

Design and implementation of a connectivity manager for virtual scalable network environments

Bachelorarbeit
von

Manuel Bergler

01.12.2014 – 08.02.2015

Referent: Prof. Dr. Thomas Baar
Korreferent: Prof. Dr. Thomas Baar
Betreuer: Benjamin Reichel

Manuel Bergler
Urbanstr. 26
10967 Berlin

Hiermit versichere ich, dass ich die von mir vorgelegte Arbeit selbstständig verfasst habe, dass ich die verwendeten Quellen, Internet-Quellen und Hilfsmittel vollständig angegeben habe und dass ich die Stellen der Arbeit – einschließlich Tabellen, Karten und Abbildungen –, die anderen Werken oder dem Internet im Wortlaut oder dem Sinn nach entnommen sind, auf jeden Fall unter Angabe der Quelle als Entlehnung kenntlich gemacht habe.

Berlin, den 08. Februar 2015

(Unterschrift)

Manuel Bergler

Contents

| | |
|--|-----------|
| Abbildungsverzeichnis | 5 |
| Tabellenverzeichnis | 7 |
| Algorithmenverzeichnis | 9 |
| 1 Einführung | 11 |
| 1.1 Motivation, Zielsetzung und Beitrag der Arbeit | 11 |
| 1.2 Aufbau und Kapitelübersicht | 11 |
| 2 Stand der Technik | 21 |
| 2.1 Klassifizierung der Verfahren zur | 21 |
| 2.2 Vergleich und Bewertung der Verfahren | 21 |
| 3 Grundlagen | 23 |
| 3.1 Text | 23 |
| 3.2 Text | 23 |
| 3.2.1 Text | 23 |
| 3.2.2 Text | 23 |
| 3.3 Text | 24 |
| 4 Umsetzung | 25 |
| 4.1 Text | 25 |
| 4.2 Text | 25 |
| 4.3 Text | 25 |
| 5 Systemarchitektur | 27 |
| 5.1 Hardware | 27 |
| 5.2 Software | 27 |
| 5.2.1 Verwendete Bibliotheken | 27 |
| 5.2.2 Klassendiagramm | 27 |
| 5.2.3 Anwenderschnittstelle | 27 |

| | |
|--|-----------|
| 6 Experimentelle Validierung | 29 |
| 6.1 Systemparameter | 29 |
| 6.2 Ergebnisse zu Genauigkeit, Auf'-lösung und Wiederholrate | 29 |
| 7 Schlussbetrachtungen | 31 |
| 7.1 Ergebnisse der Arbeit | 31 |
| 7.2 Diskussion und Ausblick | 31 |
| A Mathematik | 33 |
| A.1 Vektorrechnung | 33 |
| A.1.1 Vektorprodukt | 33 |
| A.1.2 Invertierung einer Matrix | 34 |
| A.1.3 Geraden | 34 |
| A.1.4 Ebenen | 34 |
| A.1.5 Schnitt einer Geraden mit einer Ebene | 35 |
| A.1.6 Rotationen | 35 |
| A.1.7 Homogene Koordinaten | 36 |
| A.2 Numerik | 38 |
| A.2.1 Methode der kleinsten Quadrate | 38 |
| A.2.2 Gauß-Elimination | 39 |
| A.2.3 Cholesky-Verfahren | 40 |
| B Format der Parameterdateien | 41 |
| C Quelltextauszüge | 43 |
| D Datenblätter | 49 |
| E Glossar | 51 |
| Literatur | 57 |
| Weiterführende Literatur zu wissenschaftlichen Ausarbeitungen | 59 |
| Sachverzeichnis | 59 |

List of Figures

| | | |
|-----|---|----|
| 1.1 | Beispiel für eine Vektorgrafik (Fonts: DCR10, ISOCTEURItalic), Bildquelle: [Hoischen 88]. | 16 |
| 1.2 | Beispiel für eine Vektorgrafik (Font: Times New Roman). | 16 |
| 1.3 | Beispiel für eine Gnuplot-Kurve. | 17 |
| 1.4 | Beispiel für einen Messwert-Plot mit Gnuplot. | 17 |
| 1.5 | Beispiel für einen 3D-Plot mit Gnuplot (Ausgabe im Postscript-Format, dann Konvertierung nach pdf mittels Ghostscript). | 18 |
| E.1 | Triangulation (Bildquelle: [Wikipedia 07]). | 55 |

List of Tables

| | | |
|-----|---|----|
| 1.1 | Auflistung häufig benötigter Latex-Sonderzeichen. | 14 |
| 1.2 | Übersicht der HD44780 Befehle, (*) bedeutet: Bit nicht verwendet. | 15 |
| 1.3 | Beispiel für eine MS-Word-Tabelle (kein Unterschied erkennbar). | 15 |

Algorithmenverzeichnis

| | | |
|---|--|----|
| 1 | LöseLGS Gauß(A, \mathbf{b}) $\rightarrow \mathbf{x}$ | 39 |
| 2 | LöseLGS Cholesky(A, \mathbf{b}) $\rightarrow \mathbf{x}$ | 40 |

Chapter 1

Einführung

1.1 Motivation, Zielsetzung und Beitrag der Arbeit

Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquid ex ea commodi consequat. Quis aute iure reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint obcaecat cupiditat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

Duis autem vel eum iriure dolor in hendrerit in vulputate velit esse molestie consequat, vel illum dolore eu feugiat nulla facilisis at vero eros et accumsan et iusto odio dignissim qui blandit praesent luptatum zzril delenit augue duis dolore te feugait nulla facilisi. Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed diam nonummy nibh euismod tincidunt ut laoreet dolore magna aliquam erat volutpat.

1.2 Aufbau und Kapitelübersicht

Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquid ex ea commodi consequat. Quis aute iure reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint obcaecat cupiditat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

Duis autem vel eum iriure dolor in hendrerit in vulputate velit esse molestie consequat, vel illum dolore eu feugiat nulla facilisis at vero eros et accumsan et iusto odio dignissim qui blandit praesent luptatum zzril delenit augue duis dolore te feugait nulla facilisi. Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed diam nonummy nibh euismod tincidunt ut laoreet dolore magna aliquam erat volutpat.

Es folgen die ersten Formatierungen: Aufzählungspunkte, als `\itemize` als `\enumerate` und als `\description`.

Übersicht über Zweitakt-Spülverfahren:

- Querstromspülung mit Nasenkolben
- Umkehrspülung
- Dreistromspülung
- Drehschiebersteuerung
- Membransteuerung

Aufzählung der gängigen Zweitakt-Spülverfahren:

1. Querstromspülung mit Nasenkolben
2. Umkehrspülung
3. Dreistromspülung
4. Drehschiebersteuerung

Die nachfolgenden Punkte sind zusammengehalten über eine minipage, vergleichbar dem „Absatz zusammenhalten“ in WORD, um einen Seitenumbruch zwischen den Punkten zu vermeiden (die eigentlich hierfür vorgesehenen Formatierungen `\samepage` oder `\nopagebreak` funktionieren nicht zuverlässig). Diese Art der Formatierung sollte erst ganz am Ende der Dokumenterstellung verwendet werden. Vgl. hierzu auch den Befehl `\mbox` für einzelne Wörter, die nicht getrennt werden sollen (zu verwenden bei falschen Trennungen).

Vor- und Nachteile der gängigen Zweitakt-Spülverfahren:

Querstromspülung mit Nasenkolben Einfache Bauform, hohes Kolbengewicht, große Spülverluste.

Umkehrspülung Höhere Verdichtung, geringere Spülverluste, geringerer Kraftstoffverbrauch, geringere Neigung zur Selbstzündung.

Dreistromspülung Verbesserte Spülung durch stabilere Strömung. Das Kraftstoff-Luft-Gemisch durchströmt zuerst den Kolben, wobei der Kolbenbolzen zusätzlich geschmiert wird.

Drehschiebersteuerung Verbesserte Kurbelkammerfüllung, geringere Spülverluste, ca. 10 % Kraftstoffeinsparung.

Membransteuerung Verbesserte Füllung und Motorleistung über einen weiteren Drehzahlbereich, weniger Spülverluste, geringerer Kraftstoffverbrauch.

Anmerkung: im obigen Text wurde der Ausdruck „10 %“ mittels `10\,%` formuliert. Es wurde also zwischen 10 und % ein halbes, geschütztes Leerzeichen eingefügt. Diese Art der Formatierung ist auch generell zwischen Zahl und Einheit zu verwenden: 220 km, 5 VDC ...

Es folgen die wichtigsten Textformatierungen:

`\emph{}`: *Lorem ipsum dolor sit amet, consectetur adipiscing elit.*

`\textbf{}`: **Lorem ipsum dolor sit amet, consectetur adipiscing elit.**

`\textsc{}`: LOREM IPSUM DOLOR SIT AMET, CONSECTETUR ADIPISICI ELIT.

`\verb$$$`: Lorem ipsum dolor sit amet, consectetur adipiscing elit.

`\large{}`: Lorem ipsum dolor sit amet, consectetur adipiscing elit.

`\Large{}`: Lorem ipsum dolor sit amet, consectetur adipiscing elit.

`\LARGE{}`: Lorem ipsum dolor sit amet, consectetur adipiscing elit.

`\huge{}`: Lorem ipsum dolor sit amet ...

`\Huge{}`: Lorem ipsum dolor sit amet ...

Mit folgendem Befehl wird wieder auf normale Formatierung umgeschaltet:

`\normalsize{}`

Es folgt ein manueller Seitenumbruch mittels `\newpage`. Auch diese Art der Formatierung ist erst ganz am Ende der Dokumentenerstellung zu verwenden, wenn gesichert ist, dass sich der Text nicht mehr verändert.

Untenstehend sind häufig benötigte Sonderzeichen aufgeführt, der Latex-Quelltext zur Tabelle kann darüber hinaus auch für eigene Tabellen als Vorlage dienen:

| Zeichen | Latex-Quelltext | Anmerkung |
|-------------|------------------------------|--|
| Leerzeichen | <code>~</code> | nach Möglichkeit zu vermeiden. . . |
| - | <code>-</code> | kurzer Bindestrich (O-Beine) |
| — | <code>--</code> | langer Bindestrich, Gedankenstrich (10–18 Uhr) |
| — | <code>---</code> | langer engl. Gedankenstrich |
| „Text“ | <code>\glqq ... \grqq</code> | deutsche Anführungszeichen |
| ,Text‘ | <code>\glq ... \grq</code> | deutsche, einfache Hochkommata |
| ∅ | <code>\o{}</code> | Durchmesser |
| \$ | <code>\\$</code> | Dollarzeichen |
| % | <code>\%</code> | Prozentzeichen |
| & | <code>\&</code> | Ampersand |
| # | <code>\#</code> | Doppelkreuz |
| { | <code>\{</code> | rechts offene geschweifte Klammer |
| } | <code>\}</code> | links offene geschweifte Klammer |
| <u>-</u> | <code>_</code> | Unterstrich |
| § | <code>\S</code> | Paragraph |
| © | <code>\copyright</code> | Copyright |
| £ | <code>\pounds</code> | Englische Pfund |
| ... | <code>\dots</code> | Fortsetzungspunkte |
| ... | <code>\cdots</code> | Fortsetzungspunkte, zentriert |
| × | <code>\times</code> | Multiplikationskreuz |
| \ | <code>\backslash</code> | Backslash, kursiv (Mathemodus) |
| \ | <code>\textbackslash</code> | Backslash |
| ← | <code>\leftarrow</code> | |
| → | <code>\rightarrow</code> | |
| ↑ | <code>\uparrow</code> | |
| ↓ | <code>\downarrow</code> | |
| ⇐ | <code>\Leftrightarrow</code> | |
| ⇒ | <code>\Rightarrow</code> | |
| ⇑ | <code>\Uparrow</code> | |
| ⇓ | <code>\Downarrow</code> | |
| ↵ | <code>\hookrightarrow</code> | Deutet eine Zeilenumbruchsfortsetzung an |
| ± | <code>\pm</code> | |
| ∓ | <code>\mp</code> | |

Table 1.1: Auflistung häufig benötigter Latex-Sonderzeichen.

Es folgt eine etwas aufwändigere Tabelle, um die Leistungsfähigkeit und Flexibilität von Latex zu zeigen. Die Tabelle entstammt dem Buch Embedded Robotics (Elektor-Verlag) und wurde von Alexander Bierbaum zur Verfügung gestellt. Die Schrift in der Tabelle ist in Größe `\footnotesize` formatiert (Alternativen für kleine Schriften: `\tiny`, `\scriptsize`, `\footnotesize`, `\normalsize`).

| RS | R/W | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 | Beschreibung |
|----|-----|-------|----|----|----|-----|-----|-----|----|--|
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | Anzeige löschen |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | * | Cursor Home (Position 1) |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | I/D | S | Cursor Bewegungsrichtung |
| 0 | 0 | 0 | 0 | 0 | 0 | 1 | D | C | B | Display/Cursor (un)sichtbar |
| 0 | 0 | 0 | 0 | 0 | 1 | S/C | R/L | * | * | Cursor/Display bewegen |
| 0 | 0 | 0 | 0 | 1 | DL | N | F | * | * | LCD+Interface Konfiguration |
| 0 | 0 | 0 | 1 | AC | | | | | | Setze CG RAM Adresse (Zeichenmaske) |
| 0 | 0 | 1 | AC | | | | | | | Setze DD RAM Adresse (Cursorposition) |
| 0 | 1 | BF | AC | | | | | | | Status auslesen |
| 1 | 0 | Daten | | | | | | | | Daten an Cursorposition oder in CG RAM schreiben |
| 1 | 1 | Daten | | | | | | | | Daten an Cursorposition oder von CG RAM lesen |

Table 1.2: Übersicht der HD44780 Befehle, (*) bedeutet: Bit nicht verwendet.

Es folgt eine weitere Tabelle, dieses Mal allerdings effizienter in MS WORD erstellt, als .pdf exportiert und in Adobe Acrobat zugeschnitten. Zu weiteren Tricks (Font, Größe) vgl. das zugehörige WORD-Dokument mitsamt zugehörigem Font im Unterverzeichnis `\BilderKap1` (beachte auch die Verwendung der Umgebung `\table`, damit in der Bildunterschrift „Tabelle“, statt „Abbildung“ steht).

| Wertetafel | |
|---------------------|------|
| Geschw.-Zunahme | Zeit |
| 0 km/h bis 60 km/h | 6 s |
| 0 km/h bis 80 km/h | 10 s |
| 0 km/h bis 100 km/h | 14 s |
| 0 km/h bis 120 km/h | 24 s |
| 0 km/h bis 140 km/h | 40 s |
| 0 km/h bis 152 km/h | 65 s |

Table 1.3: Beispiel für eine MS-Word-Tabelle (kein Unterschied erkennbar).

Anzumerken ist noch, dass diese Tabelle felsenfest verankert wurde mit der Option `[H]` unter gleichzeitiger Verwendung des Packages `float`. Diese Maßnahme ist nur bei sehr vielen, ungeplant wandernden Bildern oder ganz am Ende der Dokumentenerstellung sinnvoll, da sie sonst zu sehr in den Satz von Latex eingreift und unschöne Lücken entstehen lässt.

Die nachfolgenden exemplarischen Vektorgrafiken wurden ebenfalls mit MS Word bzw. (fast gleichbedeutend) mit MS Visio erstellt und über pdf-Export und Zuschnitt mittels Adobe Acrobat (Beschneidungswerkzeug) im Latex-Dokument eingebettet.

Diese Art der Erstellung von Zeichnungsobjekten eröffnet auch die Möglichkeit, Zeichnungen, Charts, CADs, UML-Diagramme o.ä. mit jedwem Programm zu erstellen, welches eine Druckoption bietet. Der Export geschieht dann über den Druckertreiber „Adobe PDF“ (Vorsicht, dieser wird nur bei Vollinstallation des kommerziellen Adobe Acrobat-Studios installiert).

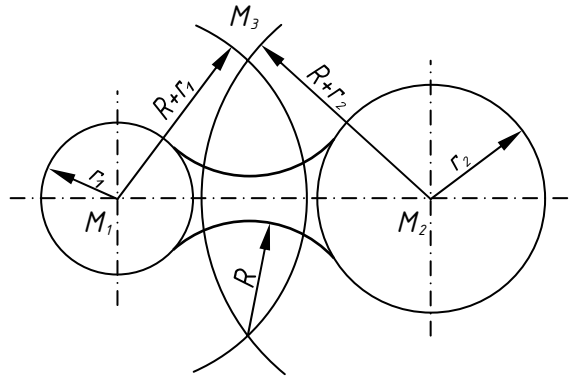


Figure 1.1: Beispiel für eine Vektorgrafik (Fonts: DCR10, ISOCTEURItalic), Bildquelle: [Hoischen 88].

Im zugehörigen Word-Dokument finden sich noch weitere Zeichnungen und Erklärungen, wie diese erstellt wurden. Weiterhin sind dort auch die zugehörigen Fonts abgelegt, welche für die Beispiele in das Verzeichnis `\windows\fonts` kopiert werden müssen.

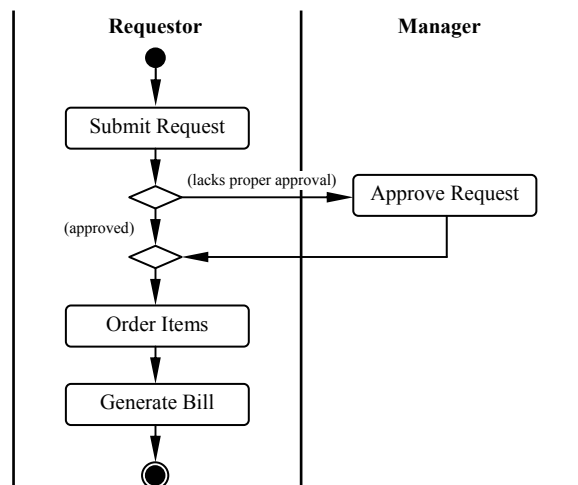


Figure 1.2: Beispiel für eine Vektorgrafik (Font: Times New Roman).

Beim Umgang mit .pdf-Dateien ist stets besonders darauf zu achten, dass die verwendeten Fonts eingebettet werden. Einstellbar ist dies im Adobe Acrobat über Bearbeiten / Grundeinstellungen / ... Eine Kontrolle erfolgt über Datei / Dokumenteigenschaften / Schriften oder über Erweitert / Preflight / Liste mit Text ohne eingebettete Schriften.

Verwenden Sie eine Version ≥ 6.0 ; gerade bei der Schrifteneinbettung gab es bei früheren Versionen öfters Schwierigkeiten.

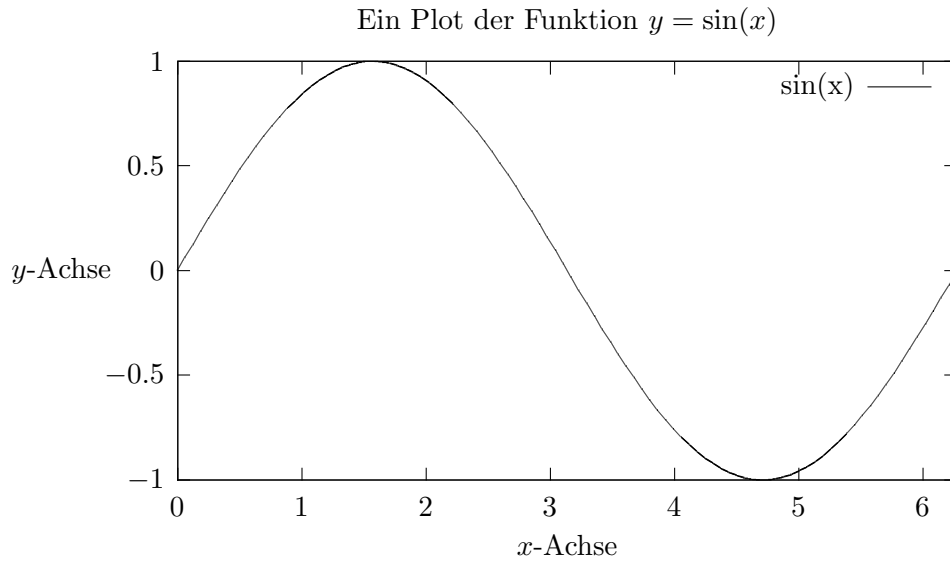


Figure 1.3: Beispiel für eine Gnuplot-Kurve.

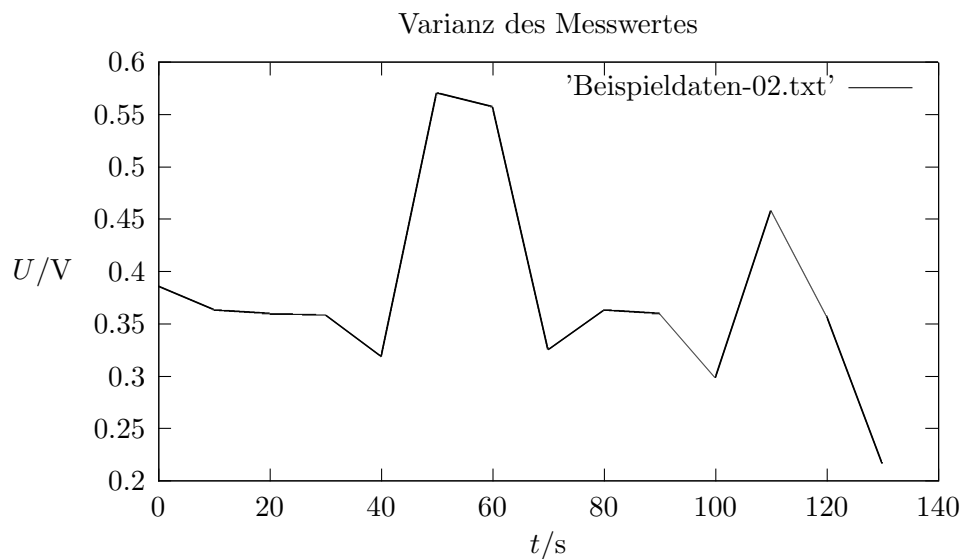


Figure 1.4: Beispiel für einen Messwert-Plot mit Gnuplot.

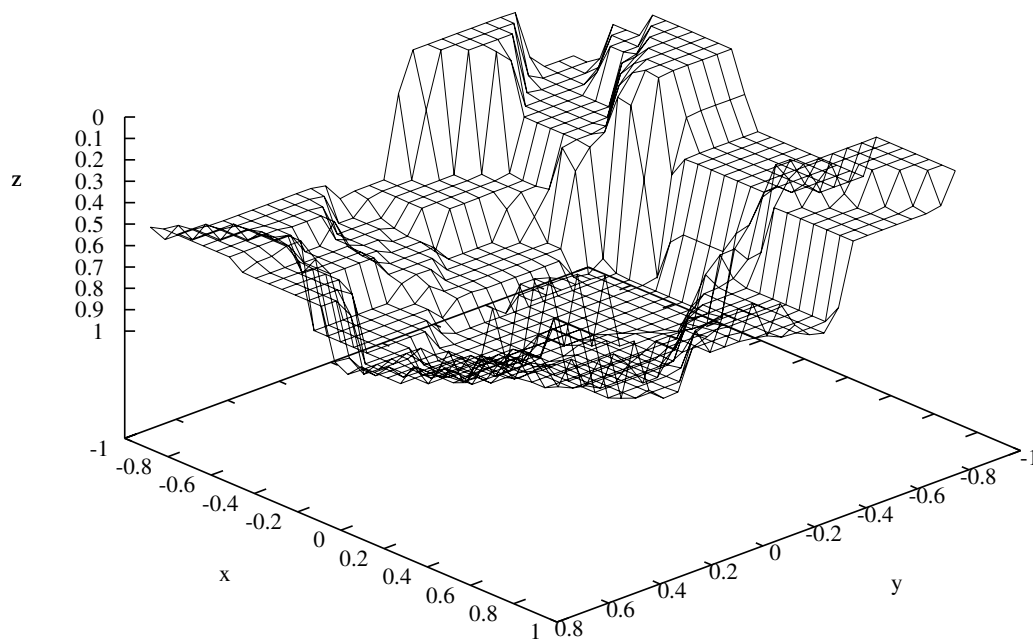


Figure 1.5: Beispiel für einen 3D-Plot mit Gnuplot (Ausgabe im Postscript-Format, dann Konvertierung nach pdf mittels Ghostscript).

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquid ex ea commodi consequat. Quis aute iure reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint obcaecat cupiditat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

Einschub: Kommentare im Latex-Quelltext sind auf zwei Arten möglich:

1.) Umwandlung einer Zeile bzw. eines zusammenhängenden Absatzes zum Kommentar:
Durch ein vorangestelltes %.

2.) Umwandlung eines längeren Blocks zum Kommentar: Mittels Klammerung:

```
\usepackage{verbatim}
\begin{comment}
    Dies ist ein Kommentar
\end{comment}
```

Einschub: der Befehl `\verb$...$` ermöglicht die Eingabe eines Textes zwischen den Dollarzeichen, der nicht von Latex interpretiert wird. Somit stellt der Befehl die einfachste Art dar, Text mit vielen Sonderzeichen einzugeben. Typischerweise sind dies Quelltextzeilen, Dateinamen, Verzeichnisnamen, Programmaufrufe mit Parametern und URLs. Ausgegeben wird der Text in der Schriftart Courier New.

Beispiel, ein Compiler-Aufruf:

```
sdcc -I c:\sdcc\include -L c:\sdcc\lib\large simpletest.c --model-large
```

Beispiel, mit angedeuteten Leerzeichen, einstellbar via `\verb*$...$`:

```
sdcc_I_c:\sdcc\include_L_c:\sdcc\lib\large_simpletest.c--model-large
```

Zu Formelsatz und Algorithmen: vgl. Anhang A: Mathematik.

Chapter 2

Stand der Technik

2.1 Klassifizierung der Verfahren zur ...

Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquid ex ea commodi consequat. Quis aute iure reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint obcaecat cupiditat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

2.2 Vergleich und Bewertung der Verfahren

Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquid ex ea commodi consequat. Quis aute iure reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint obcaecat cupiditat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

Duis autem vel eum iriure dolor in hendrerit in vulputate velit esse molestie consequat, vel illum dolore eu feugiat nulla facilisis at vero eros et accumsan et iusto odio dignissim qui blandit praesent luptatum zzril delenit augue dui dolore te feugait nulla facilisi. Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed diam nonummy nibh euismod tincidunt ut laoreet dolore magna aliquam erat volutpat.

Chapter 3

Grundlagen

3.1 Text

Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquid ex ea commodo consequat. Quis aute iure reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint obcaecat cupiditat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

3.2 Text

Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquid ex ea commodo consequat. Quis aute iure reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint obcaecat cupiditat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

3.2.1 Text

Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquid ex ea commodo consequat. Quis aute iure reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint obcaecat cupiditat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

3.2.2 Text

Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquid ex ea commodo consequat. Quis aute iure reprehenderit in voluptate velit esse cillum dolore eu Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed eiusmod tempor

incidunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquid ex ea commodi consequatur. Quis aute iure reprehenderit in voluptate velit esse cillum dolore eu

3.3 Text

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed eiusmod.nostrud exercitation ullamco laboris nisi ut aliquid ex ea commodi consequatur. Quis aute iure reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint obcaecat cupiditat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

Chapter 4

Umsetzung

4.1 Text

Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquid ex ea commodo consequat. Quis aute iure reprehenderit in voluptate velit esse cillum dolore eu.

4.2 Text

Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquid ex ea commodo consequat. Quis aute iure reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint obcaecat cupiditat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum. Duis autem vel.

4.3 Text

Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquid ex ea commodo consequat. Quis aute iure reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint obcaecat cupiditat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquid ex ea commodo consequat. Quis aute iure reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint obcaecat cupiditat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

Chapter 5

Systemarchitektur

5.1 Hardware

Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco.

5.2 Software

Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco.

5.2.1 Verwendete Bibliotheken

Duis autem vel eum iriure dolor in hendrerit in vulputate velit esse molestie consequat, vel illum dolore eu feugiat nulla facilisis at vero eros et accumsan et iusto odio dignissim qui.

5.2.2 Klassendiagramm

Nam liber tempor cum soluta nobis eleifend option congue nihil imperdiet doming id quod mazim placerat facer possim assum. Lorem ipsum dolor sit amet, consectetur adipiscing elit.

5.2.3 Anwenderschnittstelle

Consetetur sadipscing elitr, sed diam nonumy eirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diam voluptua. At vero eos et accusam et justo duo dolores et ea rebum.

Chapter 6

Experimentelle Validierung

6.1 Systemparameter

Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquid ex ea commodi consequat. Quis aute iure reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint obcaecat cupiditat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

Duis autem vel eum iriure dolor in hendrerit in vulputate velit esse molestie consequat, vel illum dolore eu feugiat nulla facilisis at vero eros et accumsan et iusto odio dignissim qui blandit praesent luptatum zzril delenit augue dui dolore te feugait nulla facilisi. Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed diam nonummy nibh euismod tincidunt ut laoreet dolore magna aliquam erat volutpat.

6.2 Ergebnisse zu Genauigkeit, Auf"-lösung und Wiederholrate

Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquid ex ea commodi consequat. Quis aute iure reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint obcaecat cupiditat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

Duis autem vel eum iriure dolor in hendrerit in vulputate velit esse molestie consequat, vel illum dolore eu feugiat nulla facilisis at vero eros et accumsan et iusto odio dignissim qui blandit praesent luptatum zzril delenit augue dui dolore te feugait nulla facilisi. Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed diam nonummy nibh euismod tincidunt ut laoreet dolore magna aliquam erat volutpat.

Chapter 7

Schlussbetrachtungen

7.1 Ergebnisse der Arbeit

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquid ex ea commodo consequat. Quis aute iure reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint obcaecat cupiditat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

Duis autem vel eum iriure dolor in hendrerit in vulputate velit esse molestie consequat, vel illum dolore eu feugiat nulla facilisis at vero eros et accumsan et iusto odio dignissim qui blandit praesent luptatum zzril delenit augue duis dolore te feugait nulla facilisi. Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed diam nonummy nibh euismod tincidunt ut laoreet dolore magna aliquam erat volutpat.

7.2 Diskussion und Ausblick

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquid ex ea commodo consequat. Quis aute iure reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint obcaecat cupiditat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

Duis autem vel eum iriure dolor in hendrerit in vulputate velit esse molestie consequat, vel illum dolore eu feugiat nulla facilisis at vero eros et accumsan et iusto odio dignissim qui blandit praesent luptatum zzril delenit augue duis dolore te feugait nulla facilisi. Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed diam nonummy nibh euismod tincidunt ut laoreet dolore magna aliquam erat volutpat.

Appendix A

Mathematik

Anmerkungen: Der nachfolgende Text ist der Quelle [Azad 07] entnommen und soll exemplarisch die Verwendung des Formelsatzes von Latex aufzeigen. Nicht alle Gleichungen sind nummeriert, man beachte hier die Unterscheidung zwischen \dots bzw. \dots und $\begin{equation}$, $\end{equation}$ bzw. auch die Verwendung von \nonumber .

Bei komplexen Formeln hat sich die Verwendung des freien, schlanken Formeleditors TEXAIDE bewährt. In diesem Editor kann eine Gleichung rasch mit der Maus zusammengeklickt und dann über die Zwischenablage in den Latex-Editor übernommen werden. Das Tool ist erhältlich unter der URL: <http://www.dessci.com/en/products/texaide>.

Unter Einbindung des Paketes `amstext` kann in Gleichungen normalformatierter (nicht kursiver) Text eingefügt werden mittels: $\text{\texttt{\textit{a}}}$.

A.1 Vektorrechnung

A.1.1 Vektorprodukt

Seien $\mathbf{a}, \mathbf{b} \in \mathbb{R}^3$ und linear unabhängig, so ist das *Vektorprodukt* (oder auch *Kreuzprodukt*) $\mathbf{a} \times \mathbf{b}$ definiert als der Vektor mit den folgenden Eigenschaften:

- $\mathbf{a} \times \mathbf{b}$ steht senkrecht auf \mathbf{a} und \mathbf{b}
- $\mathbf{a}, \mathbf{b}, \mathbf{a} \times \mathbf{b}$ bilden in dieser Folge ein Rechtssystem
- $|\mathbf{a} \times \mathbf{b}| = |\mathbf{a}| |\mathbf{b}| \sin \omega(\mathbf{a}, \mathbf{b})$

Es kann wie folgt berechnet werden:

$$\mathbf{a} \times \mathbf{b} = \begin{pmatrix} a_2 b_3 - a_3 b_2 \\ a_3 b_1 - a_1 b_3 \\ a_1 b_2 - a_2 b_1 \end{pmatrix} \quad (\text{A.1})$$

A.1.2 Invertierung einer Matrix

Die Inverse einer 3×3 -Matrix muss nicht mit einem Eliminationsverfahren berechnet werden, sondern kann direkt aufgestellt werden. Gegeben sei die reguläre Matrix:

$$A = \begin{pmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{pmatrix} \quad (\text{A.2})$$

Die inverse Matrix A^{-1} berechnet sich dann zu:

$$\det A := \frac{1}{-a_{13}a_{22}a_{31} + a_{12}a_{23}a_{31} + a_{13}a_{21}a_{32} - a_{11}a_{23}a_{32} - a_{12}a_{21}a_{33} + a_{11}a_{22}a_{33}}$$
$$A^{-1} := \det A \cdot \begin{pmatrix} -a_{23}a_{32} + a_{22}a_{33} & a_{13}a_{32} - a_{12}a_{33} & -a_{13}a_{22} + a_{12}a_{23} \\ a_{23}a_{31} - a_{21}a_{33} & -a_{13}a_{31} + a_{11}a_{33} & a_{13}a_{21} - a_{11}a_{23} \\ -a_{22}a_{31} + a_{21}a_{32} & a_{12}a_{31} - a_{11}a_{32} & -a_{12}a_{21} + a_{11}a_{22} \end{pmatrix} \quad (\text{A.3})$$

A.1.3 Geraden

Eine Gerade g wird im \mathbb{R}^3 durch folgende Gleichung beschrieben:

$$g : \mathbf{x} = \mathbf{a} + r \cdot \mathbf{u}$$

mit $r \in \mathbb{R}$ und $\mathbf{x}, \mathbf{a} \in \mathbb{R}^3$ und $\mathbf{u} \in \mathbb{R}^3 \setminus \{\mathbf{0}\}$. Die Gerade wird eindeutig durch den Aufpunktvektor und den Richtungsvektor beschrieben. \mathbf{a} ist der Ortsvektor des Aufpunktes, ein beliebiger Punkt der Geraden. Die Richtung der Geraden wird durch den Richtungsvektor \mathbf{u} vorgegeben. Für jedes beliebige r bezeichnet \mathbf{x} den Ortsvektor eines Punktes der Geraden.

A.1.4 Ebenen

Es werden drei verschiedene Darstellungsformen einer Ebene im \mathbb{R}^3 gegeben:

1. Parameterdarstellung

$$E : \mathbf{x} = \mathbf{a} + r \cdot \mathbf{u} + s \cdot \mathbf{v}$$

mit $r, s \in \mathbb{R}$ und $\mathbf{x}, \mathbf{a} \in \mathbb{R}^3$ und $\mathbf{u}, \mathbf{v} \in \mathbb{R}^3 \setminus \{\mathbf{0}\}$. Die Ebene wird eindeutig durch den Aufpunktvektor und den beiden Richtungsvektoren beschrieben. \mathbf{a} ist der Ortsvektor des Aufpunktes, ein beliebiger Punkt der Ebene. Die Lage der Ebene im Raum wird durch die beiden Richtungsvektoren \mathbf{u}, \mathbf{v} vorgegeben. Für jedes beliebige Paar (r, s) bezeichnet \mathbf{x} den Ortsvektor eines Punktes der Ebene.

2. Normalenform

$$E : [\mathbf{x} - \mathbf{a}] \cdot \mathbf{n} = 0$$

mit $\mathbf{x}, \mathbf{a} \in \mathbb{R}^3$ und $\mathbf{n} \in \mathbb{R}^3 \setminus \{\mathbf{0}\}$. Die Ebene wird eindeutig durch den Aufpunktvektor und den Normalenvektor beschrieben. \mathbf{a} ist der Ortsvektor des Aufpunktes, ein beliebiger Punkt der Ebene. Die Lage der Ebene im Raum wird durch den Normalenvektor \mathbf{n} vorgegeben.

Jeder Punkt der Ebene erfüllt die Gleichung.

3. Koordinatendarstellung

$$E : n_1 \cdot x_1 + n_2 \cdot x_2 + n_3 \cdot x_3 = c$$

mit $n_1, n_2, n_3, x_1, x_2, x_3, c \in \mathbb{R}$, wobei nicht alle n_i gleich Null sind. Man erhält die Koordinatendarstellung durch Ausmultiplizieren der Normalenform: die n_i sind die Komponenten des Normalenvektors, c ist das Skalarprodukt von Aufpunktsvektor und Normalenvektor.

A.1.5 Schnitt einer Geraden mit einer Ebene

Gegeben seien eine Ebene E in Normalenform und eine Gerade g :

$$\begin{aligned} E : & [\mathbf{x} - \mathbf{p}_E] \cdot \mathbf{n} = 0 \\ g : & \mathbf{x} = \mathbf{p}_g + r \cdot \mathbf{u} \end{aligned}$$

Unter der Voraussetzung, dass $\mathbf{u} \cdot \mathbf{n} \neq 0$, d.h. die Gerade g verläuft nicht parallel zur Ebene E , lässt sich der Ortsvektor \mathbf{s} des Schnittpunktes S wie folgt berechnen:

$$\mathbf{s} = \mathbf{p}_g - \mathbf{u} \frac{(\mathbf{p}_g - \mathbf{p}_E) \cdot \mathbf{n}}{\mathbf{u} \cdot \mathbf{n}} \quad (\text{A.4})$$

A.1.6 Rotationen

Eine Rotation kann sowohl im Zweidimensionalen als auch im Dreidimensionalen durch eine Matrixmultiplikation ausgedrückt werden. Gegeben sei ein Vektor \mathbf{x} . Wird er als Richtungsvektor interpretiert, so wird seine Richtung gedreht. Wird er dagegen als Ortsvektor interpretiert, so wird die Drehung des Punktes um den Ursprung des Koordinatensystems berechnet.

Im \mathbb{R}^2 ist die Berechnung einer solchen Rotationsmatrix eindeutig, da nur eine Drehachse existiert. Gegeben sei ein Vektor $\mathbf{x} = (x, y)$ und ein Drehwinkel θ . Die Drehung gegen den Uhrzeigersinn von \mathbf{x} um den Winkel θ berechnet sich zu:

$$\begin{pmatrix} x' \\ y' \end{pmatrix} = \begin{pmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix}$$

Im \mathbb{R}^3 existieren drei Basisrotationen, um die Achsen x , y und z :

$$\begin{aligned} R_x(\theta) &= \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos \theta & -\sin \theta \\ 0 & \sin \theta & \cos \theta \end{pmatrix} \\ R_y(\theta) &= \begin{pmatrix} \cos \theta & 0 & \sin \theta \\ 0 & 1 & 0 \\ -\sin \theta & 0 & \cos \theta \end{pmatrix} \\ R_z(\theta) &= \begin{pmatrix} \cos \theta & -\sin \theta & 0 \\ \sin \theta & \cos \theta & 0 \\ 0 & 0 & 1 \end{pmatrix} \end{aligned}$$

Aus diesen Rotationen kann nach Euler's Theorem mit drei Variablen jede beliebige Rotation im Raum zusammengesetzt werden. Hierzu existieren zwei verschiedene Konventionen für die Interpretation der Reihenfolge der Einzelrotationen. Für *raumfeste* Drechachsen werden die Einzelrotationen von rechts nach links interpretiert, wie anhand des folgenden Beispiels zu sehen ist:

$$R_{XYZ}(\alpha, \beta, \gamma) = R_Z(\gamma) R_Y(\beta) R_X(\alpha)$$

Für *mitgedrehte* Drechachsen werden die Einzelrotationen von links nach rechts interpretiert:

$$R_{Z'Y'X'}(\gamma, \beta, \alpha) = R_Z(\gamma) R_Y(\beta) R_X(\alpha)$$

Für eine detaillierte Erläuterung sei auf auf [Craig 03] verwiesen.

A.1.7 Homogene Koordinaten

Gegeben sei ein Punkt $\mathbf{p} \in \mathbb{R}^n$ mit $\mathbf{p} = (p_1, \dots, p_n)$. Die homogenen Koordinaten dieses Punktes sind $(n + 1)$ -dimensional:

$$\mathbf{x} = (x_1, \dots, x_n, x_{n+1})$$

Für sie muss gelten:

$$p_k = \frac{x_k}{x_{k+1}} \text{ für alle } k \in \{1, \dots, n\}$$

Dabei ist h_{k+1} ein Skalierungsfaktor, der für die Anwendung von Rotationen und Translationen den Wert Eins besitzt. Wird dagegen eine Projektion durchgeführt, so gilt für das Ergebnis im Allgemeinen $h_{k+1} \neq 1$. Ein Vorteil von homogenen Koordinaten ist die Möglichkeit, eine räumliche Transformation bestehend aus einer Rotation und Translation geschlossen in einer quadratischen Matrix ausdrücken zu können. Im \mathbb{R}^2 kann

$$\begin{pmatrix} x' \\ y' \end{pmatrix} = R \cdot \mathbf{p} + \mathbf{t} = \begin{pmatrix} r_{11} & r_{12} \\ r_{21} & r_{22} \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} + \begin{pmatrix} t_1 \\ t_2 \end{pmatrix} \quad (\text{A.5})$$

ausgedrückt werden als:

$$\begin{pmatrix} x' \\ y' \\ 1 \end{pmatrix} = \left(\begin{array}{cc|c} R & \mathbf{t} \\ \hline \mathbf{0} & 1 \end{array} \right) \cdot \mathbf{x} = \left(\begin{array}{cc|c} r_{11} & r_{12} & t_1 \\ r_{21} & r_{22} & t_2 \\ \hline 0 & 0 & 1 \end{array} \right) \begin{pmatrix} x \\ y \\ 1 \end{pmatrix} \quad (\text{A.6})$$

Analog kann im \mathbb{R}^3

$$\begin{pmatrix} x' \\ y' \\ z' \end{pmatrix} = R \cdot \mathbf{p} + \mathbf{t} = \begin{pmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{pmatrix} \begin{pmatrix} x \\ y \\ z \end{pmatrix} + \begin{pmatrix} t_1 \\ t_2 \\ t_3 \end{pmatrix} \quad (\text{A.7})$$

ausgedrückt werden als:

$$\begin{pmatrix} x' \\ y' \\ z' \\ 1 \end{pmatrix} = \left(\begin{array}{c|c} R & \mathbf{t} \\ \hline \mathbf{0} & 1 \end{array} \right) \cdot \mathbf{x} = \left(\begin{array}{ccc|c} r_{11} & r_{12} & r_{13} & t_1 \\ r_{21} & r_{22} & r_{23} & t_2 \\ r_{31} & r_{32} & r_{33} & t_3 \\ \hline 0 & 0 & 0 & 1 \end{array} \right) \begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix} \quad (\text{A.8})$$

Mithilfe von homogenen Koordinaten können Geraden im \mathbb{R}^2 durch einen Vektor $\mathbf{l} \in \mathbb{R}^3$ dargestellt werden. Gegeben sei ein Punkt $\mathbf{p} = (x, y)$ mit homogenen Koordinaten $\mathbf{x} = (x, y, 1)$. Dann definiert $\mathbf{l} = (l_1, l_2, l_3)$ eine Gerade g wie folgt:

$$g : \mathbf{l} \cdot \mathbf{x} = 0$$

Dies ist eine kompakte Darstellung der Koordinatenform einer Geraden im Zweidimensionalen und kann umformuliert werden zu:

$$g : \begin{pmatrix} l_1 \\ l_2 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} + l_3 = 0$$

Daraus wird ersichtlich, dass (l_1, l_2) der Normalenvektor dieser Geraden ist. Sie kann wie folgt in Parameterdarstellung umgeformt werden:

$$g : \mathbf{x} = \mathbf{a} + r \cdot \begin{pmatrix} -l_2 \\ l_1 \end{pmatrix}$$

Dabei kann der Aufpunkt \mathbf{a} berechnet werden zu:

$$\mathbf{a} = \begin{cases} (-\frac{l_3}{l_1}, 0) & \text{falls } l_1 \neq 0 \\ (0, -\frac{l_3}{l_2}) & \text{sonst} \end{cases} \quad (\text{A.9})$$

A.2 Numerik

A.2.1 Methode der kleinsten Quadrate

Gegeben sei ein überbestimmtes LGS der Form $A\mathbf{x} = \mathbf{b}$, mit $A \in \mathbb{R}^{(m,n)}$, $\mathbf{x} \in \mathbb{R}^n$ und $\mathbf{b} \in \mathbb{R}^m$, mit $m > n$. Ein solches LGS ist im Allgemeinen nicht lösbar. Es wird jedoch angenommen, dass A vollen Rang besitzt: $\text{rang}(A) = n = \min\{n, m\}$. Mit der *Methode der kleinsten Quadrate* nach Gauß lässt sich das vorliegende LGS bestmöglich lösen [Huckle 02].

Das Verfahren minimiert den Abstand $A\mathbf{x} - \mathbf{b}$ bezüglich der euklidischen Norm

$$\min_{\mathbf{x}} |A\mathbf{x} - \mathbf{b}|. \quad (\text{A.10})$$

Die Verwendung der euklidischen Norm führt zu einer Minimierungsaufgabe mit einer differenzierbaren Funktion. Um die anfallenden Rechnungen zu vereinfachen, geht man zu der quadrierten Funktion über und definiert

$$\begin{aligned} f(x_1, \dots, x_n) &= |A\mathbf{x} - \mathbf{b}|^2 \\ &= \left| \left(\sum_{j=1}^n a_{kj}x_j - b_k \right)_{k=1}^m \right|^2 \\ &= \sum_{k=1}^m \left(\sum_{j=1}^n a_{kj}x_j - b_k \right)^2. \end{aligned}$$

Diese Summe von quadratischen Termen nimmt ihr Minimum an, wenn alle Ableitungen gleich Null sind

$$\begin{aligned} 0 &= \frac{df}{dx_i^*} = 2 \sum_{k=1}^m \left(\sum_{j=1}^n a_{kj}x_j^* - b_k \right) a_{ki}, \quad i = 1, \dots, n \\ \Leftrightarrow \sum_{k=1}^m a_{ki} \sum_{j=1}^n a_{kj}x_j^* &= \sum_{k=1}^m a_{ki}b_k, \quad i = 1, \dots, n. \end{aligned}$$

Mit der Matrixnotation dieser n Gleichungen erhält man das Gleichungssystem

$$A^T A \mathbf{x}^* = A^T \mathbf{b}, \quad (\text{A.11})$$

das wegen $\text{rang}(A^T A) = \text{rang}(A) = n$ eindeutig lösbar ist. Man bezeichnet $A^T A \mathbf{x}^* = A^T \mathbf{b}$ als die *Normalgleichung* zu A und \mathbf{b} . Der Lösungsvektor \mathbf{x}^* des vorliegenden LGS minimiert den Abstand $|A\mathbf{x} - \mathbf{b}|$. Da $A^T A$ stets eine positiv definite Matrix ist, lässt sich \mathbf{x}^* durch die Berechnung der unteren Dreiecksmatrix von $A^T A$ und Anwendung der Cholesky-Verfahren effizient berechnen.

A.2.2 Gauß-Elimination

Gegeben sei ein LGS $A\mathbf{x} = \mathbf{b}$, mit $A \in \mathbb{R}^{(n,n)}$, $\mathbf{x} \in \mathbb{R}^n$ und $\mathbf{b} \in \mathbb{R}^n$. Weiterhin sei A regulär. Eine Möglichkeit, den Lösungsvektor \mathbf{x} zu bestimmen, besteht in der Anwendung der *Gauß-Elimination*. Im Folgenden wird der Algorithmus mit einer Spalten-Pivotsuche in Pseudocode dargestellt. Sollen A und \mathbf{b} unverändert bleiben, so müssen die Werte kopiert werden. Vgl. auch [Huckle 02].

Algorithmus 1 LöseLGSGauß(A, \mathbf{b}) $\rightarrow \mathbf{x}$

```
for  $i := 0$  to  $n - 1$  do
     $max := 0, p := -1$ 

    for  $j := i$  to  $n - 1$  do
        if  $(|a_{ji}| > max)$  then
             $max := |a_{ji}|, p := j$ 
        end if
    end for

    if  $p = -1$  then
        STOP {Matrix  $A$  ist nicht regulär}
    end if

    if  $p \neq i$  then
        VertauscheZeilen( $A, i, p$ )
         $s := b_i, b_i := b_p, b_p := s$ 
    end if

     $pivot := a_{ii}$ 

    for  $j := i + 1$  to  $n - 1$  do
         $factor := a_{ji}/pivot$ 
         $b_j := b_j - factor \cdot b_i$ 
        for  $k := i + 1$  to  $n - 1$  do
             $a_{jk} := a_{jk} - factor \cdot a_{ik}$ 
        end for
    end for

    for  $i := n - 1$  downto  $0$  do
         $sum := 0$ 
        for  $j := i + 1$  to  $n - 1$  do
             $sum := sum + a_{ij} \cdot x_j$ 
        end for
         $x_i := (b_i - sum)/a_{ii}$ 
    end for
```

A.2.3 Cholesky-Verfahren

Gegeben sei ein LGS $A\mathbf{x} = \mathbf{b}$, mit $A \in \mathbb{R}^{(n,n)}$, $\mathbf{x} \in \mathbb{R}^n$ und $\mathbf{b} \in \mathbb{R}^n$. Weiterhin sei A positiv definit. Dann lässt sich der Lösungsvektor \mathbf{x} mit dem Cholesky-Verfahren mit etwa dem halben Aufwand einer Gauß-Elimination bestimmen. Im Folgenden wird der Algorithmus in Pseudocode dargestellt, wobei $L \in \mathbb{R}^{(n,n)}$. Nach Ablauf des Algorithmus gilt $A = LL^T$. A und \mathbf{b} bleiben unverändert. Vgl. auch [Huckle 02].

Algorithmus 2 LöseLGSCholesky(A, \mathbf{b}) $\rightarrow \mathbf{x}$

```
if  $a_{00} \leq 0$  then
  STOP {Matrix  $A$  ist nicht positiv definit}
end if

 $l_{00} := \sqrt{a_{00}}$ 

for  $i := 1$  to  $n - 1$  do
  for  $j := 0$  to  $i - 1$  do
     $sum := a_{ij}$ 
    for  $k := 0$  to  $j - 1$  do
       $sum := sum - l_{ik} \cdot l_{jk}$ 
    end for
     $l_{ij} := sum / l_{jj}$ 
  end for

   $sum_1 := a_{ii}$ ,  $sum_2 := b_i$ 

  for  $j := 0$  to  $i - 1$  do
     $sum_1 := sum_1 - l_{ij}^2$ ,  $sum_2 := sum_2 - l_{ij} \cdot x_j$ 
  end for

  if  $sum \leq 0$  then
    STOP {Matrix  $A$  ist nicht positiv definitiv}
  end if

   $l_{ii} := \sqrt{sum_1}$ 
   $x_i := sum_2 / l_{ii}$ 
end for

for  $i := n - 1$  downto  $0$  do
   $sum := x_i$ 
  for  $j := i + 1$  to  $n - 1$  do
     $sum := sum - l_{ji} \cdot x_j$ 
  end for
   $x_i := sum / l_{ii}$ 
end for
```

Appendix B

Format der Parameterdateien

Anmerkung: Ein Anhang zur Erklärung der zum System zugehörigen Parameterdateien hat sich als sinnvoll und hilfreich erwiesen, damit nach Abschied des Diplomanden auch uneingeweihte Personen ohne Quelltextsichtung das System zumindest für Demozwecke in Betrieb nehmen können.

Die Parameterdateien enthalten die Information über die Punktmuster, bzw. Plattenstapel, welche im Rahmen der Testfeldkalibrierung für Kamera und Projektor verwendet werden. Es sind dies: Anordnung und Anzahl der Punkte auf dem Testfeldmuster und Plattendicke und -anzahl.

Kamerakalibrierung, Datei `world_camera.txt`

[Anzahl der Punkte in einer Zeile, d.h. in x-Richtung, Zahlenformat: *int*]

[Anzahl der Punkte in einer Spalte, d.h. in y-Richtung, *int*]

[Anzahl der aufzulegenden Ebenen, *int*]

[Relative Position der Ebenen zueinander, in z-Richtung, *double*, *negativ*]

[x_{w1} y_{w1} , *double*, *double*, durch Leerzeichen getrennt]

:

[x_{wn} y_{wn}]

Der vierte Parameter, die relative Position der Ebenen zueinander, entspricht der Dicke einer Glasplatte. Die Anzahl der Punkte n ist gleich dem Wert, den man durch Multiplikation der ersten drei Parameter erhält. Die x- und y-Koordinaten der einzelnen Punkte bezeichnen ihre Lage in der xy-Ebene des Weltkoordinatensystems. Die Einheit ist [mm]. Die Punkte sind zeilenweise sortiert einzugeben, beginnend mit dem ersten Punkt der obersten Zeile.

Beispiel:

```
11
9
5
-10.0
0.0 0.0
```

```
5.0 0.0
10.0 0.0
15.0 0.0
:
0.0 5.0
5.0 5.0
10.0 5.0
15.0 5.0
:
45.0 40.0
50.0 40.0
```

Projektorkalibrierung, Datei `world_projector.txt`

[Anzahl der aufzulegenden Ebenen, *int*]

[Relative Position der Ebenen zueinander, in z-Richtung, *double, negativ*]

Der zweite Parameter, die relative Position der Ebenen zueinander, entspricht der Dicke einer Glasplatte.

Beispiel:

```
5
-10.0
```

Appendix C

Quelltextauszüge

Anmerkungen: Quelltextauszüge zu einer Implementierung sind im Anhang dann sinnvoll, wenn einige, spezielle Implementierungstechniken aufgezeigt werden sollen, die in der Darstellung als Algorithmus oder Pseudocode nicht deutlich werden. Keinesfalls soll der gesamte Quelltext angehängt werden und weiterhin soll auch in einer Vorbemerkung die Auswahl der Quelltextauszüge genau erklärt werden.

Verwendet wird das freie Quelltext-Pretty-Printing-Tool `a2ps.exe` mit folgender Aufrufkonvention (vgl. [a2ps 07, grep 07]):

```
a2ps.exe --pretty-print=cxx -i test.cpp -o test.ps -T3
```

Die Quelltextdateien dürfen hierfür eine Zeilenlänge von 80 nicht überschreiten. Die störenden Kommentare im Header und Footer des entstandenen .ps-Files können mithilfe des freien Tools `grep.exe` automatisiert entfernt werden. Eine Batch-Datei für den gesamten Konvertierungsprozess inklusive Konvertierung in das pdf-Format hat beispielsweise folgenden Inhalt:

```
a2ps --pretty-print=cxx -i %1 -o tmp -T3
grep -v "Gedruckt von" tmp | grep -v ") footer" > %1.ps
del tmp
"c:\programme\adobe\acrobat 7.0\Acrobat\acrobat.exe" %1.ps
```

Bei Adobe Acrobat Prof. 6.0 muss u.U. das Seitenformat korrigiert werden auf DIN-A4 = 210 mm × 297 mm, die Voreinstellung ist falsch (zu klein, ein Bug), zu korrigieren in: Bearbeiten / Grundeinstellungen / in pdf konvertieren / Postscript/EPS / Einstellungen bearbeiten / bearbeiten / Standardpapierformat. Weiterhin muss an der gleichen Stelle das korrekte und vollständige Einbetten der Schriften eingestellt und später auch kontrolliert werden (vgl. Kapitel 1).

Die nachfolgenden Quelltextausschnitte entstammen dem Programmmodul zum Lageausgleich. Der Quelltext ist hier im Anhang exemplarisch aufgenommen, da die Implementierung des Lageausgleiches auf Basis zentraler Momente für einige Leser von besonderem Interesse sein könnte.

Der Ausschnitt umfasst ca. 300 Programmzeilen, die gesamte im Rahmen der vorliegenden Arbeit entstandene Implementierung umfasst ca. 8.000 Programmzeilen.

| Montag Februar 26, 2007 23:01 | main.cpp | Seite 1/6 |
|--|----------|-----------|
| <pre> // ***** // Project: Alignment and Gauging for industrial parts. // Copyright: Tilo Gockel (Author) // Date: February 25th 2007 // Filename: main.cpp // Author: Tilo Gockel, Chair Prof. Dillmann (IAIM), // Institute for Computer Science and Engineering (ITEC/CSE), // University of Karlsruhe. All rights reserved. // ***** // Description: // Program searches *.jpg-Files in the current directory. Then: // calculation of center of gravity and principal axis for alignment, // Then gauging (measurement) of a given distance, // // Algorithms: // Spatial moments, central moments, // calculation of direction of major axis, // gauging (counting pixels to next d/w change), in [Pixels]. // // Comments: // OS: Windows 2000 or XP; Compiler: MS Visual C++ 6.0, // Libs used: IVT, QT, OpenCV. // ***** #include "Image/ByteImage.h" #include "Image/ImageAccessCV.h" #include "Image/ImageProcessor.h" #include "Image/ImageProcessorCV.h" #include "Image/PrimitivesDrawer.h" #include "Image/PrimitivesDrawerCV.h" #include "Image/IplImageAdaptor.h" #include "Math/Constants.h" #include "Helpers/helpers.h" #include "gui/QTWindow.h" #include "gui/QTApplicationHandler.h" #include <cv.h> #include <QString.h> #include <QStringList.h> #include <qdir.h> #include <iostream> #include <iomanip> #include <windows.h> #include <string.h> #include <math.h> using namespace std; // modified version of DrawLine(): returns sum of visited non-black pixels </pre> | | |

| Montag Februar 26, 2007 23:01 | main.cpp | Seite 2/6 |
|--|----------|-----------|
| <pre> // (but here also used for line-drawing) int WalkTheLine(CByteImage *pImage, const Vec2d &p1, const Vec2d &p2, int x, int y, int g, int b) { int pixelcount = 0; const double dx = p1.x - p2.x; const double dy = p1.y - p2.y; if (fabs(dy) < fabs(dx)) { const double slope = dy / dx; const int max_x = int(p2.x + 0.5); double y = p1.y + 0.5; if (p1.x < p2.x) { for (int x = int(p1.x + 0.5); x <= max_x; x++, y += slope) { if (pImage->pixels[int(y) * pImage->width + x] != 0) pixelcount++; PrimitivesDrawer::DrawPoint(pImage, x, int(y), r, g, b); } } else { for (int x = int(p1.x + 0.5); x >= max_x; x--, y -= slope) { if (pImage->pixels[int(y) * pImage->width + x] != 0) pixelcount++; PrimitivesDrawer::DrawPoint(pImage, x, int(y), r, g, b); } } } else { const double slope = dx / dy; const int step = (p1.y < p2.y) ? 1 : -1; const int max_y = int(p2.y + 0.5); double x = p1.x + 0.5; if (p1.y < p2.y) { for (int y = int(p1.y + 0.5); y <= max_y; y++, x += slope) { if (pImage->pixels[y * pImage->width + int(x)] != 0) pixelcount++; PrimitivesDrawer::DrawPoint(pImage, int(x), y, r, g, b); } } else { for (int y = int(p1.y + 0.5); y >= max_y; y--, x -= slope) { if (pImage->pixels[int(y) * pImage->width + int(x)] != 0) pixelcount++; PrimitivesDrawer::DrawPoint(pImage, int(x), y, r, g, b); } } } return pixelcount; } </pre> | | |

| Montag Februar 26, 2007 23:01 | main.cpp | Seite 3/6 |
|--|----------|-----------|
| <pre> void MomentCalculations(CByteImage *pImage, Vec2d &center, PointPair2d &orientation, double &theta) { // calculate moments IplImage *pIplInputImage = IplImageAdaptor::Adapt(pImage); CvMoments moments; CvMoments momInputImage, &moments, 1); //!/: treat grayvalues != 0 as 1 cvReleaseImageHeader(&pIplInputImage); // for center of gravity const double m00 = cvGetSpatialMoment(&moments, 0, 0); const double m01 = cvGetSpatialMoment(&moments, 0, 1); const double m10 = cvGetSpatialMoment(&moments, 1, 0); // for angle of major axis const double u11 = cvGetCentralMoment(&moments, 1, 1); const double u20 = cvGetCentralMoment(&moments, 2, 0); const double u02 = cvGetCentralMoment(&moments, 0, 2); theta = 0.0; // now: case differentiation: // cmp.: [Johannes Kilian 01], Simple Image Analysis by Moments] // online: http://serdis.dis.ulpgc.es/~itis-fia/FIA/doc/Moments/OpenCv/ // but: STILL AMBIGUOUS in n * 180 Degrees ! if (((u20 - u02) == 0) && (u11 == 0)) // 1 theta = 0.0; if (((u20 - u02) == 0) && (u11 > 0)) // 2 theta = PI / 4.0; if (((u20 - u02) == 0) && (u11 < 0)) // 3 theta = - (PI / 4.0); if (((u20 - u02) > 0) && (u11 == 0)) // 4 theta = 0.0; if (((u20 - u02) < 0) && (u11 == 0)) // 5 theta = - (PI / 2); if (((u20 - u02) > 0) && (u11 > 0)) // 6 theta = 0.5 * atan(2 * u11 / (u20 - u02)); if (((u20 - u02) > 0) && (u11 < 0)) // 7 theta = 0.5 * atan(2 * u11 / (u20 - u02)); if (((u20 - u02) < 0) && (u11 > 0)) // 8 theta = (0.5 * atan(2 * u11 / (u20 - u02))) + PI / 2; if (((u20 - u02) < 0) && (u11 < 0)) // 9 theta = (0.5 * atan(2 * u11 / (u20 - u02))) - PI / 2; Math2d::SetVec(center, m10 / m00, m01 / m00); // now: determine direction of major axis // go cross-like, start from COG, go to borders // count pixels... (cmp. visualization) Vec2d v; v.x = cos(theta) * 250 + center.x; v.y = sin(theta) * 250 + center.y; int count1 = WalkTheLine(pImage, center, v, 255, 0, 0); v.x = cos(theta + PI) * 230 + center.x; v.y = sin(theta + PI) * 230 + center.y; int count2 = WalkTheLine(pImage, center, v, 255, 255, 0); </pre> | | |

| Montag Februar 26, 2007 23:01 | main.cpp | Seite 4/6 |
|--|----------|-----------|
| <pre> v.x = cos(theta + PI/2) * 230 + center.x; v.y = sin(theta + PI/2) * 230 + center.y; int count3 = WalkTheLine(pImage, center, v, 128, 0, 0); v.x = cos(theta - PI/2) * 230 + center.x; v.y = sin(theta - PI/2) * 230 + center.y; int count4 = WalkTheLine(pImage, center, v, 64, 0, 0); if ((count1 > count2) && (count3 < count4)) theta = theta + PI; // Optional / for debugging: Console output // cout << "Area: " << m00 << endl; // cout << "Center (x,y): " << center.x << " " << center.y << endl; // cout << "Theta [DEG]: " << (theta * 180.0) / PI << endl << endl; } int main(int argc, char *argv[]) { double theta = 0.0; QString path = QDir::currentDirPath(); QDir dir(path); QStringList files = dir.entryList(QStringList::Files); if (files.empty()) { cout << "Error: could not find any *.jpg Files" << endl; return 1; } QStringList::Iterator it = files.begin(); QString buf = QFile::info(path, *it).fileName(); buf += ".jpg"; QApplicationHandler qtApplicationHandler(argc, argv); qtApplicationHandler.Reset(); // width, height must be multiples of 4 (!) CByteImage colorImage; if (!ImageAccessCV::LoadFromFile(&colorImage, buf.ascii())) { printf("Error: could not open input image file\n"); return 1; } CByteImage grayImage(colorImage.width, colorImage.height, CByteImage::eGrayScale); CByteImage binaryImage(colorImage.width, colorImage.height, CByteImage::eGrayScale); ImageProcessor::ConvertImage(&colorImage, &grayImage); // calculations in grayImage and binaryImage // drawings and writings in colorImage for display CQTWindow imgwindow1(colorImage.width, colorImage.height); imgwindow1.DrawImage(&colorImage); </pre> | | |

| Montag Februar 26, 2007 23:01 | main.cpp | Seite 5/6 |
|---|----------|-----------|
| <pre> imgwindow1.Show(); CQTWindow imgwindow2(binaryimage.width, binaryimage.height); imgwindow2.DrawImage(&binaryimage); imgwindow2.Show(); // main loop: cyclic loading all *.jpg in the directory and processing while (!qtApplicationHandler.ProcessEventsAndGetExit()) { buf = QFile::Info(path, *it).fileName(); buf += ".jpg"; cout << buf.ascii() << endl; if (!ImageAccessCV::LoadFromFile(&colorimage, buf.ascii())) { printf("error: could not open input image file\n"); return 1; } // Inversion: OpenCV calculates Moments for _white_ objects! ImageProcessor::ConvertImage(&colorimage, &grayimage); ImageProcessor::Invert(&grayimage, &grayimage); // (!) ImageProcessor::ThresholdBinarize(&grayimage, &binaryimage, 128); // Moments... Vec2d center; PointPair2d orientation; MomentCalculations(&binaryimage, center, orientation, theta); // Visualization / Output: // Center PrimitivesDrawerCV::DrawCircle(&colorimage, center, 3, 0, 255, 0, -1); // Two Lines to show coordinate system Vec2d v1, v2; v1.x = cos(theta) * 100 + center.x; v1.y = sin(theta) * 100 + center.y; WalkTheLine(&colorimage, center, v1, 255, 0, 0); v1.x = cos(theta + PI/2) * 100 + center.x; v1.y = sin(theta + PI/2) * 100 + center.y; WalkTheLine(&colorimage, center, v1, 255, 255, 0); ImageProcessor::Rotate(&binaryimage, &binaryimage, center.x, center.y, theta, true); // we gauge the cross section near the minor axis // (going parallel to the minor axis): v1.x = center.x+5; v1.y = center.y - 200; v2.x = center.x+5; v2.y = center.y + 200; int i = WalkTheLine(&binaryimage, v1, v2, 255, 255, 255); cout << "Gauging after alignment [pixel]: " << i << endl << endl; </pre> | | |

| Montag Februar 26, 2007 23:01 | main.cpp | Seite 6/6 |
|--|----------|-----------|
| <pre> char text[512]; sprintf(text, "Cross section in pixels: %d", i); PrimitivesDrawerCV::PutText(&colorimage, text, 20, 60, 0.8, 0.8, 255, 0, 100, 1); imgwindow1.DrawImage(&colorimage); imgwindow2.DrawImage(&binaryimage); //Sleep(1200); // oops, too fast to see anything.... ++it; if (it == files.end()) it = files.begin(); // until hell freezes over } return 0; } </pre> | | |

Appendix D

Datenblätter

Anmerkung: Wenn das Thema der Diplomarbeit auch Hardware-Komponenten bzw. den Aufbau eines Demonstrators eingeschlossen hat, so ist ein Anhang mit den wichtigsten Datenblättern sehr sinnvoll. Zum Einen können interessierte Leser direkt ohne Internetrecherche die Betriebsparameter der Komponenten einsehen, zum Anderen ist somit auch eine gute Dokumentation des Systems für die Bedienung durch andere Anwender als den Autor gegeben. Auch wenn die Datenblätter normalerweise online verfügbar sind, so erspart der beigefügte Anhang dem Anwender eine aufwändige Recherche. Die Anzahl Seiten sollte 25–30 nicht überschreiten.

Genau wie bei den Quelltextabschnitten im Anhang muss aber auch bei den Datenblättern ein kurzer Abschnitt vorweg geschickt werden, welcher die Auswahl der Datenblätter und die Relevanz erklärt.

Die einzufügenden Datenblätter sollten im pdf-Dateiformat vorliegen und nicht als Grafik, sondern als ganze Seite einzufügen.

Die nachfolgenden Datenblätter erläutern Systemparameter, Funktionsweise, Anschlussvarianten und Betriebsarten zu dem im Rahmen der vorliegenden Arbeit verwendeten Maxon-Motorregler ADS_E 50-10.

Der ADS_E 50/10 ist ein leistungsstarker PWM Servoverstärker für permanentmagneternetzte Gleichstrommotoren von 80 bis ca. 500 Watt Abgabeleistung.

Er ermöglicht folgende Betriebsarten:

- DC-Tachoregelung
- Digital-Encoderregelung
- IxR-Kompensation
- Stromregelung

Die gewünschte Betriebsart wird durch einen DIP-Schalter ausgewählt.

Der ADS_E 50/10 ist geschützt gegen Überstrom, Übertemperatur und Kurzschluss der Motorleitungen.

Durch die Verwendung moderner MOSFET Technologie wird ein Wirkungsgrad von bis zu 95% erreicht.

Eine eingebaute Motordrossel erlaubt in Verbindung mit der hohen PWM-Taktfrequenz von 50 kHz auch den Anschluss von Motoren mit sehr niedriger Induktivität.

Durch den weiten Eingangsspannungsbereich von 12 - 50 VDC kann der ADS_E 50/10 sehr flexibel mit unterschiedlichen Spannungsquellen verwendet werden. Das Europakarten-Format ist geeignet für den Einbau in ein 19"-Rack oder in ein Steckkartensystem. Dank der robusten Reglerauslegung des ADS_E 50/10 ist eine rasche und problemlose Inbetriebnahme möglich.



Inhaltsverzeichnis

| | | |
|----|---|----|
| 1 | Sicherheitshinweise | 2 |
| 2 | Technische Daten | 3 |
| 3 | Minimalverdrahtung bei den verschiedenen Betriebsmodi | 4 |
| 4 | Inbetriebnahme | 5 |
| 5 | Funktionsbeschreibung der Ein- und Ausgänge | 7 |
| 6 | Zusätzliche Einstellmöglichkeiten | 10 |
| 7 | Betriebszustandsanzeige | 12 |
| 8 | Fehlerbehandlung | 13 |
| 9 | EMV-gerechte Installation | 13 |
| 10 | Blockschaltbild | 13 |
| 11 | Steckerbelegung DIN 41612 Bauform H7/F24 | 14 |
| 12 | Massbild | 15 |
| 13 | Zubehör (nicht im Lieferumfang) | 15 |

Diese Bedienungsanleitung steht im Internet als PDF-Datei zur Verfügung unter www.maxonmotor.com, Rubrik «Service & Downloads», Sachnummer 168049.

Appendix E

Glossar

Anmerkung: Das vorliegende Glossar wurde ohne die Zuhilfenahme der speziellen Glossarumgebungen von Latex erstellt, um eine etwas freiere Formatierung nutzen zu können.

2,5D-Datensatz →Tiefenbild.

3D-Modell, 3D-Modellerfassung (optische) Der Begriff des 3D-Modells wird in der vorliegenden Arbeit für wasserdichte Oberflächenmodelle verwendet. Dies dient zur Abgrenzung gegenüber 3D-Volumenmodellen und →Tiefenbildern. Der Begriff der Optischen 3D-Modellerfassung umschließt hier neben der eigentlichen Sensordatenauswertung auch die →3D-Registrierung und die Schritte der Nachbearbeitung wie Glätten und Ausdünnen der Daten und Sticking-Operationen.

3D-Registrierung Vgl. Abschnitte 2.4, 4.3 und →Registrierung.

Aktive Musterprojektion Der Begriff der Aktiven Musterprojektion kennzeichnet Musterprojektionsverfahren, welche einen kalibrierten Musterprojektor voraussetzen. Vgl. entsprechend auch →Passive Musterprojektion.

Aktive Optische Verfahren Der Terminus Aktive Optische Verfahren kennzeichnet 3D-Datenerfassungsverfahren, welche eine zusätzliche Lichtquelle voraussetzen. Dies ist auch bei dem Verfahren der durch →Passive Musterprojektion ergänzten Stereopsis der Fall, dennoch wurde in der Darstellung zur Klassifizierung dieses Verfahren naheliegenderweise als Sonderfall hinter der Standard-Stereopsis aufgeführt (Abbildung 2.1).

Apertur, numerische Die numerische Apertur NA eines optischen Elementes, beispielsweise eines Objektivs, ist ein Maß für seine Lichtstärke bzw. sein Auflösungsvermögen. Die NA ist proportional zum Öffnungswinkel: $NA = n \cdot \sin \frac{\alpha}{2}$, mit n : Brechzahl. Oft wird statt der NA auch der Begriff der Blendenzahl F verwendet, hier besteht ein umgekehrt proportionaler Zusammenhang.

A priori-Szenenwissen Liegen vor Beginn der 3D-Erfassung bereits Informationen zur Geometrie des zu erfassenden Objektes im Erfassungssystem vor, so spricht man von a priori-Szenenwissen. Dies kann beispielsweise die Kenntnis der maximalen Objektausdehnung in z -Richtung sein, hilfreich zur Beschränkung des Suchraumes bei der Korrespondenzfindung. Ein weiteres Beispiel ist die Kenntnis der Anordnung der Profillinien auf der Objektoberfläche beim Lichtschnittverfahren; sie ermöglicht eine einfache und schnelle \rightarrow Vernetzung der \rightarrow Tiefenbilder.

Bildaufnehmer Der allgemeine Begriff des Bildaufnehmers steht in der vorliegenden Arbeit für zweidimensionale optische Sensoren bzw. Matrixkameras mit CCD- oder CMOS-Sensor.

Direkte Lineare Transformation, DLT \rightarrow Abschnitt 3.2.3.

Disparität Bei dem Ansatz des \rightarrow Stereosehens versteht man unter der Disparität den durch den Abstand der beiden Bildaufnehmer zueinander entstehenden Versatz der Abbildungen eines Objektpunktes. Dieser Versatz ist abhängig von der Entfernung des Objektes und entsprechend kann hieraus auf die Entfernung rückgerechnet werden.

Epipolargeometrie, Epipolarlinie, Epipol \rightarrow Abschnitt 3.2.5.

Extrinsische Parameter Hiermit sind Position und Orientierung eines \rightarrow Bildaufnehmers im vorgegebenen Weltkoordinatensystem bezeichnet. Sie sind gemeinsam mit den \rightarrow intrinsischen Parametern aus der Kalibrierung erhältlich (vgl. Abschnitte 3.2 und 3.3.1).

Fringe Pattern, Fringe Projection Der englische Begriff Fringe Pattern bzw. Fringe Projection (Fringe: dtsh. Rand, Saum) entspricht dem deutschen Begriff des Streifenmusters bzw. der Streifen-, Binär- oder Graycode-Projektion (vgl. Abschnitt 2.2.2).

Geodäsie (Griech.: geo: Erde, dasei: teilen). Nach der klassischen Definition von F. R. Helmert ist die Geodäsie die „Wissenschaft von der Ausmessung und Abbildung der Erdoberfläche.“ Ein wichtiges Messprinzip der Geodäsie ist die \rightarrow Triangulation.

Homographie Eine Homographie bedeutet die lineare Abbildung bzw. projektive Transformation einer Ebene im dreidimensionalen Raum auf eine andere Ebene in Form einer 3×3 -Transformationsmatrix. Eine gängige Anwendung ist die Entzerrung projizierter Bilder. Hier wird die \rightarrow Rektifikation durch eine Homographie hergestellt.

hsv-Farbraum, hsv-Modell Vgl. Rot-Grün-Blau-Farbraum, hier stellen die drei Werte nicht die drei Grundfarben dar, sondern stehen für *hue* (Farbwert), *saturation* (Sättigung) und *value* (Helligkeit). Die Umwandlung zwischen *rgb* und *hsv* ist möglich, allerdings nicht immer eindeutig (vgl. Abschnitt 2.2.4).

Intraoral Im Mundraum.

Intrinsische Parameter Hiermit sind Kenngrößen wie Brennweite f des Objektivs, Pixelgröße des Bildaufnehmers in x und y , Bildursprung (Position des Bildursprungs relativ zum Kamerakoordinatensystem) und Objektivverzeichnungen eines Bildaufnehmers mit zugehöriger Optik bezeichnet. Zusammen mit den \rightarrow extrinsischen Parametern sind sie erhältlich aus der Kalibrierung (vgl. Abschnitte 3.2 und 3.3.1).

Kondensor, Kondensorlinse Ein Kondensor ist ein optisches System aus einer oder mehreren Sammellinsen oder Spiegelflächen zwischen Lichtquelle und abzubildendem Objekt (Dia beim Projektor, Objektträger beim Mikroskop). Der Kondensor lenkt das Licht, welches das Objekt durchsetzt, in das abbildende Objektiv. Kondensorlinsen sind oft asphärisch ausgeformt, um eine möglichst kurze Brennweite und damit eine möglichst kleine Bauform des Gerätes zu ermöglichen.

Korrespondenzproblem, Korrespondenzanalyse Vgl. Abschnitt 2.2.1.

Lichtschnitt →Linienlaser.

Linienlaser Er stellt einen um eine Zylinder- bzw. Powell-Linse erweiterten Punktlaser dar. Der Laserstrahl wird hierdurch fächerförmig aufgeweitet und kann somit auf einer Projektionsfläche als gut erkennbare Laserlinie abgebildet werden. Anwendung findet der Linienlaser häufig in triangulationsbasierenden Laserscannern, das Verfahren wird dann als Lichtschnittverfahren bezeichnet. Das Verfahren ist bis auf die einfachere Korrespondenzauf"-lösung identisch mit der zeitlich codierten Musterprojektion (Binäre M., Graycode-M., vgl. 2.2.1), dennoch wird im Sprachgebrauch unterschieden. Diese Verfahren werden meist als *auf strukturiertem Licht basierend* bezeichnet.

Matching Wird der Begriff des Matchings wie in der vorliegenden Arbeit auf Oberflächendatensätze (→Tiefenbilder) angewandt, so kennzeichnet er die Bestimmung der Transformation, um die Datensätze in ein gemeinsames Koordinatensystem zu überführen. Er ist gleichbedeutend mit dem Begriff der →3D-Registrierung.

Merging Der Begriff des Merging wird in der vorliegenden Arbeit für den Vorgang der Eingliederung eines bereits registrierten, vernetzten →2,5D-Datensatzes in einen bestehenden 3D-Datensatz auf dem Weg zur Erstellung eines geschlossenen →3D-Modells verwendet. Hierzu ist zumindest teilweise eine Neuvernetzung erforderlich (zur Vorgehensweise und zu Problemen hierbei vgl. beispielsweise [Völzow 03]).

Meshing →Vernetzung.

One-Shot-Verfahren Der Begriff kennzeichnet musterprojektionsbasierte Verfahren zur optischen 3D-Datenakquisition, bei welchen ein einziges aufgenommenes Bild ausreicht, um ein →Tiefenbild zu erstellen. Der Begriff wurde von Marc Proesmans geprägt [Proesmans 96a].

Passive Musterprojektion Im Gegensatz zu dem Verfahren der →Aktiven Musterprojektion kommen hier zwei oder mehr kalibrierte Kameras, ergänzt durch einen Musterprojektor, zum Einsatz (→Stereopsis). Dieser ist nicht kalibriert, sondern dient nur dazu, die Szene zu aktivieren bzw. der Szene neue Merkmale zuzuordnen, um dieserart die Korrespondenzfindung zu erleichtern.

Phasenschiebeverfahren, Phase Shifting Vgl. Abschnitt 2.2.3

Punktwolke Mit dem Begriff der Punktwolke wird in der vorliegenden Arbeit allgemein ein nicht →vernetzter →2,5D-Datensatz oder 3D-Datensatz bezeichnet. Äquivalent ist im Englischen der Begriff *Point Cloud* bzw. – spezifisch im unorganisierten Fall – *Scattered Point Data*.

Region Growing, Regionenwachstum Das Region Growing-Verfahren ist ein regionenbasiertes Segmentierungsverfahren. Bei diesem Verfahren wachsen homogene Regionen ausgehend von vorgegebenen Saatpunkten. Zu einer dieserart entstehenden Region werden angrenzende Bildpunkte hinzugenommen, solange ein vorgegebenes Homogenitätskriterium erfüllt ist (typisches Kriterium: Schwellwert für die Grauwertdifferenz).

Registrierung Der Begriff der Registrierung wird in der vorliegenden Arbeit als Kurzform für \rightarrow 3D-Registrierung verwendet. Dies dient der Abgrenzung gegenüber der anders lautenden Definition der Registrierung in der (medizinischen) Bildverarbeitung. Hier wird der Begriff für die Zuordnung von 2D-Bildinformationen zueinander gebraucht (monomodal: Bsp. radiologische Daten, multimodal: Bsp. radiologische Daten zu Ultraschalldaten).

Rektifikation Unter Rektifikation versteht man den Vorgang, die Bildebenen zweier Kameras, die in der Realität nicht in der gleichen Ebene liegen, sondern verschoben und/oder gegeneinander rotiert sind, auf eine gemeinsame virtuelle Bildebene abzubilden, in welcher dann die jeweiligen Epipolarlinien kollinear zueinander und zu den Basisvektoren verlaufen. Ein Vorteil, der hieraus entsteht, ist die Möglichkeit der Anwendung einfacherer und schnellerer Korrelationsverfahren. Die Rektifikation wird durch eine \rightarrow Homographie hergestellt (vgl. auch [Trucco 98,7.3.7]).

Running Sum \rightarrow Sum Table

Schärfentiefe Mit Schärfentiefe wird die Ausdehnung des scharf abgebildeten Bereiches entlang der optischen Achse eines optischen Systems bezeichnet (auch: Abbildungstiefe, umgangssprachlich oft auch: \rightarrow Tiefenschärfe). Zur rechnerischen Bestimmung wird für den Zerstreuungskreis auf dem \rightarrow Bildaufnehmer ein maximaler Durchmesser vorgegeben (vgl. auch [Schröder 77]).

Splatting Ein Problem bei der Visualisierung von 3D-Oberflächendaten ist die Notwendigkeit der zeitaufwändigen \rightarrow Vernetzung. Neue Verfahren zur Visualisierung von 3D-Daten setzen keine Nachbarschaftsbeziehungen zwischen den Punkten voraus, sondern dehnen im Moment des Renderings bzw. der Anzeige der Daten durch Projektion auf eine Fläche (Bildschirm) einzelne den Punkten zugewiesene Elementfunktionen so weit aus, dass sich der visuelle Eindruck einer geschlossenen Fläche ergibt. Man unterscheidet hierbei Surface Splatting und Volume Splatting (vgl. auch QSplat, [Rusinkiewicz 00]).

Stereopsis, Stereosehen Unter dem Begriff Stereopsis versteht man den Vorgang der visuellen Wahrnehmung der Tiefe oder Entfernung von Objekten. Der Begriff leitet sich ab aus den griechischen Worten *Stereo* für räumlich und *Opsis* für Sehen oder Sicht. Die Tiefeninformation wird hierbei gewonnen aus dem entfernungsabhängigen Versatz der Bildinformationen zwischen den zwei Augen oder Bildaufnehmern.

Sum Table Der Begriff Sum Table bezeichnet die Ablage einer fortlaufend gebildeten Summe (Running Sum) in einer n-dimensionalen Tabelle. Bsp. (1D): $S_{\sum x_k} = [\sum_1 x_k, \sum_2 x_k, \dots, \sum_n x_k]$. Teilsumme $\sum_{k=l}^m x_k$ kann hiermit nun schnell gebildet werden. Für den zweidimensionalen Fall vgl. Abschnitt 3.4.2: Optimierte ZNCC.

Tiefenbild Mit dem Begriff Tiefenbild wird eine Darstellungsform für Höheninformationen bezeichnet, welche in einem Grauwertbild mit den Bildpunktkoordinaten x, y diesen auch einen z -Wert in Form ihres Grauwertes zuordnet. Die Darstellungsform kann leicht umgewandelt werden in eine $\rightarrow 3D$ -Punktwolke, deren Punkte alle eindeutig in die xy -Ebene projizierbar sind (man spricht dann von 2,5D-Datensatz). Ein Tiefenbild kann somit kein geschlossenes $\rightarrow 3D$ -Modell darstellen, sondern nur eine Sicht auf ein Objekt vermitteln. Seltener auch: Tiefenkarte.

Tiefenschärfe \rightarrow Schärfentiefe.

Transluzenz Die Transluzenz bezeichnet die Eigenschaft eines Objektes, teilweise lichtdurchlässig zu sein. In Abgrenzung zur Transparenz kann man Transluzenz als Lichtdurchlässigkeit beschreiben und Transparenz als Bilddurchlässigkeit.

Triangulation Der Begriff der Triangulation wird in der vorliegenden Arbeit zweifach verwendet. Zum einen bezeichnet er den Vorgang der Vernetzung einer \rightarrow Punktwolke zu einem Dreiecksnetz (Bsp.: Delaunay-Triangulation, Abschnitt 4.5.1), zum anderen die Berechnung der Tiefe auf Basis von Dreiecksbeziehungen bei der optischen 3D-Datenerfassung (vgl. Abbildung E.1 und Abschnitt 3.2.4).

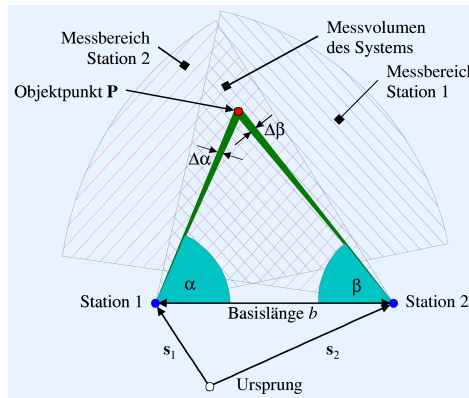


Figure E.1: Triangulation (Bildquelle: [Wikipedia 07]).

„(...) Von zwei verschiedenen Stationen an den Positionen s_1 und s_2 wird der zu bestimmende Zielpunkt P angepeilt. Dabei erhält man die beiden Winkel α und β mit der Genauigkeit $\Delta\alpha$ und $\Delta\beta$. Unter Kenntnis der Basislänge b kann man dann die Koordinaten von P relativ zum Koordinatenursprung bestimmen.“ (Zitat aus: [Wikipedia 06]). Ein bekanntes optisches Winkelmessgerät der \rightarrow Geodäsie ist der Theodolit.

Vernetzung Mit dem Begriff der Vernetzung wird die Erstellung einer Oberfläche aus einer Punktwolke durch Einfügen von Kanten zwischen den Punkten bezeichnet. Im allgemeinen Fall entstehen hieraus n -Ecke, im speziellen Fall Dreiecke (\rightarrow Triangulation).

Weltkoordinatensystem Der Begriff Weltkoordinatensystem ist in der vorliegenden Arbeit anders belegt als in der klassischen Geodäsie: Gemeint ist ein dreidimensionales kartesisches Koordinatensystem, in welchem die Daten in der Einheit [Meter] bzw. [Millimeter] eingetragen sind, welches aber nicht in Bezug steht zu geographischen Landeskordinaten.

Bibliography

- [Abdel-Aziz 71] Y. I. Abdel-Aziz and H. M. Karara, „Direct linear transformation from comparator coordinates into object space coordinates in close-range photogrammetry“, in: Symposium on Close-Range Photogrammetry, issue 11, pp. 1–18, University of Illinois at Urbana-Champaign, 1971.
- [AutTech 07] Firma Automation Technology GmbH in 22946 Trittau, Produktübersicht, Downloads und Datenblätter. URL: <http://www.automationstechnology.de>
- [Dang 06] T. Dang, C. Stiller and C. Hoffmann, „Self-calibration for Active Automotive Stereo Vision“, Proc. of the IEEE Intelligent Vehicles Symposium, pp. 364–369, Japan, Tokyo, 2006.
- [Fisher 96] R. B. Fisher and D. K. Naidu. „A Comparison of Algorithms for Subpixel Peak Detection“ in: Advances in Image Processing, Multimedia and Machine Vision, Springer-Verlag, Heidelberg, 1996. Online erhältlich via CiteSeer.
- [Gamma 04] E. Gamma, D. Riehle u. a., „Entwurfsmuster – Elemente wiederverwendbarer objektorientierter Software“, Addison-Wesley-Verlag, München, 2004.
- [Gühring 02] J. Gühring, „3D-Erfassung und Objektrekonstruktion mittels Streifenprojektion“, Dissertation, Universität Stuttgart, Institut für Photogrammetrie, 2002. Online erhältlich via URL: http://elib.uni-stuttgart.de/opus/volltexte/2006/2715/pdf/Guehring_diss.pdf
- [Hoppe 02] H. Hoppe, C. Kübler, J. Raczkowsky und H. Wörn, „Ein neues und leicht zu implementierendes Modell zur präzisen Kalibration von Kameras und Videoprojektoren“, in: Medicine Meets Virtual Reality (MMVR 02), pp. 229–232, Newport Beach, USA, 2002.
- [Huckle 02] T. Huckle und S. Schneider, „Numerik für Informatiker“, Springer-Verlag, Heidelberg, 2002.
- [Käferstein 98] B. Käferstein, „3D-Verformungsmessungen auf 10nm genau – Grundlagen und Anwendungen der Speckle-Interferometrie“, Institutsmitteilung Nr. 23, Institut für Maschinenwesen (IMW), Technische Universität Clausthal, 1998.
- [Kitware 07] Firma Kitware Inc. (USA), Online-Produktpräsentation und freier Download der Software-Bibliothek VTK – Visualization Toolkit. URL: <http://www.kitware.com>
- [Luhmann 03] T. Luhmann, „Nahbereichsphotogrammetrie: Grundlagen – Methoden – Anwendungen“, Wichmann-Verlag, Heidelberg, 2003.

- [Marqu 63] D. Marquardt, „An Algorithm for Least-Squares Estimation of Nonlinear Parameters“, SIAM J. Appl. Math., vol. 11, pp. 431–441, 1963.
- [Pentax 07] Firma Pentax, Online-Produktspektrum und Datenblätter. URL: <http://www.pentax.de>
- [Pollefeys 03] M. Pollefeys, L. Van Gool u.a., „3D Capture of Archaeology and Architecture with a Hand-Held Camera“, Proc. of the ISPRS workshop on Vision Techniques for Digital Architectural and Archaeological Archives, Ancona, Italy, The Int. Archive of the Photogrammetry, Remote Sensing and Spatial Information Sciences, Vol. XXXIV, Part 5/W12, pp. 262–267, July 2003.
- [SAC 07] Firma SAC GmbH in 76149 Karlsruhe, Online-Produktspektrum und Datenblätter. URL: <http://www.sac-vision.de>.
- [SuK 07] Firma Schäfter und Kirchhoff GmbH in 22525 Hamburg, Online-Produktspektrum, bes. interessant: die Applikationsschriften. URL: <http://www.sukhamburg.de>
- [TI 07] Firma Texas Instruments, Online-Produktspektrum, Datenblätter und Downloads. URL: <http://www.ti.com>
- [TIS 07] Firma The Imaging Source Europe GmbH in 28215 Bremen, Produktübersicht, Downloads und Datenblätter. URL: <http://www.theimagingsource.com>
- [Trolltech 07] Fa. Trolltech Inc., Produktinformationen und Download zur Software-Bibliothek Qt. URL: <http://www.trolltech.com>
- [Trucco 98] E. Trucco and A. Verri, „Introductory Techniques for 3-D Computer Vision“, Verlag Prentice Hall, 1998.
- [VC 07] Firma Vision Components in 76275 Ettlingen, Online-Produktübersicht, Datenblätter und Tutorials, URL: <http://www.vision-components.com>
- [Vuylsteke 90] P. Vuylsteke and A. Oosterlinck, „Range image acquisition with a single binary-encoded light pattern“, Tagungsband: IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI), Vol. 12, No. 2, pp. 148–164, 1990.
- [Wörn 05] H. Wörn und U. Brinkschulte, „Echtzeitsysteme“, Springer-Verlag, Heidelberg, 2005.
- [Zhang 00] Z. Zhang, „Flexible camera calibration by viewing a plane from unknown orientations“, IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI), Vol. 22, No. 11, pp. 1330–1334, 2000, URL: <http://ftp.research.microsoft.com/pub/tr/tr-98-71.pdf> (31.05.2007).

