has two other degrees of freedom, in which it can move in accordance with the constraints imposed by other points and planes.

The principle advantage in avoiding the use of nearest points is more rapid convergence.

William & Bennamoun (2001) present an ICP method for [simultaneous registration of multiple point clouds](#). Theirs is a global approach that is immune to the problem of systematic error propagation. This phenomenon occurs when point clouds are added sequentially in pair-wise registration. → *Loop closure*

See also: global ICP,

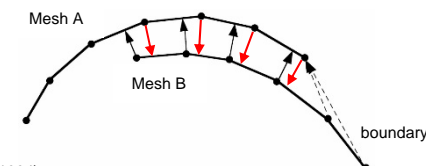
<https://www.geo.tuwien.ac.at/downloads/pg/pctools/globalICPGeneralDescription.html>



ICP distance reckoning method: (b) point-plane distance

A number of heuristics have been developed to overcome false correspondences:

- *Maximum match distance.* Reject any matches between points separated by a distance greater than a threshold value.
- *Normal vector compatibility.* Prevent matches between points whose normal have an angle between them greater than some threshold. Other feature such as surface curvature or colour may also be used.
- *Boundary match rejection.* Reject matches where the closest point is on the boundary of the other surface.



(Quelle: Turk & Levoy, 1994)

Automatic marker-free registration using SIFT key points extracted from the reflectance data of the scans (Böhm & Becker, 2007)

(see also the work of Barnea & Filin, 2007)

Scale Invariant Feature Transform (SIFT) proposed by Lowe (2004) is used for the extraction of features. The operator works on an image pyramid and computes the **Difference-of-Gaussian** (DoG) for each level. Local extrema are candidate features. An invariant representation of the area around the feature point is used as a **feature descriptor**. The features have shown to be largely invariant to scale changes and varying illumination.

1) Feature extraction

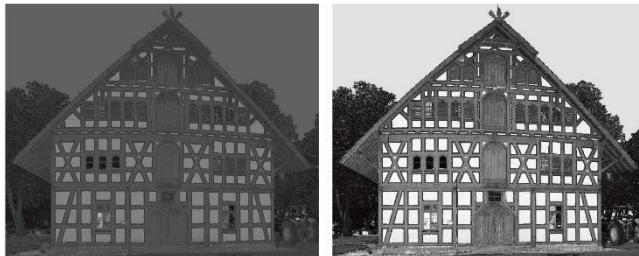


Figure 1: The laser return image before (left) and after (right) histogram equalization.

Scan data: Leica HDS 3000

Reflectance information: 10 bit

→ Histogram equalization

→ Scaling the data: 8 bit

The image obtained looks like a typical intensity image (Figure 1)

Feature extraction using SIFT method

Remark: In the same way keypoint detectors [corner detectors] can also be applied to range images (Barnea & Filin, 2008).

2) Pair-wise registration

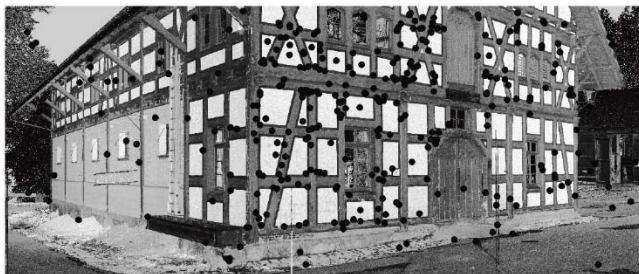


Figure 2: The feature points extracted and matched using SIFT method.
(7772 key points in the first image; 552 corresponding points matched in the second image; only 69 points are matched as valid 3D corresponding tie points using RANSAC; ⇒ 87.5% percent outliers)

RANSAC (Fischler & Bolles, 1981) filtering scheme : Randomly a sample of three point pairs is drawn from all SIFT matches. From the pair of three points a rigid body transformation is computed. All SIFT matches are checked against this transformation for consensus. The sample with the largest consensus is selected for registration. The final result is computed from all points in consensus with the sample. The author state that the automatic registrations give the same results as manual registration.

3) Marker-free registration of images

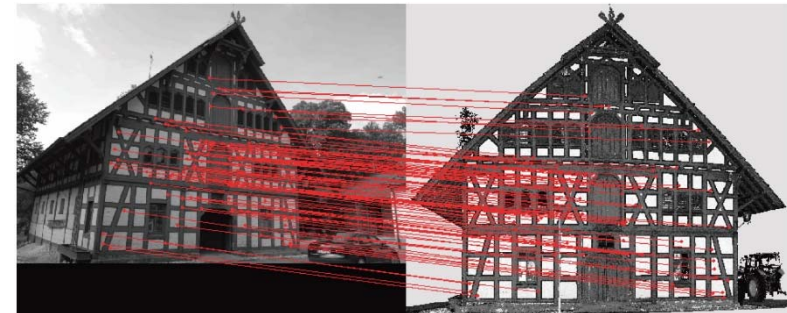


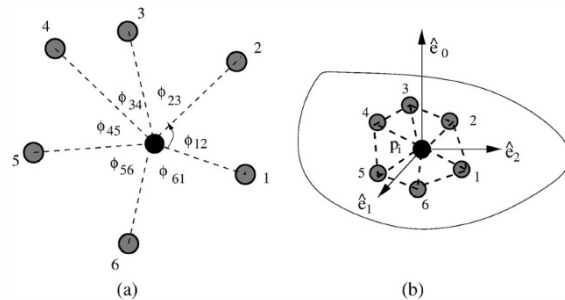
Figure 3: Key point correspondences for a photograph (left) and a laser reflectance image (right).
The photograph was captured by a NIKON camera.

In order to have similar intensity values in both images, only the green channel of the photograph has been considered for the determination of corresponding points. The resulting key points were extracted and matched by means of the sift algorithm. Wrong matches are removed by a RANSAC based computation of a closed form space resection. The number of inlier matches is about 20% (see Fig. 3).

Alignment using features extracted from the 3D point cloud

(Quelle: Lichti & Skaloud, 2010)

Bae and Lichti (2008) determine the approximate alignment using features extracted from the 3D point cloud. Extraction and subsequent matching are performed using approximate curvature measures computed from the eigenvalue decomposition of each point's local neighbourhood covariance matrix.

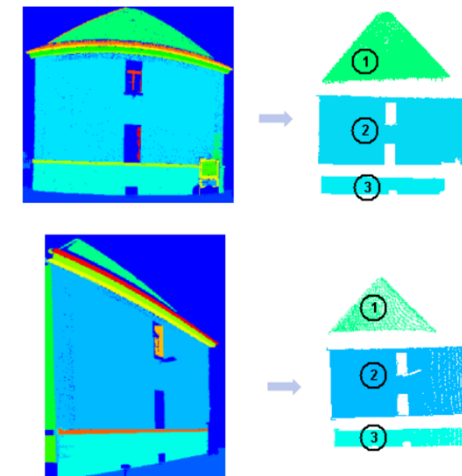


(a) A point and its neighbourhood are projected onto the plane whose normal vector is the eigenvector of the smallest eigenvalue of its covariance matrix. (b) The eigenvectors of $\text{COV}(p_i)$, $\hat{e}_{i=0, \dots, 2}$, are shown. (Source: Bae & Lichti, 2008)

Beinat et al. (2006) use more rigorously computed Gaussian curvature.

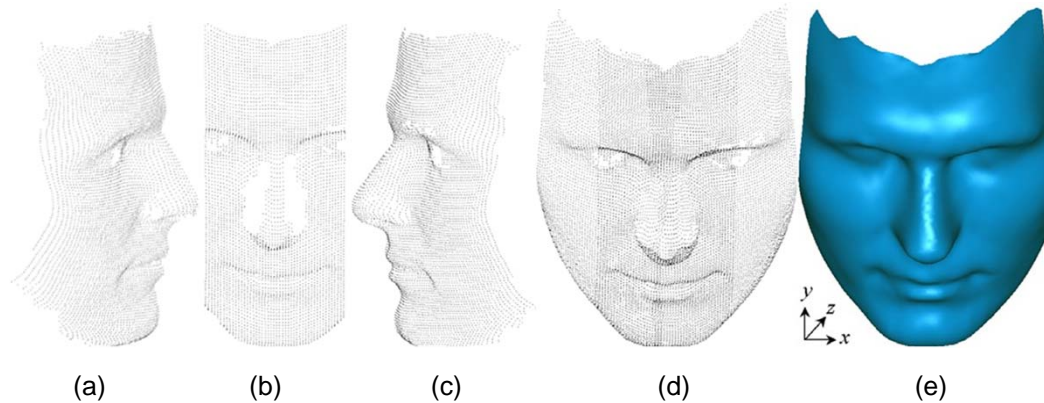
Planar segmentation for feature extraction is used by **Dold & Brenner (2006)**. The extracted planar patches are matched across scans to retrieve the pose parameters. A significant reduction of the number of combinations in the matching process is achieved by using geometrical constraints. Attributes are derived from the planar patches which allow to find corresponding matches without the exploration of the whole search space. For the calculation of geometrical features the convex hull of points supporting a planar patch is derived at first. Therefore the 3D points are projected onto the planar patch and the coordinates are transferred in a way, that one coordinate axis coincides with the patch normal. So 2D points are achieved and a 2D convex hull is calculated. The following features derived from laser scanning data are determined:

- Area
- Length of the boundary
- Bounding box oriented along the longest edge
(A characteristic feature of the bounding box is the ratio between the edges.)
- Mean intensity value



Segmentation result (left) and example for derived geometry (right) of two different scan positions (Source: Dold & Brenner, 2006).

Grün & Akca (2005) outline their least-squares 3D surface matching algorithm, which allows one or more slave surfaces to be matched to a 3D master (template) surface. The objective function of minimum sum of squares of distances is the same as in other methods, but the parametric least-squares formulation ([Gauss-Markov model](#)) allows incorporation of observational weights, rigorous error propagation, outlier detection and shaping parameters to model deformation between point clouds. Good initial alignment is required, but convergence is more rapid than with the standard ICP method (Lichti & Skaloud, 2010).



Example "face".

- (a) Left-search surface,
- (b) centre-template surface,
- (c) right-search surface,
- (d) obtained 3D point cloud after LS3D surface matching,
- (e) shaded view of the final composite surface.

(Source: Grün & Akca, 2005)

$$f(x, y, z) = g(x, y, z) \dots$$

f and g are discrete representations of the template and search surfaces. According to this equation each surface element on the template surface $f(x, y, z)$ patch has an exact coorespdnent surface element on the search surface $g(x, y, z)$, or vice-versa, if both of the surfaces would analytically be continuous surfaces without any deterministic or stocastic discrepancies.

$$f(x, y, z) - e(x, y, z) = g(x, y, z) \quad e(x, y, z) \dots \text{true error vector}$$

$$\sum \|\vec{d}\|^2 = \min \dots$$

Least squares minimization of the sum of squares of the Euclidean distances between the template and the search surface elements.

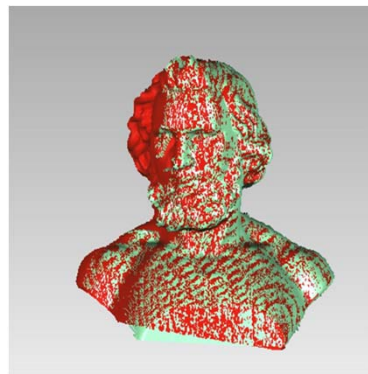
$$f(x, y, z) - e(x, y, z) = g^0(x, y, z) + \frac{\partial g^0(x, y, z)}{\partial x} dx + \frac{\partial g^0(x, y, z)}{\partial y} dy + \frac{\partial g^0(x, y, z)}{\partial z} dz \rightarrow 7\text{-parameter 3D similarity transformation}$$

$$-e(x, y, z) = g_x dt_x + g_y dt_y + g_z dt_z + (g_x a_{10} + g_y a_{20} + g_z a_{30}) dm + (g_x a_{11} + g_y a_{21} + g_z a_{31}) d\omega + (g_x a_{12} + g_y a_{23} + g_z a_{33}) d\phi + (g_x a_{13} + g_y a_{23} + g_z a_{33}) d\kappa - (f(x, y, z) - g^0(x, y, z)) \rightarrow \text{Gauss-Markov model: } -e = Ax - l, P$$

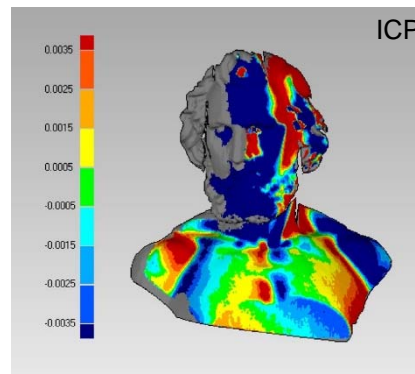
The automatic co-registration of point clouds, representing three-dimensional (3D) surfaces, is an important technique in 3D reconstruction and is widely applied in many different disciplines. An alternative approach is proposed here that estimates the transformation parameters of one or more 3D search surfaces with respect to a 3D template surface. The approach uses the nonlinear **Gauss-Helmert model**, minimizing the quadratically constrained least squares problem. This approach has the ability to match arbitrarily oriented 3D surfaces captured from a number of different sensors, on different time-scales and at different resolutions. In addition to the 3D surface-matching paths, the mathematical model allows the precision of the point clouds to be assessed after adjustment. The error behavior of surfaces can also be investigated based on the proposed approach. Some practical examples are presented and the results are compared with the iterative closest point and the linear least-squares approaches to demonstrate the performance and benefits of the proposed technique (Quelle: Ge & Wunderlich, 2016).

$$f(x, y, z) + e_f(x, y, z) = g\{x, y, z\} + e_g(x, y, z) \dots \text{condition equation with the measurement errors in all observations}$$

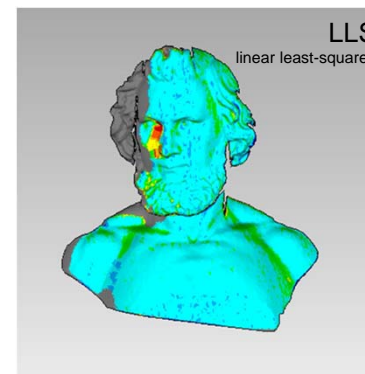
$$\sum (\|e_f\|^2 + \|g\{e_g\}\|^2) = \min \dots \text{least squares problem} \rightarrow \text{Gauss-Helmert model}$$



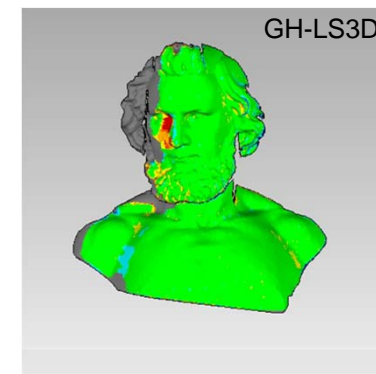
View of the final composite surface after matching using the GH-LS3D approach. The source surface is shown in red and the transformed target surface is shown in green.



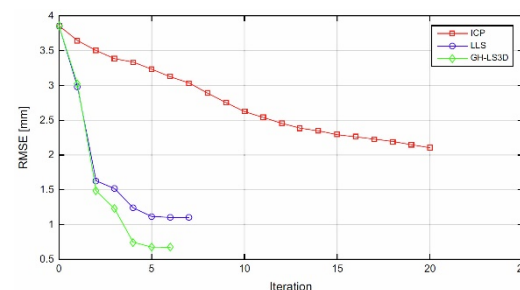
Colored residuals between the source surface and the target surface after matching with the ICP method.



Colored residuals between the source surface and the target surface after matching with the LLS approach.



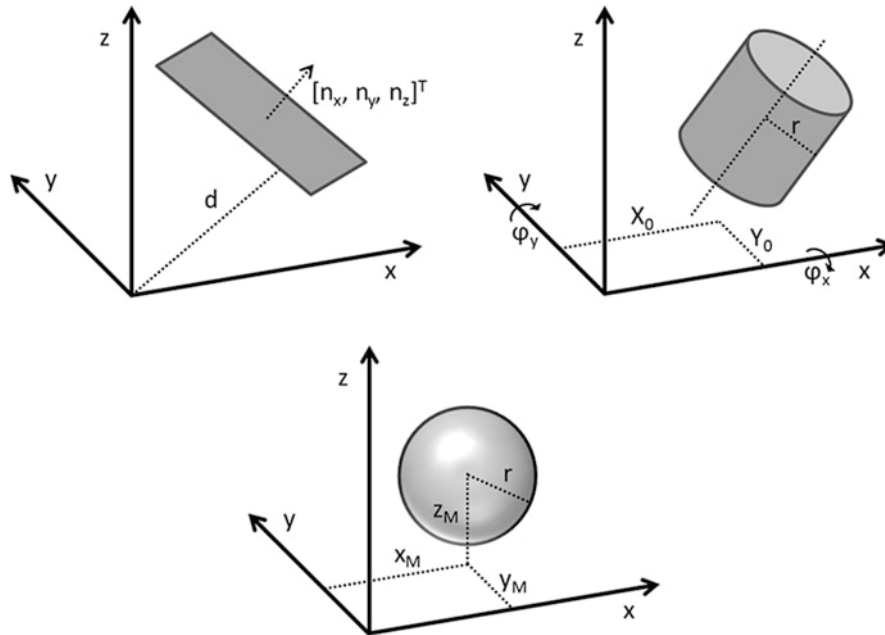
Colored residuals between the source surface and the target surface after matching with the GH-LS3D approach.



Comparison of the rate of convergence for the three different algorithms.

To accelerate the computation and to avoid memory problems in computer programs resulting from limited random access memory (RAM), the authors down-sampled the huge raw point cloud using a voxel grid filter. The whole scan volume was divided into a regular 3D voxel grid, and only one point per voxel, computed as the centroid of the points inside a grid cell, was retained.

Zunehmend existieren aber auch Verfahren, die automatisch Ebenen oder andere Objekte in den Scans detektieren und diese als Korrespondenzen für überlappende Scans verwenden (Quelle: Kuhlmann & Host, 2017).



Geometrische Primitive: Ebene, Zylinder und Kugel mit ihren Parametern
(Quelle: Kuhlmann & Holst, 2017)

- **Ebene** mit den Parametern $\mathbf{p} = [n_x, n_y, n_z, d]^T$ (Normalenvektor und geometrische Distanz):

$$f(\mathbf{x}, \mathbf{p}): n_x \cdot x + n_y \cdot y + n_z \cdot z - d = 0$$

- **Zylinder** mit den Parametern $\mathbf{p} = [X_0, Y_0, \varphi_x, \varphi_y, r]^T$ (Translation, Rotation, Radius):

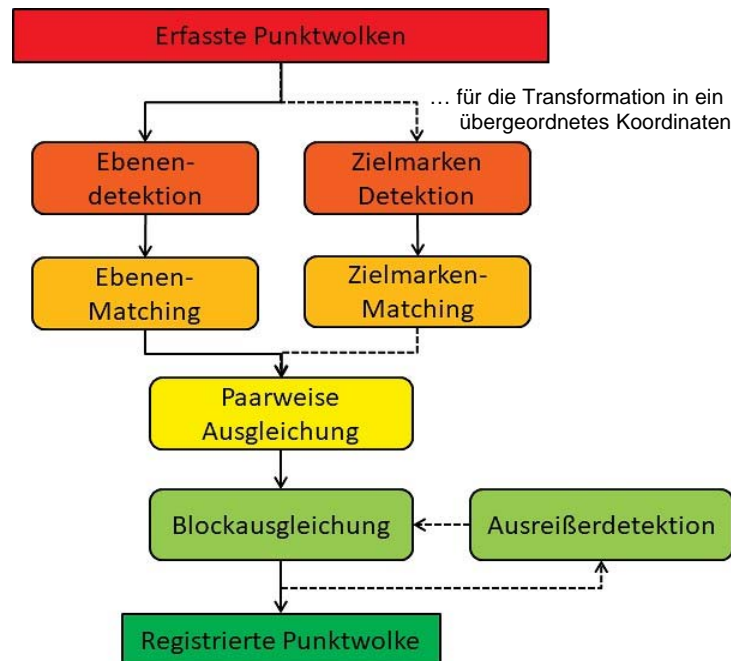
$$f(\mathbf{x}, \mathbf{p}): \left(\frac{y \cdot \cos \varphi_x + x \cdot \sin \varphi_x \cdot \sin \varphi_y + z \cdot \sin \varphi_x \cdot \cos \varphi_y - Y_0}{r} \right)^2 + \left(\frac{x \cdot \cos \varphi_y - z \cdot \sin \varphi_y - X_0}{r} \right)^2 - 1 = 0$$

- **Kugel** mit den Parametern $\mathbf{p} = [x_M, y_M, z_M, r]^T$ (Mittelpunkt und Radius):

$$f(\mathbf{x}, \mathbf{p}): (x - x_M)^2 + (y - y_M)^2 + (z - z_M)^2 - r^2 = 0$$

Parameterschätzung durch das Gauß-Helmert-Modell.

(Quelle: Wujanz et al., 2018)



Flussdiagramm von Scantra
<https://www.technet-gmbh.com/produkte/scantra/>

1. Ebenendetektion: Dadurch kann der Speicherbedarf drastisch reduziert werden, da im Ergebnis nur Ebenenparameter und deren stochastische Eigenschaften entstehen.

Segmentierungsverfahren:

- im 3D-Raum (Region growing oder RANSAC)
- im 2D-Raum (Software Scantra):
 Hierzu werden die Punkte der Punktwolke in einen Baum eingefügt, welcher anschließend alternierend entlang der Komponenten φ und θ halbiert wird. Der Zerlegungsprozess wird fortgeführt bis entweder in den betreffenden Teilbaum eine Ebene gefittet werden kann oder ein Abbruchkriterium für die Unterteilung erreicht ist. Das Ergebnis ist ein Satz von Ebenen (ca. 500 bis 1200 pro Scan), die in der Projektdatenbank abgelegt werden. Jede Ebene erhält individuelle stochastische Eigenschaften in Form einer Kovarianzmatrix, die sich aus der Ebenengleichung ergibt.

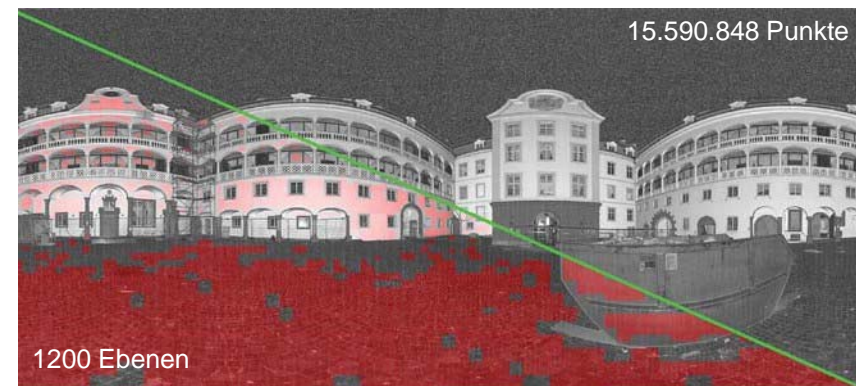


Abb.1: Intensitätsbild (oben rechts) mit den detektierten Ebenen (rot eingefärbt, unten links)
 (Quelle: Wujanz et al., 2018)

2. Ebenen-Matching und paarweise Ausgleichung: Anschließend müssen Ebenen-Korrespondenzen zwischen Paaren von Standpunkten berechnet werden. Dazu werden **geometrische Deskriptoren** verwendet, die gegenüber dem individuellen geodätischen Datum der gegebenen Ebenen invariant sind. Informationen über Nachbarschaften zwischen Punktwolken können entweder manuell oder durch Messung der relativen Position von Standpunkten definiert werden. Siehe Abbildung 2. Für alle in dieser Abbildung dargestellten Beziehungen wird eine erste Ausgleichung berechnet. Das Ergebnis ist in Abb. 3 dargestellt. Nach der paarweisen Ausgleichung der Transformationsparameter auf Grundlage korrespondierender Ebenen ist die Farbe der Pfeile grün geworden, was bedeutet, dass die Standardabweichung aller translatorischen Komponenten σ_t gleich oder kleiner 1 mm ist (vgl. Tab. 1).

| | σ_t [mm] | σ_φ [rad] | Anzahl an Korrespondenzen |
|-----|--------------------|---------------------------|---------------------------|
| 1-2 | 0.5 | 0.0027 | 386 |
| 1-5 | 0.2 | 0,0024 | 563 |
| 2-3 | 0.3 | 0.0031 | 437 |
| 2-4 | 0.2 | 0.0032 | 646 |
| 2-5 | 0.2 | 0.0029 | 568 |
| 3-4 | 0.2 | 0.0030 | 609 |
| 4-5 | 0.5 | 0.0075 | 370 |

Tab.1: Ergebnis des Ebenen-Matchings
(Quelle: Wujanz et al., 2018)

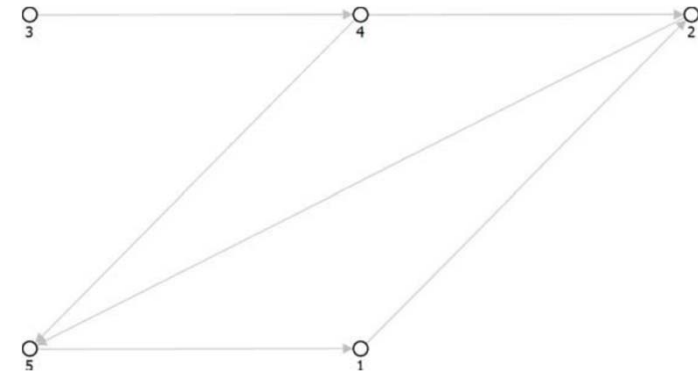


Abb. 2: Topologische Beziehungen zwischen fünf Scans
(Quelle: Wujanz et al., 2018)

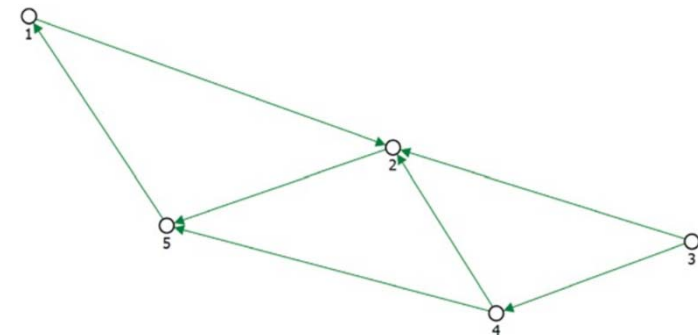


Abb. 3: Ergebnis des Ebenen-Matchings
(Quelle: Wujanz et al., 2018)

3. Blockausgleichung: Der Registrierungsprozess wird durch eine Blockausgleichung abgeschlossen, welche die relative Position und Ausrichtung aller Laserscans auf Basis redundant erfasster Bereiche optimiert. Optional kann ein Fehlererkennungsalgorithmus gestartet werden, der den Benutzer bei der Identifizierung und Behebung eventuell aufgetretener Fehler unterstützt. Das Endergebnis des Algorithmus ist eine registrierte Punktwolke mit Qualitätsmaßen aus der Blockausgleichung.

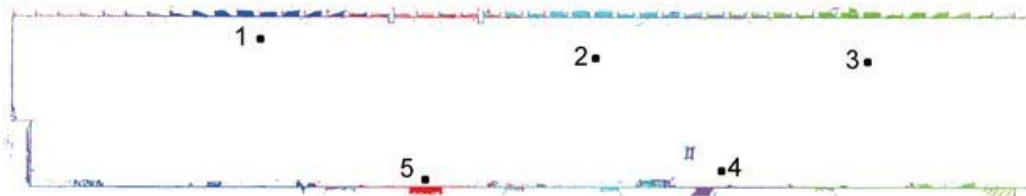


Abb. 4: Horizontalschnitt durch die registrierte Punktwolke
(Quelle: Wujanz et al., 2018)

Als hardwarebasierte Registrierung versteht sich die Bestimmung der Transformationsparameter unter Nutzung externer Sensorik. Aus Basis unterschiedlicher Sensoren ist hier eine Teilregistrierung oder eine vollständige Registrierung möglich. Kommen hierbei Sensoren zum Einsatz, die einen absoluten Bezug aufweisen (z.B. GNSS-Sensoren, Kompass, Neigungssensoren), entspricht diese (Teil-)Registrierung dann direkt einer (teilweisen) Georeferenzierung.

Hardwarebasierte Teilregistrierungen nutzen meistens **GNSS-Antennen**, **Neigungssensoren**, einen **Kompass** oder auch **Tachymetrie**. So ist durch eine einfache GNSS-Laserscanner-Kombination oder Prisma-Laserscanner-Kombination die Bestimmung der Position des Laserscanners mit lokalem oder absoluten Bezug möglich. Im Falle von GNSS-Antennen ist der absolute Bezug unmittelbar gegeben, im Falle von modernen Tachymetern, die das Prisma beobachten, liegt zunächst nur ein lokaler Bezug vor. Der absolute Bezug kann hier aber auch über tachymetrische Anschlussmessungen hergestellt werden.

Die Rotationen um die x- und y-Achse sind unter Verwendung von Neigungssensoren bestimmbar, das Azimut mit Hilfe eines Kompasses. Die Genauigkeit der auf Basis der Sensormessungen bestimmten Transformationsparameter hängen dabei jeweils immer von der Messunsicherheit der jeweils verwendeten Sensoren ab. Gemein ist all diesen Teilregistrierungen, dass nur noch die nicht auf Basis der Sensormessungen ermittelten Transformationsparameter mit Hilfe zielzeichen- oder softwarebasierter Registrierung zu bestimmen sind. (Quelle: Kuhlmann & Holst, 2017)

GNSS-gestützte Georeferenzierung ohne Neigungssensoren



Absolute Lagerung der zueinander registrierten Scans (= Gesamtblock) mit Hilfe der gemessenen Standpunktkoordinaten.

△ Mit einem zusätzlichen Pass-/Standpunkt könnte die Querneigung bestimmt werden.

Die Scan-Standpunkte liegen nahezu kollinear. $\Rightarrow \epsilon_x$ ist schlecht bzw. nicht bestimmbar.

Die relative Lage der Scan-Standpunkte erlaubt eine Horizontierung der registrierten Scans.

Vollständige hardwarebasierte Georeferenzierung

Direct geo-referencing of a static terrestrial laser scanner (Paffenholtz et al., 2010; Paffenholtz, 2012)

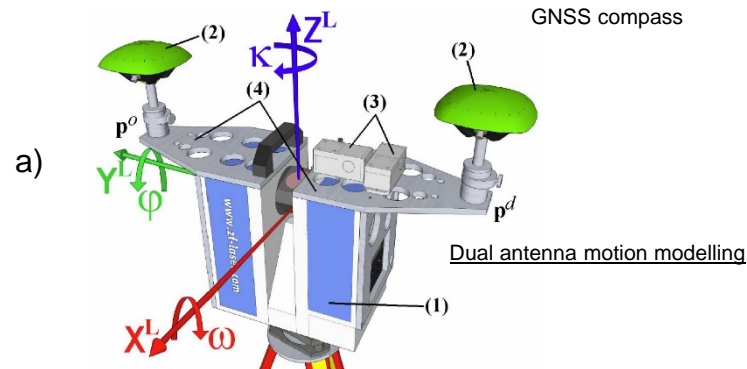


Fig. 1: Multi-sensor system: (1) Laser scanner, (2) GNSS antenna, (3) inclinometer, (4) Wing support structure. (Source: Paffenholtz, 2012)

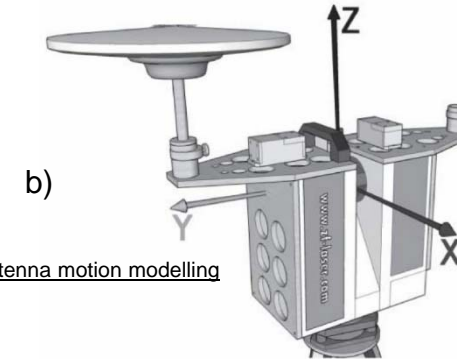


Fig. 2: Using inclinometer measurements the spatial rotation angles about the X- und Y-axis can be estimated by an adaptive extended Kalman filter (AEKF). Only one GNSS antenna is needed. (Source: Paffenholtz et al., 2010)

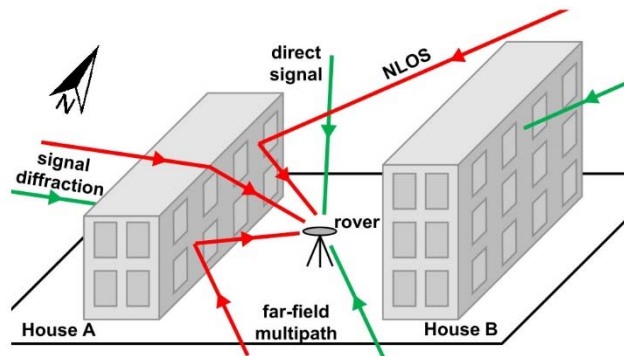


Fig. 4: Site-dependent GNSS effects: The green arrows represent the direct signal paths and the red arrows represent the reflected or diffracted signals. (Source: Zimmermann et al., 2018)

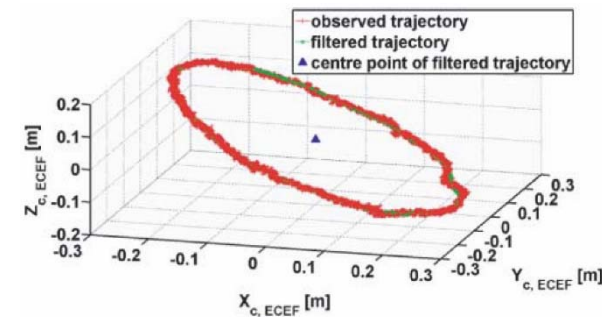


Fig. 3: Observed and filtered trajectories of the antenna reference point (ARP) in an ECEF coordinate system reduced to the centre point. (Source: Paffenholtz et al., 2010)

(Quelle: Kuhlmann & Klingbeil, 2017; Holst et al., 2015)

Die Grundidee bei der Anwendung mobiler Multisensorsysteme ist die Vermessung der Umgebung mit Hilfe von Sensoren, die sich auf einer bewegten Plattform befinden (Abb. 1).

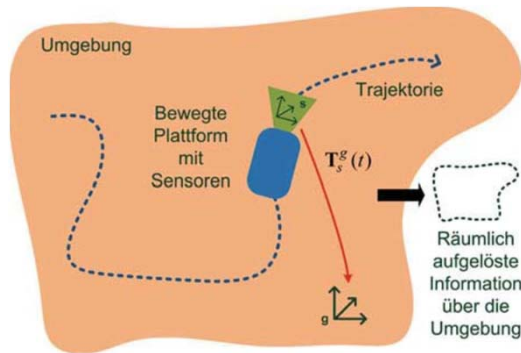


Abb. 1: Ein bewegtes Multisensorsystem nimmt während der Bewegung Daten über die Umgebung oder Objekte darin auf.

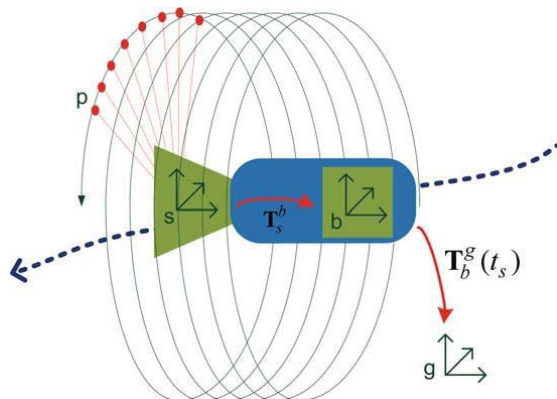


Abb. 2: Idealisiertes mobiles Multisensorsystem bestehend aus einem Profillaserscanner und einem Sensor zur Trajektorienbestimmung.

In allen Fällen der Datengewinnung ist es notwendig, dass zu den Zeitpunkten der Messung Position und Orientierung des jeweiligen Sensors bekannt ist. Nur so können die Messungen, die im Koordinatensystem des Sensors aufgenommen werden, in das übergeordnete globale Koordinatensystem überführt werden, in dem die Informationen letztendlich gesammelt und dargestellt werden sollen.

$$p^g = T_b^g(t_s) T_s^b p^s$$

Die Bestimmung der Trajektorie (Position und Orientierung der Plattform) stellt eine große Herausforderung dar. Um eine möglichst hohe Genauigkeit und Zuverlässigkeit bei unterschiedlichen Plattfomeigenschaften und Umgebungsbedingungen zu ermöglichen, werden dazu die Daten vieler Sensoren mit entsprechenden Algorithmen zusammengeführt.

(Quelle: Kuhlmann & Klingbeil, 2017; Holst et al., 2015)

Realisierung von Multisensorsystemen (\Rightarrow registrierte 3D-Punktwolke)

1. Feststellung des räumlichen Bezugs der Sensoren zueinander (Rotation und Translation)
Systemkalibrierung (T_s^b)
2. Bestimmung einer gemeinsamen Zeitbasis aller Sensoren
Zeitsynchronisation (t_s)
3. Bestimmung der Plattformtrajektorie
Trajektorienschätzung (T_b^g)

Je nach Anwendung und Messverfahren kann die Auswertung der dafür erfassten Sensordaten entweder in *Echtzeit* oder im *Postprocessing* geschehen.

| | Direkte Verfahren | Indirekte Verfahren |
|--------------------|---|---|
| Relative Verfahren | Inertialnavigation Koppelortung Odometrie GNSS-Geschwindigkeiten | Scan-Matching (Laser) Visuelle Odometrie (Kamera) |
| Absolute Verfahren | GNSS Neigungssensorik Kompass | Passpunkte/Landmarken (Laser, Kamera) Bündelausgleichung (Kamera) SLAM (Laser, Kamera) Map-Matching (Laser, Kamera, Trajektorie) |

Tab. 1: Mögliche Kategorisierung verschiedener Verfahren zur Trajektorienbestimmung.

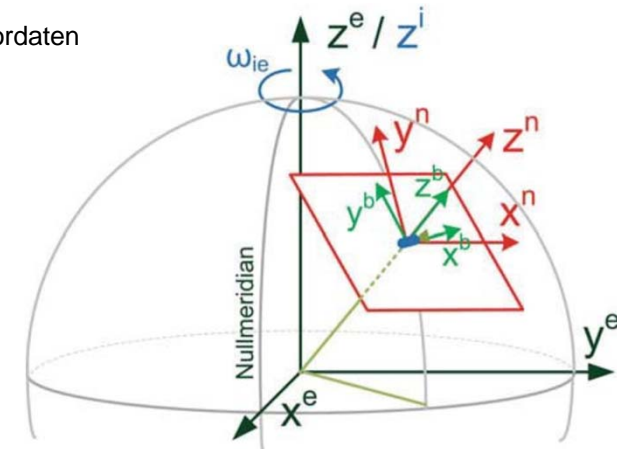


Abb. 3: Verschiedene Koordinatensysteme, die bei der Trajektorienbestimmung für sich bewegende Systeme eine Rolle spielen.

- Sensorkoordinatensystem (s)
- Körperfestes System oder Plattformsystem (b)
- Navigationssystem (n)
- Erdfestes System (e)
- Inertialsystem (i)

(Quelle: Kuhlmann & Klingbeil, 2017; Holst et al., 2015)

Systemkalibrierung

Unabhängig davon, ob die Trajektorie der Plattform in Echtzeit oder im Postprocessing berechnet wird, ob sie nur Sensoren wie GNSS und Inertialsensoren oder auch die Objektraumsensoren wie Laserscanner und Kameras verwendet, ist die **räumliche** und **zeitliche** Zuordnung aller Beobachtungen, die von den Sensoren des Systems durchgeführt werden, eine wichtige Voraussetzung für den Betrieb von Multisensorsystemen. Da fehlerhafte Zuordnungen sich als systematische Abweichungen im Endprodukt des Mobile-Mapping-Vorgangs (z.B. einer Punktwolke) äußern, wird die Herstellung dieser Zuordnung, insbesondere der räumlichen, meist Systemkalibrierung genannt. Die Kalibrierung der Einzelsensoren, also die Bestimmung sensorinterner systematischer Abweichungen, wie z.B. die Zielachs- und Kippachsabweichung bei Laserscanner, wird üblicherweise mit gesonderten Verfahren (vgl. 03_TLS_Prüfung und Kalibrierung.pdf) bestimmt.

a) Räumliche Kalibrierung

Bei der räumlichen Kalibrierung wird, wie in Abb. 4 links dargestellt, die Transformation (Rotation und Translation) eines jeden einzelnen Sensors bezüglich eines gemeinsamen Plattformkoordinatensystems bestimmt.

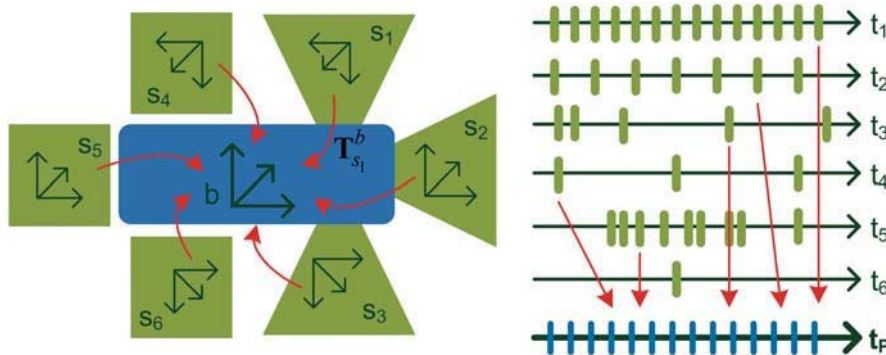


Abb. 4: Kalibrierung, räumliche und zeitliche Zuordnung der Sensordaten.

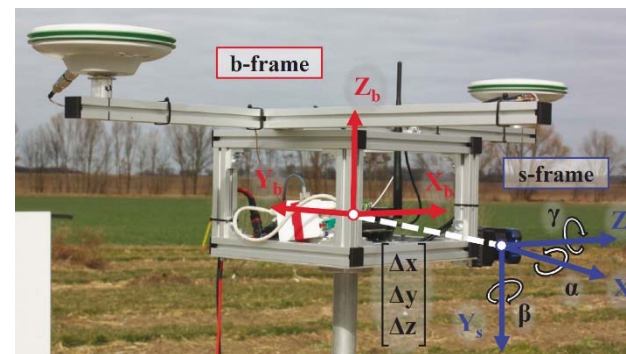
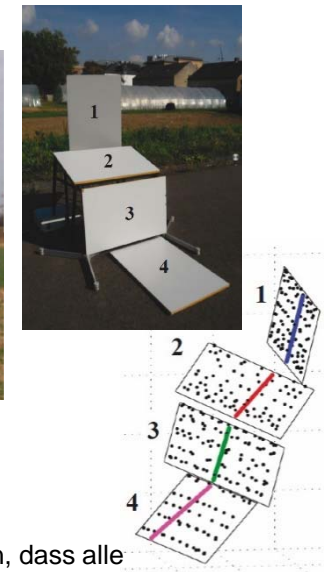


Abb. 5: Systemkalibrierung mit ebenen Flächen.
(Quelle: Heinz et al., 2015)



b) Synchronisierung der Sensordaten

Das Ziel der Synchronisierung ist es, wie in Abb. 4 rechts illustriert, jede Beobachtung so mit einem Zeitstempel zu versehen, dass alle Sensoren die gleiche Zeitbasis haben. Ein Grund für die Schwierigkeit der Synchronisierung liegt in den vielen unterschiedlichen Messprinzipien und Schnittstellen der einzelnen Sensoren. Die Rolle des Frequenznormals (t_P) wird bei mobilen Multisensorsystemen häufig durch das GPS eingenommen. Die meisten GPS-Empfänger stellen ein sog. PPS-Signal (Puls-Per-Second) zur Verfügung.

| | Zielzeichenbasiert | Softwarebasiert | Hardwarebasiert |
|------------------|---|---|--|
| Vorteile | <ul style="list-style-type: none"> – Konfiguration der Zielzeichen beeinflussbar – Qualität der Zielzeichen hoch und bekannt – unabhängig von Objekt und Überlappung | <ul style="list-style-type: none"> – hohe Überbestimmung – Qualität einzelner Korrespondenzen über hohe Redundanz bewertbar – keine zusätzliche Hardware notwendig | <ul style="list-style-type: none"> – hohe Überbestimmung bei kontinuierlichen Sensormessungen – ableitbare Qualitätsinformation aus Sensormessungen – unabhängig von Objekt und Überlappung |
| Nachteile | <ul style="list-style-type: none"> – Aufwand für Auslegung einer geeigneten Anzahl von Zielzeichen – Begehrbarkeit muss sichergestellt sein – geringe Überbestimmung bei vertretbarem Aufwand – Qualität der Registrierung nur bei hoher Anzahl von Zielzeichen zuverlässig bewertbar | <ul style="list-style-type: none"> – Konfiguration der Korrespondenzen kaum beeinflussbar (bei geringer Überlappung Extrapolationseffekt) – Qualität der Korrespondenzen kaum beeinflussbar – abhängig von Objekt (Struktur), Überlappung und Punktauflösung der Einzelscans | <ul style="list-style-type: none"> – Kosten durch zusätzliche Sensoren – Aufbau eines MSS notwendig (Synchronisation, Auswertung, ...) – abhängig von der Qualität der Sensoren (z.B. Abschattung GNSS) |

Tabelle: Vergleich zielzeichenbasierter, softwarebasierter und hardwarebasierter Verfahren zur Registrierung und Georeferenzierung
(Quelle: Kuhlmann & Holst, 2017)

Auswahl:

- Arun, K.S., Hilden, H.M., Blostein, S.D., 1987: Least-squares fitting of two 3-D point sets. IEEE Transactions on Pattern Analysis and Machine Intelligence, 9(5), 698-700.
- Bae, K.-H., Lichti, D.D., 2008: A method for automated registration of unorganised point clouds. ISPRS Journal of Photogrammetry & Remote Sensing, 63, 36-54.
- Barnea, S., Filin, S., 2007: Registration of terrestrial laser scans via image based features. Int. Arch. Photogramm. Remote Sens., 36(3-W52), 26-31.
- Barnea, S., Filin, S. 2008: Keypoint based autonomous registration of terrestrial laser point-clouds. ISPRS Journal of Photogrammetry & Remote Sensing, 63, 19-35.
- Beinat, A., Crosilla, F., Sepic, F., 2006: Automatic morphological pre-alignment and global hybrid registration of close range images. Int. Arch. Photogramm. Remote Sens., 36(5), (on CD-ROM).
- Besl, P.J., McKay, N.D., 1992: A Method for Registration of 3-D Shapes. IEEE Transactions on Pattern Analysis and Machine Intelligence, 14(2), 239-256.
- Böhm, J., Becker, S., 2007: Automatic marker-free registration of terrestrial laser scans using reflectance features. In: Proceedings of 8th Conference on Optical 3D Measurement Techniques, Zurich, Switzerland, 338-344.
- Chen, Y., Medioni, G., 1991: Object Modeling by Registration of Multiple Range Images. Proceedings of the 1991 IEEE International Conference on Robotics and Automation, Sacramento, California – April 1991, 2724-2729.
- Dold, C., Brenner, C., 2006: Registration of terrestrial laser scanning data using planar patches and image data. ISPRS Volume XXXVI, Part 5, ISPRS Commission V Symposium, Dresden, 25-27 September 2006, 78-83.
- Fischler, M.A., Bolles, R.C., 1981: Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography. Communications of ACM, Vol. 24, No. 6, 381-395.
- Grussenmeyer, P., Landes, T., Doneus, M., Lerma, J.L., 2016: Basics of Range-Based Modelling Techniques in Cultural Heritage 3D Recording. In: Stylianidis, E., Remondino, F. (Eds.), 2016, 3D Recording, Documentation and Management of Cultural Heritage, Whittles Publishing, 305-368.
- Grün, A., Akca, D., 2005: Least squares 3D surface and curve matching. ISPRS Journal of Photogrammetry & Remote Sensing, 59, 151-174.
- Handbuch RiSCAN PRO, 2009: rscan_pro-1.5.2sp2_Manual_Riegl_LMS-Z620.pdf
- Heinz, E., Eling, C., Wieland, M., Klingbeil, L., Kuhlmann, H., 2015: Development, Calibration and Evaluation of a Portable and Direct Georeferenced Laser Scanning System for Kinematic 3D Mapping. Journal of Applied Geodesy, 9(4), 227-243.
- Holst, C., Kuhlmann, H., Paffenholz, J.-A., Neumann, I., 2015: TLS im statischen, stop & go sowie kinematischen Einsatz. Terrestrisches Laserscanning 2015 (TLS 2015), Schriftenreihe des DVW, Band 81 (vergriffen).
- Horn B.K.P., 1986: Closed-form solution of absolute orientation using unit quaternions. Journal of the Optical Society of America, 4, 629-642.
- Janßen, J., Holst, C., Kuhlmann, H., 2018: Registrierung mit Targets: Wie genau ist das?. In: DVW e.V. (Hrsg.), Terrestrisches Laserscanning 2018 (TLS 2018), Schriftenreihe des DVW, 93, Wißner-Verlag, 75-94.

- Kraus, K., 1996: Photogrammetrie. Verfeinerte Methoden und Anwendungen. Band 2, Dümmler Verlag, 488 S.
- Kraus, K., 2004: Photogrammetrie. Geometrische Informationen aus Photographien und Laserscanneraufnahmen. 7. Auflage, deGruyter Lehrbuch, 516 S.
- Kraus, K., 2007: Photogrammetry. Geometry from Images and Laser Scans. Translated by Hartley, I. & Kyle S., second edition, Walter de Gruyter, 459 p.
- Kraus, K., 2013: Laser-Scanning – Ein Paradigma-Wechsel in der Photogrammetrie. https://publik.tuwien.ac.at/files/PubDat_120149.pdf
- Kuhlmann, H., Holst, C., 2017: Flächenhafte Abtastung mit Laserscanning. Messtechnik, flächenhafte Modellierung und aktuelle Entwicklungen im Bereich des terrestrischen Laserscannings. In: Schwarz, W. (Hrsg.), Ingenieurgeodäsie, Springer Spektrum, 167-212.
- Kuhlmann, H., Klingbeil, L., 2017: Mobile Multisensorsysteme. In: Schwarz, W. (Hrsg.), Ingenieurgeodäsie, Springer Spektrum, 93-129.
- Lichti, D., Skaloud, J., 2010: Registration and Calibration. In: Vosselman, G., Maas, H.-G. (Eds.), 2010, Airborne and Terrestrial Laser Scanning, Whittles Publishing, CRC Press, Taylor & Francis Group, 83-133.
- Lowe, D.G., 2004: C. International Journal of Computer Vision, 60(2), 91-110.
- Luhmann, T., 2018: Nahbereichsphotogrammetrie. Grundlagen – Methoden – Beispiele. 4., neu bearbeitete und erweiterte Auflage, Wichmann, 783 S.
- Mayer, J., Metzler, B., 2018: Simultane Lokalisierung und Kartenerstellung (SLAM) im terrestrischen Laserscanning. In: DVW e.V. (Hrsg.), Terrestrisches Laserscanning 2018 (TLS 2018), Schriftenreihe des DVW, 93, Wißner-Verlag, 109-115.
- Paffenholz, J.-A., Alkhatib, H., Kutterer, H., 2010: Direct geo-referencing of a static terrestrial laser scanner. Journal of Applied Geodesy, 4, 115-126.
- Paffenholz, J.-A., 2012: Direct geo-referencing of 3D point clouds with 3D positioning sensors. DKG, Reihe C, Dissertationen, Heft Nr. 689,
- Pfeifer, N., Mandlbürger, G., Glira, P., 2017: Laserscanning. In: Heipke, C. (Hrsg.), Photogrammetrie und Fernerkundung, Springer Reference Naturwissenschaften, Springer Spektrum, 431-481.
- Ullrich, A., Fürst, C., 2017: Vollautomatischer Ansatz für die Onboard-Datenregistrierung im terrestrischen Laserscanning. In: DVW e.V. (Hrsg.), Terrestrisches Laserscanning 2017 (TLS 2017), Schriftenreihe des DVW, 88, Wißner-Verlag, 145-153.
- Welsch, W., Heunecke, O., Kuhlmann, H., 2000: Auswertung geodätischer Überwachungsmessungen. Reihe Handbuch Ingenieurgeodäsie, Herbert Wichmann Verlag, 247 S.
- Wieser, A., Wunderlich, T., 2017: Tutorial „Monitoring mit Terrestrischem Laserscanning (TLS)“, Ingenieurvermessungskurs '17, 25.4.2017, TU Graz, schriftliche Unterlagen.
- Williams, J., Bennamoun, M., 2001: Simultaneous Registration of Multiple Corresponding Point Sets. Computer Vision and Understanding, 81, 117-142.
- Wujanz, D., Schaller, S., Gielsdorf, F., Gründig, L., 2018: Ebenenbasierte Registrierung von mehreren Tausend Laserscans. In: DVW e.V. (Hrsg.), Terrestrisches Laserscanning 2018 (TLS 2018), Schriftenreihe des DVW, 93, Wißner-Verlag, 95-107.
- Zimmermann, F., Holst, C., Klingbeil, L., Kuhlmann, H., 2018: Accurate georeferencing of TLS point clouds with short GNSS observation durations even under challenging measurement conditions. Journal of Applied Geodesy, 12(4), 289-301.