

Messunsicherheitsanalyse mittels virtueller Streifenprojektionstechnik auf Grundlage von Polygonmodellen

Measurement Uncertainty Analysis Using Virtual Fringe Projection Based on Polygon Models

Klaus Haskamp, Markus Kästner, Rüdiger Gillhaus, Eduard Reithmeier, Institut für Mess- und Regelungstechnik, Hannover

Zusammenfassung Virtuelle Sensorsysteme können parallel zur Bauteilkonstruktion zur Messbarkeitsprüfung und für Messunsicherheitsuntersuchungen verwendet werden. Beispielsweise ist es möglich, zu entscheiden, ob die funktionsrelevanten Bauteilflächen mit der geforderten Messunsicherheit erfasst werden können. In diesem Beitrag wird ein Verfahren vorgestellt, mit dem es möglich ist, reale Messunsicherheiten mit Hilfe einer numerischen Computersimulation abzuschätzen. Weiterhin wird die abgeschätzte Unsicherheit mit empirisch abgeschätzten Unsicherheiten verglichen, um die Güte des virtuellen Messsystems zu verifizieren. ▶▶▶ **Sum-**

mary Virtual sensor systems can be applied to verify the measurability and for measurement uncertainty inspections. This can be done parallel to the component construction. It is possible, for example, to decide if all relevant workpiece surfaces can be measured with the required measurement uncertainty. In this article a method for the estimation of real measurement uncertainties is presented using a numerical computer simulation. To verify the quality of the virtual measurement system, the estimated uncertainties are in addition compared to uncertainties estimated using empirical methods.

Schlagwörter Virtuelle Streifenprojektion, virtueller Messablauf, Abschätzung von Messabweichungen, Unsicherheitsanalyse

▶▶▶ **Keywords** Virtual fringe projection, virtual measurement, estimation of measurement uncertainty, uncertainty analysis

1 Einleitung

An den Produktionsprozess schließt sich die Qualitätsprüfung an. Diese muss gewährleisten, dass alle signifikanten Geometriebereiche in dem vom Konstrukteur vorgegebenen Toleranzbereich geprüft werden. Hierzu müssen umfangreiche Unsicherheitsanalysen an einem Referenzbauteil bzw. Normal durchgeführt werden, um abzuschätzen, wie hoch die Genauigkeit der einzusetzenden Messgeräte beim Erfassen der Geometriebereiche bzw. wie groß die verbleibende Messunsicherheit ist. Damit verbunden ist ein hoher Zeit- und Kostenaufwand, der sich durch eine numerische Simulation des Messgerätes minimieren lässt. Mit Hilfe eines

solchen virtuellen Systems kann anschließend der gesamte Messprozess simuliert werden. In Kombination mit Monte-Carlo-Methoden ist es dann möglich, eine Unsicherheitsanalyse, beispielsweise anhand eines CAD-Modells durchzuführen. Um dies realisieren zu können, muss neben dem virtuellen Messsystem auch der Messprozess virtuell im Rechner beschrieben werden.

Nach der Modellierung des Messsystems müssen alle signifikanten Systemparameter durch eine Kalibrierung hinreichend genau identifiziert werden. Für die Modellierung von optischen Komponenten bzw. Messgeräten bietet sich beispielsweise die Software „ZEMAX“ an. Im vorliegenden Fall soll der Messprozess als numerische

Computersimulation auf Basis von „C++“ umgesetzt werden, da das virtuelle Streifenprojektionssystem in eine auf „C++“ basierende Gesamtsimulation eingebunden werden soll.

Die Beschreibung des virtuellen Messprozesses eines Streifenprojektionssystems und die Simulation realer Messunsicherheiten bzw. Messabweichungen sollen Gegenstand dieses Beitrags sein. Die Verifikation der Simulation wird über einen Vergleich der Simulationsergebnisse mit dem realen System erfolgen. Hierfür wird die Messunsicherheit mit dem virtuellen System durch das Anwenden von Monte-Carlo-Methoden erzeugt. Die Abweichung des realen Systems ergibt sich aus Wiederholmessungen. Als Messobjekt wird eine Kugel eingesetzt werden.

2 Virtuelles Abbild des Streifenprojektionssensors

Um den Messprozess virtuell abbilden zu können, wird in einem ersten Schritt ein physikalisches Modell des Streifenprojektionssystems verwendet. Beschreibungsgrundlage der mathematischen Zusammenhänge stellt die projektive Geometrie dar, die voraussetzt, dass das System auf der geometrischen Optik basiert [1;2]. Bild 1 zeigt sowohl den geometrischen Aufbau des Streifenprojektionssystems als auch Lage und Orientierung aller Koordinatensysteme, die zur Formulierung der linearen Abbildungen notwendig sind. Das hier verwendete Streifenprojektionssystem besteht aus einer Kamera und einem Projektor, deren Modellierung im Folgenden vorgestellt werden.

Die nachgestellten Indizes „K“ und „P“ kennzeichnen die Zugehörigkeit der Koordinatensysteme zu der Kamera bzw. zu dem Projektor. Das Koordinatensystem KS_{Obj} stellt das Objektkoordinatensystem dar und soll als Bezugskoordinatensystem verwendet werden [3].

Mit den Koordinatensystemen kann die Transformation von homogenen 3D-Koordinaten $\mathbf{X} = [x, y, z, 1]^T$ auf einen 2D-Bildpunkt der Kamera $\boldsymbol{\eta} = [i, j, 1]^T$ bzw. des Projektors $\boldsymbol{\eta} = [\phi, \psi, 1]^T$ beschrieben werden. Die erforderlichen Transformationsschritte sind in Bild 2 dargestellt. Da ein Projektor als inverse Kamera betrachtet werden kann, wird die Koordinatentransformation am Beispiel der Kamera näher erläutert [4].

Im ersten Schritt wird der 3D-Punkt $\mathbf{X} = [x, y, z, 1]^T$ durch eine homogene Transformationsmatrix \mathbf{T}_{Obj}^{Komp} ins Komponentenkoordinatensystem $KS_{Komp,K}$ transformiert [5]:

$$\mathbf{T}_{Obj}^{Komp} = \begin{bmatrix} \mathbf{R}_{Obj}^{Komp} & \mathbf{r}_{Obj} \\ \mathbf{0}^T & 1 \end{bmatrix}$$

Dabei stellt \mathbf{R}_{Obj}^{Komp} den Rotationsanteil und \mathbf{r}_{Obj} einen Vektor vom Ursprung des Objekt- zum Ursprung des Komponentenkoordinatensystems dar.

Anschließend erfolgt eine telezentrische Projektion \mathbf{K}_Z in das Bildkoordinatensystem $KS_{Bild,K}$ [1]:

$$\mathbf{K}_Z = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Die nichtlinearen Verzeichnungsfehler lassen sich durch die Verzeichnungsfunktion \mathbf{L} berücksichtigen [6;7]:

$$\mathbf{X}_{Bild'} = \mathbf{L}(\mathbf{X}_{Bild}, \mathbf{k})$$

wobei \mathbf{k} einen Parametervektor darstellt. In der Regel werden drei Verzeichnungsarten unterschieden [6;7]:

- radialsymmetrisch $\delta \mathbf{x}_r = [\delta x_r \ \delta y_r]^T$,
- radialasymmetrisch $\delta \mathbf{x}_d = [\delta x_d \ \delta y_d]^T$ und
- Affinität und Scherung $\delta \mathbf{x}_a = [\delta x_a \ \delta y_a]^T$.

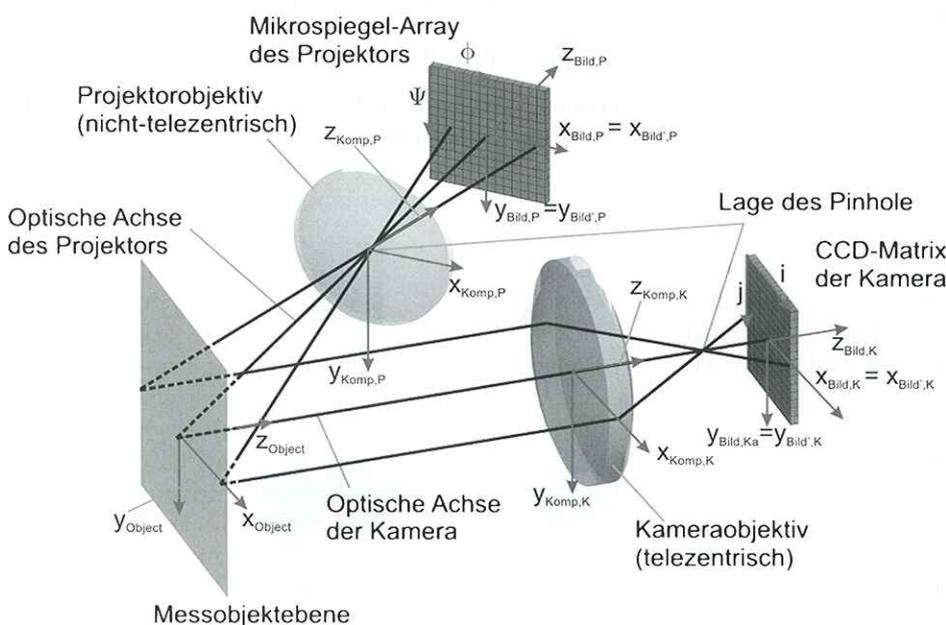


Bild 1 Modell des Streifenprojektionssystems.

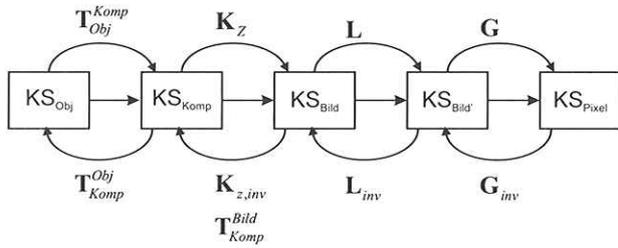


Bild 2 Transformationsschritte.

Dabei nehmen die Vektorfunktionen folgende Gestalt an:

$$\begin{aligned} \delta x_r &= x \cdot (k_1 \cdot (r^2 - r_0^2) + k_2 \cdot (r^4 - r_0^4) + k_3 \cdot (r^6 - r_0^6)), \\ \delta y_r &= y \cdot (k_1 \cdot (r^2 - r_0^2) + k_2 \cdot (r^4 - r_0^4) + k_3 \cdot (r^6 - r_0^6)), \\ \delta x_d &= p_1 \cdot (3 \cdot x^2 + y^2) + 2 \cdot p_2 \cdot x \cdot y, \\ \delta y_d &= 2 \cdot p_1 \cdot x \cdot y + p_2 \cdot (x^2 + 3 \cdot y^2), \\ \delta x_a &= A_1 \cdot x + A_2 \cdot y \quad \text{und} \\ \delta y_a &= 0. \end{aligned}$$

$k_1, k_2, k_3, p_1, p_2, A_1$ und A_2 beschreiben die Verzeichnung und werden in dem Vektor \mathbf{k} zusammengefasst.

Zum Schluss werden die Punkte durch die Matrix \mathbf{G} ins Pixelkoordinatensystem $KS_{Pixel,K}$ abgebildet und anschließend über die Pixelabmessungen in diskrete, ganzzahlige Koordinaten umgewandelt:

$$\mathbf{G} = \begin{bmatrix} 0 & 1 & \frac{h_{CCD}}{2} \\ 1 & 0 & \frac{b_{CCD}}{2} \\ 0 & 0 & 1 \end{bmatrix},$$

wobei b_{CCD} und h_{CCD} die Abmessungen der Pixelfläche des Kamerachips darstellen.

Die Transformationsschritte für die Abbildung vom Objektkoordinatensystem auf das Pixelkoordinatensystem des Projektors erfolgen analog zu denen der Kamera. Unterschiede ergeben sich lediglich bei den Werten der Parameter.

Sämtliche Modellparameter wurden durch eine Kalibrierprozedur identifiziert, so dass eine vollständige Beschreibung des virtuellen Messsystems vorliegt.

3 Virtueller Messablauf

Nach der Beschreibung des virtuellen Messsystems müssen die Messobjekte mathematisch beschrieben werden. Anschließend kann ein virtueller Messablauf mit dem Ziel der Messunsicherheitsanalyse von flächenhaften Messdatensätzen erfolgen.

3.1 Darstellung der Messobjekte

Ausgangspunkt der Simulation des Messablaufs ist neben der Formulierung des Messsystems die virtuelle Beschreibung des Messobjekts. Hierfür bieten sich sowohl 3D-Splines als auch Polygone an. Während der Vorteil von 3D-Splines in der besseren Darstellungsform von

Freiformflächen liegt, können Objekte durch Polygone einfach und kompakt dargestellt werden, sodass die Wahl auf die Polygondarstellung gefallen ist.

Ein Polygon ist eine geschlossene Figur und wird durch ein Tupel P von n Punkten beschrieben

$$P := [P_1, P_2, \dots, P_n]$$

mit

$$P_i \in \mathbb{R}^3$$

Es existiert eine Vielzahl an Polygonfiguren, von denen einige zweidimensional in Bild 3 dargestellt sind. In diesem Fall sollen alle Polygone eine Dreiecksgestalt aufweisen.

Der Nachteil in der Anwendung von Polygonen liegt in der eingeschränkten Genauigkeit der nachgebildeten Oberfläche und tritt insbesondere bei gekrümmten Oberflächen auf. Erst durch eine relativ hohe Anzahl an Dreiecken kann hier eine ausreichende Genauigkeit und damit verbunden eine gute Nachbildung des Objekts erreicht werden. In Bild 4 ist die Abhängigkeit der Genauigkeit von Messobjekten von der Anzahl der Polygone am Beispiel einer Kugelkappe gezeigt. Ein Objekt sollte somit mit einer möglichst hohen Anzahl von Punkten diskretisiert werden, um eine möglichst gute Nachbildung des Objekts zu erreichen.

Es liegt eine Beschreibung der Kugel durch Dreiecke vor. Diese Dreiecke stellen einen Satz von Polygonen dar und sollen in der Liste L abgelegt werden, wobei jedem Polygon eine Indexnummer I ($I \in \mathbb{N}^{+*}$) zugeordnet wird. Weiterhin wird für jedes Dreieck der Flächenschwerpunkt \mathbf{s} und der Normalenvektor \mathbf{n} im Objektkoordinatensystem berechnet, so dass die folgende Liste L entsteht:

$$L = \left\{ \begin{array}{l} \mathbf{P}_{1,1}, \mathbf{P}_{2,1}, \mathbf{P}_{3,1}, \mathbf{s}_1, \mathbf{n}_1 \\ \mathbf{P}_{1,2}, \mathbf{P}_{2,2}, \mathbf{P}_{3,2}, \mathbf{s}_2, \mathbf{n}_2 \\ \dots \\ \mathbf{P}_{1,i}, \mathbf{P}_{2,i}, \mathbf{P}_{3,i}, \mathbf{s}_i, \mathbf{n}_i \\ \dots \\ \mathbf{P}_{1,N}, \mathbf{P}_{2,N}, \mathbf{P}_{3,N}, \mathbf{s}_N, \mathbf{n}_N \end{array} \right\}$$

wobei $\mathbf{p}_{k,i}$ den Punkt k des Polygons i und N die Anzahl der Polygone darstellt.

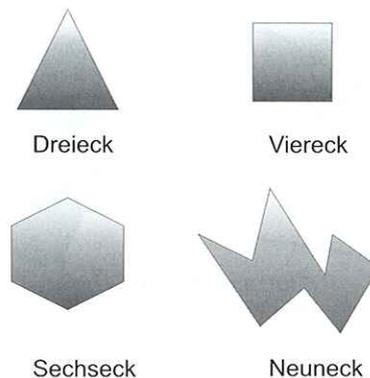


Bild 3 Verschiedene Polygone.

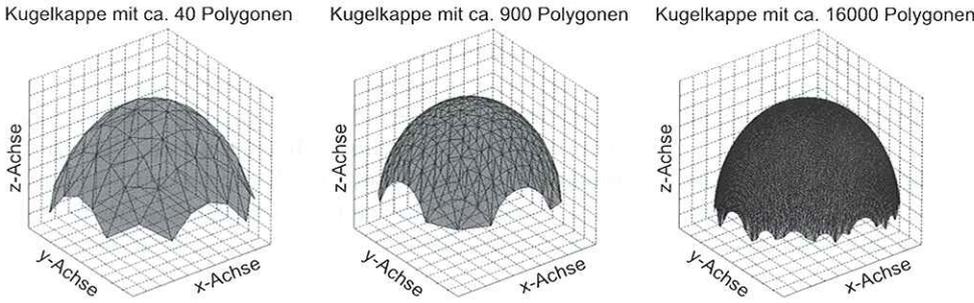


Bild 4 Darstellung der Messobjekte durch Polygone.

3.2 Raytracing

Der virtuelle Messablauf ist als Raytracing-Programm umgesetzt worden. Hierfür werden Strahlen verfolgt, die von der Pixelfläche der Kamera aus in Richtung des Messobjekts verlaufen (siehe Bild 1).

Da eine Gerade einen Körper mehr als einmal schneiden kann, wird für jeden als Gerade dargestellten Lichtstrahl und für jedes Polygon geprüft, ob ein gültiger Schnittpunkt vorliegt. Die Prüfung teilt sich dabei in zwei Schritte auf. Zunächst wird der Schnittpunkt der Ebene, die durch die drei Punkte des Polygons aufgespannt wird, und der Gerade berechnet. Falls der Punkt Element der Ebene ist, wird mit Hilfe des Umlaufwinkels θ entschieden, ob dieser Punkt auch innerhalb des Dreiecks liegt. Das ist dann der Fall, falls θ gleich 360° ist (siehe Bild 5):

$$\theta = \sum_{i=1}^3 \theta_i \begin{cases} = 360^\circ & \Rightarrow \text{Punkt liegt in Polygon} \\ \neq 360^\circ & \Rightarrow \text{Punkt liegt nicht in Polygon} \end{cases}$$

Liegen jetzt zwei oder mehr Schnittpunkte vor, dann wird der Punkt in der Simulation weiter verwendet, der am dichtesten an der Kamera liegt, da der weiter bzw. die weiter entfernt liegenden Punkte durch den Körper verdeckt und somit von der Kamera nicht erfasst werden können.

Für die Konstruktion der Geraden G , die vom Schnittpunkt $\mathbf{X}_{Schnitt}$ in Richtung des Projektors verläuft, wird die Lage des Pinholes des Projektors verwendet (siehe Bild 5):

$$G: \mathbf{X} = \mathbf{X}_{Schnitt} + r \cdot (\mathbf{X}_{Pinhole} - \mathbf{X}_{Schnitt}) \quad r \in \mathbb{R}$$

mit $\mathbf{X}_{Pinhole}$ als Lage des Pinhole im Objektkoordinatensystem. Im letzten Schritt erfolgt die Berechnung des Durchstoßpunktes von der Geraden G mit der Pixelfläche des Projektors. Dabei werden die Abmessungen der Matrix dadurch berücksichtigt, dass Punkte, die außerhalb der Fläche liegen, auf „-1“ gesetzt und in der Simulation somit verworfen werden. Das bedeutet, dass ein Kamerapixel, für das kein gültiger Punkt auf dem Projektor bzw. keine Phase existiert, nicht ausgewertet wird.

Am Ende der Strahlenverfolgung liegt für jeden Kamerapixel $\{i, j\}$ eine Information vor, welches Pixel der Projektorfläche das Kamerapixel beleuchtet. Weiterhin liegt eine Aussage darüber vor, welches Polygon I der Liste L der Lichtstrahl geschnitten hat:

- $\{i, j, \phi_{i,j}, \psi_{i,j}, I_{i,j}\}$ für gültige Kamerapixel,
- $\{i, j, -1, -1, -1\}$ für ungültige Kamerapixel.

Dabei kennzeichnen die Indizes $\{i, j\}$ die Zugehörigkeit zu dem Kamerapixel mit den Koordinaten $\{i, j\}$. Nach

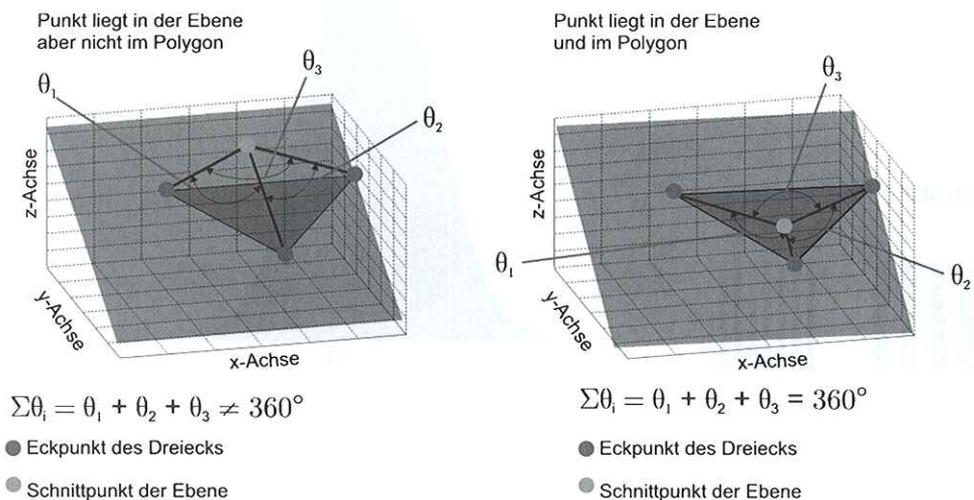


Bild 5 Schnittpunkt der Strahlen mit den Polygonen.

der Strahlensimulation erfolgt der eigentliche Messablauf, welcher aus den folgenden Schritten besteht:

- Projektion und Messung der Graycode-Sequenz,
- Projektion und Messung der Phase-Shift-Sequenz und
- Ermittlung der Projektorphase.

Da für jeden Kamerapixel die Zuordnung zu einer Phase existiert, kann aus einer virtuellen, im Rechner erzeugten Beleuchtungssequenz, bestehend aus Graycode und Phase-Shift, der „gemessene“ Phasenwert $\phi_{i,j,Mess}$ des Kamerapixels simuliert werden. Bei Kamerapixeln, deren Phasenwert $\phi_{i,j} = -1$ ist, wird

$$\phi_{i,j,Mess} = -1$$

gesetzt.

Die virtuellen Muster liegen dabei in Form von Matrizen vor (siehe Bild 6). Die von der Kamera gemessenen Intensitätswerte werden über den ganzzahligen Phasenwert $\phi_{i,j}$ aus der Matrix ausgelesen und anschließend über Bildverarbeitungsmethoden zu einem Phasenwert $\phi_{i,j,Mess}$ verrechnet. Dies bietet die Möglichkeit, Störeffekte wie Bildrauschen im Hinblick auf die Robustheit der Bildverarbeitungsmethoden bzw. der Phasenauswertung oder die Messgenauigkeit zu untersuchen. Nach der Phasenauswertung stehen, wie bei dem realen System, Wertepaare der Form:

$$\{i, j, \phi_{i,j,Mess}\}$$

zur Verfügung, so dass der 3D-Messpunkt

$$\mathbf{X}_{i,j,Mess} = [x_{i,j,Mess}, y_{i,j,Mess}, z_{i,j,Mess}]^T$$

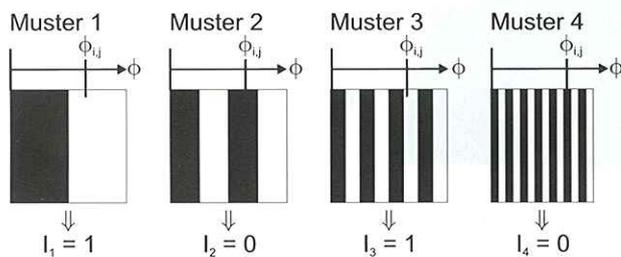
aus der Lichtebene, charakterisiert durch $\phi_{i,j,Mess}$, und der Lichtgeraden, charakterisiert durch i und j , rekonstruiert werden kann. Für Phasenwerte $\phi_{i,j,Mess} = -1$ wird der Messpunkt zu

$$\mathbf{X}_{i,j,Mess} = [NaN, NaN, NaN]^T; \quad NaN = \text{Not a Number}$$

gesetzt.

Insgesamt steht somit eine Liste von Datenpunkten für weitere Betrachtungen zur Verfügung:

$$\{i, j, \phi_{i,j}, \psi_{i,j}, I_{i,j}, \phi_{i,j,Mess}, [x_{i,j,Mess}, y_{i,j,Mess}, z_{i,j,Mess}]\}.$$



1 = Kamera beleuchtet
0 = Kamera unbeleuchtet

Bild 6 Prinzip der virtuellen Projektion des Graycodes.

3.3 Berechnung der flächenhaften Messabweichung

Für die Berechnung der flächenhaften Messabweichung wird zunächst zu jedem Punkt $\mathbf{X}_{i,j,Mess}$ der Abstand d_{ij} zu dem Polygon $I_{i,j}$ berechnet. Hierzu wird der Durchstoßpunkt \mathbf{X}_{Durch} der Geraden G , die durch den Messpunkt $\mathbf{X}_{Mess,i,j}$ und die Normale \mathbf{n}_I des Polygons $I_{i,j}$ beschrieben wird, und der Ebene E des Polygons $I_{i,j}$ berechnet (siehe Bild 8):

$$\left. \begin{aligned} G: \mathbf{X} &= \mathbf{X}_{i,j,Mess} + r \cdot \mathbf{n}_I \\ E: \mathbf{X} &= \mathbf{n}_I \cdot \mathbf{X} = \mathbf{n}_I \cdot \mathbf{s}_I \end{aligned} \right\} \Rightarrow \mathbf{X}_{Durch}$$

In der Regel kann einem Polygon, das innerhalb des Messvolumens liegt, mehr als ein Schnittpunkt und somit mehr als ein Abstand zugeordnet werden, was in Bild 7 ersichtlich ist. Auf Basis dieser Abstände wird für jedes

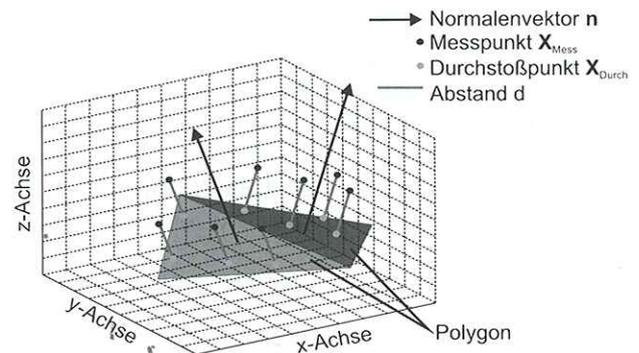


Bild 7 Prinzip zur flächenhaften Messunsicherheitsanalyse.

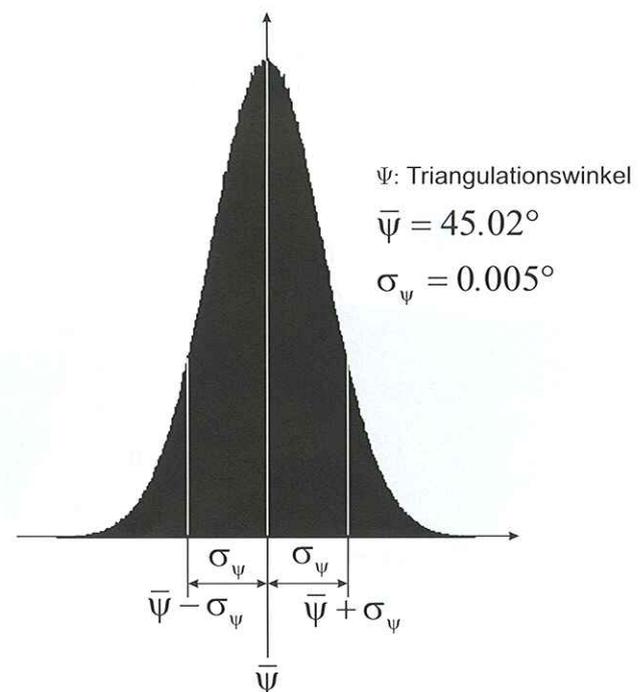


Bild 8 Verteilungsdichtefunktion des Triangulationswinkels.

Dreieck die Messabweichung nach folgender Formel berechnet:

$$s_I = \sqrt{\frac{1}{N-1} \cdot \sum_{k=1}^N (d_k - \bar{d})^2}$$

mit s_I als Messabweichung des Polygons I und N als Anzahl der Abstände des Polygons I .

Ein wesentlicher Aspekt für die Güte der Angabe ist die Anzahl der Polygone, wie vorab beschrieben (siehe Bild 4), sodass ein möglichst feines Gitternetz verwendet werden sollte.

4 Darstellung der Simulationsergebnisse

Nach der Beschreibung der Messung werden die erarbeiteten Algorithmen getestet und mit realen Messdaten verglichen, um die virtuelle Messung zu verifizieren.

4.1 Messobjekt

Als Messobjekt wird eine Kugel eingesetzt. Die reale Kugel wurde mit einem Koordinatenmessgerät gemessen, um die Formabweichung (Rundheit) und den Radius R zu bestimmen:

- Formabweichung $< 26,9 \mu\text{m}$
- Radius: $R = 14,988200 \text{ mm}$

Die virtuelle Kugel wird durch die Anzahl N an Polygonen und durch den mit dem Koordinatenmessgerät ermittelten Radius R der realen Kugel aufgebaut:

$N = 35.000$.

Die Messunsicherheit hängt bei Streifenprojektionssystemen von der Lage des Körpers im Messvolumen ab. Der bei der realen Messung ermittelte Mittelpunkt der realen Kugel wird für die virtuelle Messung genutzt, um gleiche Messbedingungen zu schaffen und eine Vergleichbarkeit zu gewährleisten.

Nach der Simulation bzw. nach der Messung liegt jeweils eine diskrete Punktwolke vor, die auf das Modell einer Kugel eingepasst wird. Die Minimierung der folgenden Zielfunktion führt dabei auf die gesuchten Parameter:

$$\varepsilon = \sum_i \sum_j \left((x_{i,j,\text{Mess}} - x_M)^2 + (y_{i,j,\text{Mess}} - y_M)^2 + (z_{i,j,\text{Mess}} - z_M)^2 - R^2 \right), \quad \min(\varepsilon) \Rightarrow x_M, y_M, z_M, R,$$

wobei N die Anzahl der Punkte, R den Radius und $\{x_M, y_M, z_M\}^T$ den Mittelpunkt der Kugel darstellt.

Zur Lösung des Minimierungsproblems wurden der Simplex-Algorithmus nach Nelder-Mead und das Verfahren nach Gauß-Newton verwendet.

Es hat sich als zweckmäßig erwiesen, zunächst mit dem Simplex-Algorithmus Näherungswerte auszurechnen und anschließend den Fehler durch das Gauß-Newton-Verfahren weiter zu minimieren. Für jedes numerische

Optimierungsproblem müssen Anfangswerte gewählt werden, ab denen die Suche nach dem Minimum beginnt. Als Startwert für das Gauß-Newton-Verfahren wird die Lösung durch den Simplex-Algorithmus gewählt. Die Startwerte für den Simplex-Algorithmus werden über den Mittelwert der Messpunkte in x -, y - bzw. z -Richtung berechnet. Als Radius wird der maximale Abstand des Mittelpunktes zu den Messpunkten gesetzt.

4.2 Ermittlung der Unsicherheit mit dem realen System

Für die Abschätzung der realen Messunsicherheit wurden Wiederholungsmessungen an einer Kugel durchgeführt. Der abgeschätzte Mittelpunkt $\{x_M, y_M, z_M\}^T$ und der Radius R sind in der Tabelle 1 aufgeführt.

Mit den ermittelten Parametern kann der Mittelwert des Mittelpunktes $\{\bar{x}_{M,Real}, \bar{y}_{M,Real}, \bar{z}_{M,Real}\}$ und des Radius \bar{R}_{Real} und die zugehörigen Standardabweichungen $s_{x_{M,Real}}, s_{y_{M,Real}}, s_{z_{M,Real}}$ und $s_{R,Real}$ abgeschätzt werden zu:

- $\bar{x}_{M,Real} = 1,309542 \text{ mm}$
- $\bar{y}_{M,Real} = 4,408636 \text{ mm}$
- $\bar{z}_{M,Real} = -16,733054 \text{ mm}$
- $\bar{R}_{Real} = 14,987896 \text{ mm}$
- $s_{x_{M,Real}} = 0,001813 \text{ mm}$
- $s_{y_{M,Real}} = 0,002829 \text{ mm}$
- $s_{z_{M,Real}} = 0,005604 \text{ mm}$
- $s_{R,Real} = 0,004594 \text{ mm}$

Tabelle 1 Parameter der Kugel aus der Einpassung der Messdaten.

Messung	x_M in mm	y_M in mm	z_M in mm	R in mm
1	1,312	4,408	16,739	14,993
2	1,308	4,411	16,738	14,990
3	1,308	4,408	16,731	14,985
4	1,311	4,406	16,733	14,989
5	1,310	4,405	16,723	14,980
6	1,308	4,412	16,732	14,986
7	1,308	4,407	16,740	14,995
8	1,307	4,411	16,732	14,986
9	1,311	4,404	16,725	14,983
10	1,307	4,411	16,731	14,986

4.3 Ermittlung der Unsicherheit mit dem virtuellen System

Für die Monte-Carlo-Simulation werden die Parameter in Form von Zufallszahlen beschrieben. Die dabei zugrunde liegenden Verteilungsdichtefunktionen werden alle als gaußverteilt angenommen. Als Beispiel wird in diesem Artikel der Triangulationswinkel, wie in Bild 8 gezeigt, variiert. Um die Standardabweichung angeben zu können,

wurde der Kalibrierungsprozess mehrfach wiederholt und auf Grundlage der Wiederholungen die Unsicherheit der Parameter bestimmt.

Um mit Monte-Carlo-Methoden einen deterministischen Prozess möglichst gut nachbilden zu können, ist es notwendig, eine große Zahl an Simulationen durchzuführen. Bild 9 zeigt die Abhängigkeit der Standardabweichung des Radius von der Anzahl der Simulationen und die Abhängigkeit des Kugelradius von der Anzahl der Polygone. Aus dem Bild geht hervor, dass sich der Wert der Standardabweichung nach ca.

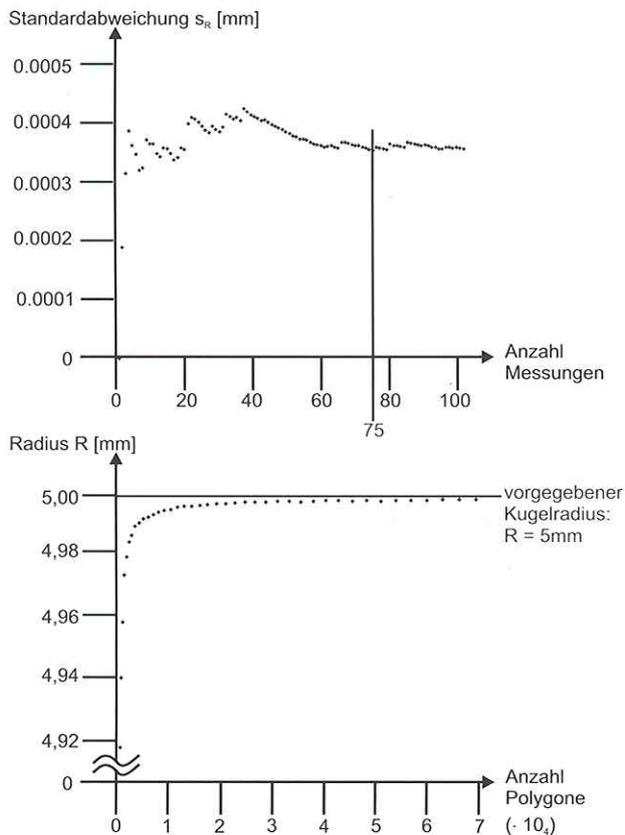


Bild 9 Abhängigkeit der Standardabweichung des Radius von der Anzahl der virtuellen Messungen und Abhängigkeit zwischen dem Radius und der Auflösung der virtuellen Messobjekte.

Tabelle 2 Parameter der Kugel aus der Einpassung der simulierten Messdaten.

Messung	x_M in mm	y_M in mm	z_M in mm	R in mm
1	1,312	4,405	-16,744	14,986
23	1,312	4,405	-16,744	14,986
26	1,311	4,400	-16,740	14,985
41	1,312	4,409	-16,746	14,986
59	1,312	4,411	-16,740	14,986
60	1,312	4,406	-16,753	14,986
75	1,311	4,401	-16,742	14,985
83	1,312	4,400	-16,757	14,985
95	1,312	4,410	-16,752	14,986
100	1,311	4,399	-16,751	14,985

75 virtuellen Messungen einem Grenzwert nähert. Weiterhin wird veranschaulicht, dass sich der Kugelradius bei immer feinerer Auflösung des virtuellen Messobjekts immer weiter dem realen Radius annähert. Um die Güte der abgeschätzten Standardabweichung zu gewährleisten, wurden in diesem Fall 100 Einzelsimulationen durchgeführt. Die Ergebnisse einiger Simulationen bzw. der virtuellen Messungen sind in Tabelle 2 aufgeführt:

Aus den virtuellen Messungen ergeben sich der Mittelwert und die Standardabweichung des Mittelpunktes und des Radius zu:

$$\begin{aligned} \bar{x}_{M,Sim} &= 1,312167 \text{ mm} \\ \bar{y}_{M,Sim} &= 4,406222 \text{ mm} \\ \bar{z}_{M,Sim} &= -16,747873 \text{ mm} \\ \bar{R}_{Sim} &= 14,986109 \text{ mm} \\ s_{X_{M,Sim}} &= 0,000242 \text{ mm} \\ s_{Y_{M,Sim}} &= 0,004665 \text{ mm} \\ s_{Z_{M,Sim}} &= 0,006800 \text{ mm} \\ s_{R,Sim} &= 0,000674 \text{ mm} \end{aligned}$$

Der mittels der Simulation ermittelte Kugelradius ist kleiner als der reale Kugelradius. Dieser Effekt kann mit der Polygonisierung der Kugel durch die Dreiecke erklärt werden. Die Größe des Radius hängt dabei mit der Anzahl der Polygone zusammen und wird mit steigender Anzahl größer (siehe Bild 9).

4.4 Vergleich der Ergebnisse

Die mit dem realen System ermittelten Unsicherheiten liegen im selben Größenordnungsbereich wie die Unsicherheiten, die aus den simulierten Messdaten errechnet wurden. Der Grund für das Ausbleiben von Abweichungen zwischen den Ergebnissen kann mit Modellgenauigkeiten erklärt werden. Beispielsweise sind die optischen Achsen des Projektors und der Kamera beim realen System zusätzlich um die x - bzw. z -Achse verkippt. Dies führt dazu, dass die Achsen windschief zueinander sind und das Objekt nicht mehr durch einfache Triangulation, wie in diesem Fall, rekonstruiert werden kann. Des Weiteren können Modellgenauigkeiten durch eine fehlerhafte Kalibrierung und somit durch fehlerhafte Modellparameter hervorgerufen werden.

Für eine weitere Genauigkeitssteigerung könnte ein weiterer Abgleich zwischen realem und virtuellem System durchgeführt werden. Hierzu bietet es sich an, einen Referenzkörper, beispielsweise eine Kalibrierebene oder eine Kalibrierkugel, an verschiedenen Positionen im Messvolumen real und virtuell zu messen. Durch nachfolgende Einpassung der virtuellen Messpunkte $\mathbf{X}_{Sim} = [x_{Sim}, y_{Sim}, z_{Sim}]^T$ auf die realen Messpunkte $\mathbf{X}_{Real} = [x_{Real}, y_{Real}, z_{Real}]^T$ können zum einen korrespondierende Punktepaare und zum anderen Ab-

weichungen $\delta\mathbf{X} = [\delta x, \delta y, \delta z]^T$ zwischen realer und virtueller Messung ermittelt und durch ein Blackboxsystem beschrieben werden, wobei folgender Zusammenhang vorausgesetzt wird:

$$\mathbf{X}_{Real} = \mathbf{X}_{Sim} + \delta\mathbf{X},$$

$$x_{Real} = x_{Sim} + \delta x \quad \text{mit} \quad \delta x = f_1(x_{Sim}, y_{Sim}, z_{Sim}),$$

$$y_{Real} = y_{Sim} + \delta y \quad \text{mit} \quad \delta y = f_2(x_{Sim}, y_{Sim}, z_{Sim}),$$

$$z_{Real} = z_{Sim} + \delta z \quad \text{mit} \quad \delta z = f_3(x_{Sim}, y_{Sim}, z_{Sim}).$$

Dies hat das Ziel, alle bisher nicht erfassten physikalischen und nicht-physikalischen Effekte quantitativ zu berücksichtigen. Im neuen Messablauf erfolgt zunächst die Ermittlung der Messpunkte mit dem in Abschnitt 3.2 beschriebenen Raytracing-Verfahren. Die simulierten Messpunkte werden anschließend durch die Verzeichnungsfunktion $\delta\mathbf{X}$ verschoben, wodurch eine bessere Anpassung der Simulation an das reale Sensorverhalten erzielt werden kann.

4.5 Beispiele für eine flächenhafte Messunsicherheitsanalyse

In diesem Abschnitt werden die Simulationsergebnisse der Messunsicherheitsanalyse flächenhafter Messdaten am Beispiel einer Kugel gezeigt. Vergleichsgrundlage stellen die für die Dreiecke des Polygonmodells ermittelten Messabweichungen dar. Um die reale Messabweichung zu erhalten, wurden die Messdaten auf das Polygonmodell einer Kugel eingepasst und aus den Abständen für jedes Polygon die Messabweichung berechnet. Die Differenz der ermittelten realen und abgeschätzten virtuellen

Messabweichung der Polygone ist in Bild 10 dargestellt. Der obere Teil der Abbildung zeigt die farbkodierte Messabweichung für jedes Dreieck. Im unteren Teil ist die Messabweichung als 3-dimensionales Bild zu sehen. Hierfür wurden die Schwerpunkte der Dreiecke auf die x-y-Ebene projiziert und jeder x-y-Koordinate die entsprechende Messabweichung zugeordnet.

Aus dem Bild geht hervor, dass die realen Abweichungen gut mit den simulierten Abweichungen übereinstimmen. Die verbleibenden Fehler können hier, wie in Abschnitt 4.4, über Modellfehler und Ungenauigkeiten erklärt werden.

5 Zusammenfassung und Ausblick

In diesem Artikel wurde eine Methode vorgestellt, mit der vorab Messabweichungen, die bei der Messung von Körpern auftreten, mittels virtueller Streifenprojektionstechnik abgeschätzt werden können. Hierzu erfolgte zunächst eine Beschreibung des verwendeten physikalischen Modells des Streifenprojektionssystems, welches als numerische Computersimulation umgesetzt worden ist. Aufbauend darauf erfolgte die Beschreibung des virtuellen Messablaufs, wobei insbesondere auf die abstrakte Formulierung der Körper und die Strahlenverfolgung eingegangen wurde.

Auf Basis des virtuellen Messsystems wurde das Prinzip zur Abschätzung der flächenhaften Messabweichung vorgestellt. Zur Verifikation der Simulation wurden die mit dem realen und dem virtuellen System ermittelte Standardabweichung des Radius einer Kugel und die flächenhafte Messabweichung miteinander verglichen. Die Ergebnisse stimmen gut überein.

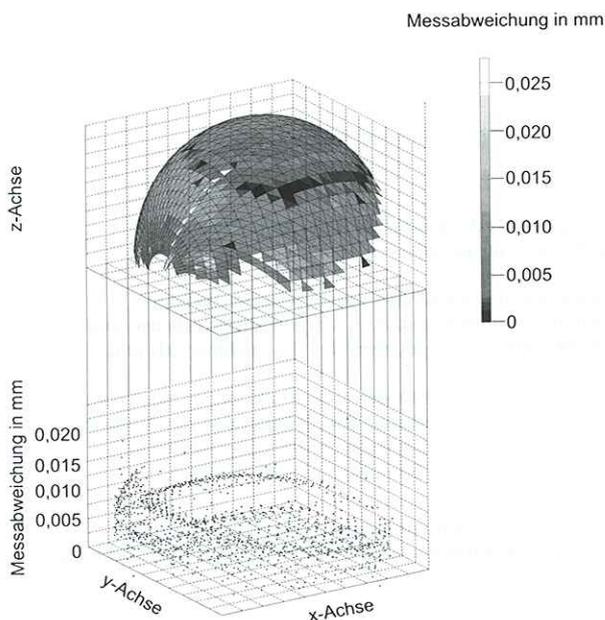
In Verbindung mit dem virtuellen Messablauf stellt das virtuelle Messsystem ein Assistenzsystem zur Messbarkeitsprüfung dar. Das System kann parallel zur Konstruktionsphase bereits eingesetzt werden, um vorab zu prüfen, ob festgelegte Bauteiltoleranzen ausreichend genau erfasst werden können, und ob die zu prüfenden Geometriebereiche überhaupt erfassbar sind.

Danksagung

Die Autoren möchten der Deutschen Forschungsgemeinschaft (DFG) für die Finanzierung des Teilprojekts B5 „Vollständige fertigungsnahe Geometrieprüfung“ des Sonderforschungsbereichs 489 „Prozesskette zur Herstellung präzisionsgeschmiedeter Hochleistungsbauteile“ danken.

Literatur

- [1] R.I. Hartley, A. Zissermann: *Multiple View Geometry in Computer Vision*. Cambridge University Press, 2. Auflage, 2004.
- [2] O. Schreer: *Stereosynthese und Bildanalyse*, Springer Verlag Berlin, Heidelberg, 2005.
- [3] T. Böttner: *Messunsicherheitsbetrachtungen an einem virtuellen Streifenprojektionssystem*. Dissertation, Leibniz Universität Hannover 2009.



oben: farbkodierte Messabweichung
unten: Messabweichung der auf die x-y-Ebene projizierten Dreiecke

Bild 10 Differenz zwischen der realen und virtuellen Messabweichung.

- [4] R. J. Valkenburg, A. M. McIvor: Accurate 3D measurement using a structured light system. In: *Image and Vision Computing* 16, Ausgabe 2, S. 99–110, 1998.
- [5] T. Luhmann: *Nahbereichsphotogrammetrie*. Wichmann Verlag, 2000.
- [6] D. C. Brown: Close-range camera calibration. In: *Photogrammetrie Engineering*, S. 855–866, 1971.
- [7] F. Remondino, C. S. Fraser: Digital Camera Calibration Methods: Considerations & Comparisons. ISPRS Commission V Symposium Image Engineering and Vision Metrology, 2006.

Manuskripteingang: 4. Januar 2010, zur Veröffentlichung angenommen: 4. Dezember 2010

Dipl.-Ing. Klaus Haskamp (Korrespondenzautor) erhielt 2008 sein Diplom an der Leibniz Universität Hannover. Seit 2008 ist er wissenschaftlicher Mitarbeiter in der Arbeitsgruppe Fertigungsmess- und Prüftechnik am Institut für Mess- und Regelungstechnik der Leibniz Universität Hannover. Aktuell befasst er sich im Rahmen des Sonderforschungsbereichs 489 mit der Entwicklung eines virtuellen Multisensorsystems zur Abschätzung von Messunsicherheiten. Hauptarbeitsgebiete: Fertigungsmesstechnik, optische Messtechnik, Streifenprojektionstechnik

Adresse: Institut für Mess- und Regelungstechnik, Nienburger Straße 17, 30167 Hannover, E-Mail: klaus.haskamp@imr.uni-hannover.de

Dr.-Ing. Markus Kästner ist Leiter der Arbeitsgruppe Fertigungsmess- und Prüftechnik des Instituts für Mess- und Regelungstechnik der Leibniz Universität Hannover. Hauptarbeitsgebiete: Fertigungsmesstechnik, Koordinatenmesstechnik, optische Messtechnik

Adresse: Institut für Mess- und Regelungstechnik, Nienburger Straße 17, 30167 Hannover, E-Mail: markus.kaestner@imr.uni-hannover.de

Dipl.-Wirtsch.-Ing. Rüdiger Gillhaus arbeitet seit 2006 als wissenschaftlicher Mitarbeiter in der Arbeitsgruppe Fertigungsmess- und Prüftechnik des Instituts für Mess- und Regelungstechnik der Leibniz Universität Hannover. Hauptarbeitsgebiete: Fertigungsmesstechnik, optische Messtechnik

Adresse: Institut für Mess- und Regelungstechnik, Nienburger Straße 17, 30167 Hannover, E-Mail: ruediger.gillhaus@imr.uni-hannover.de

Prof. Dr.-Ing. Eduard Reithmeier ist Leiter des Instituts für Mess- und Regelungstechnik der Leibniz Universität Hannover. Hauptarbeitsgebiete: robuste Regelung, linear adaptive Regelung, bildrückgeführte Regelung, optische Messtechnik

Adresse: Institut für Mess- und Regelungstechnik, Nienburger Straße 17, 30167 Hannover, E-Mail: sekretariat@imr.uni-hannover.de



Nachschlagewerk für Studium, Forschung und Entwicklung



Tilo Pfeifer, Robert Schmitt
Fertigungsmesstechnik
 3., überarbeitete und erweiterte Auflage 2010 | 481 S. | Broschur | € 54,80
 ISBN 978-3-486-59202-3

Das Werk stellt die Systematik der Fertigungsmesstechnik ausgehend von der Prüfplanung über die Prüfdatenerfassung bis hin zur Prüfdatenauswertung vor.

Dem Leser wird damit einerseits das Basiswissen zum Verständnis der vorgestellten Verfahren und zu deren praktischem Einsatz vermittelt. Andererseits wird auch die grundsätzliche Bedeutung der Fertigungsmesstechnik für die Qualitätssicherung in produzierenden Unternehmen erläutert.

Das Buch stellt somit ein unentbehrliches Nachschlagewerk für den in der Praxis tätigen Ingenieur, aber auch ein lesefreundlich aufbereitetes Lehrbuch für Studenten dar. Als Ergänzung zum Studium gibt es einen schnellen Überblick über die notwendigen Grundlagen der Fertigungsmesstechnik und vermittelt gleichzeitig den Bezug dieses theoretischen Grundlagenwissens zur Praxis.

Das Buch richtet sich an Studierende des Maschinenbaus mit Schwerpunkt Messtechnik und Qualitätsmanagement, aber auch an Berufspraktiker in den Bereichen Forschung und Entwicklung.

Bestellen Sie in Ihrer Fachbuchhandlung oder direkt bei uns:
 Tel: 089/45051-248, Fax: 089/45051-333, verkauf@oldenbourg.de
www.oldenbourg-wissenschaftsverlag.de

Oldenbourg