



Analyse von 3D-Punkt-Triangulations- Verfahren für Structure-from-Motion

Verteidigung Großer Beleg

Max Köhler

max.koehler@mailbox.tu-dresden.de

Dresden, 26.10.2011



Gliederung

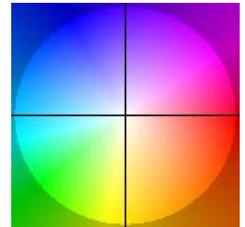
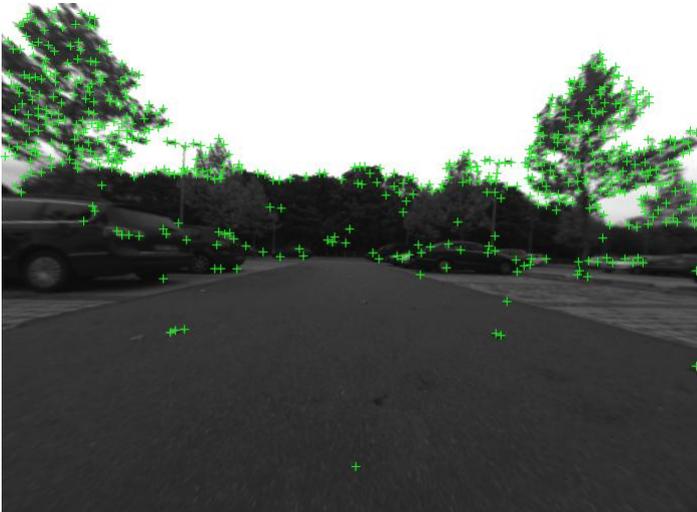
- (1) Motivation
- (2) Grundlagen
- (3) Triangulation
- (4) Implementierung
- (5) Ergebnisse
- (6) Ausblick

Motivation

Einsatzgebiete

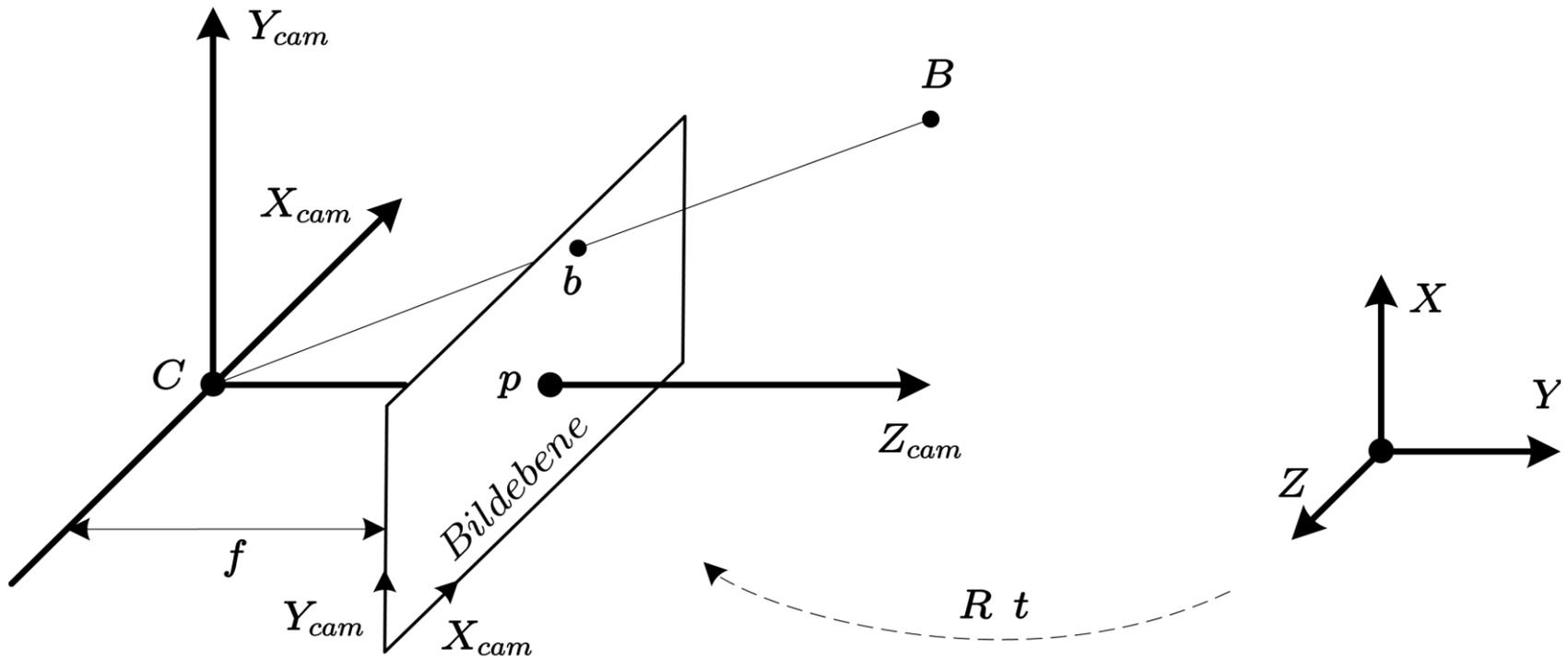
- Structure-from-Motion
 - Tiefeninformation über mehrere Ansichten oder Bildsequenz
 - nur eine Kamera benötigt
 - min. 7 Punkte zur Bewegungsrekonstruktion
 - Erkennung von statischen Hindernissen
- Nutzen für Fahrerassistenzsysteme
 - allgemeiner 3D-Umgebungssensor (Orientierung/Positionierung)
 - Parkassistenz (Kollisionsvermeidung)

Motivation dichter optischer Fluss



deutlich mehr Bildpunkte durch dichten optischen Fluss

Grundlagen Lochbildkamera



[HZ08, Abbildungen 6.1-6.3]

Grundlagen

Projektionsmatrix

Kameramatrix

$$K = \begin{bmatrix} f_x & s & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix}$$

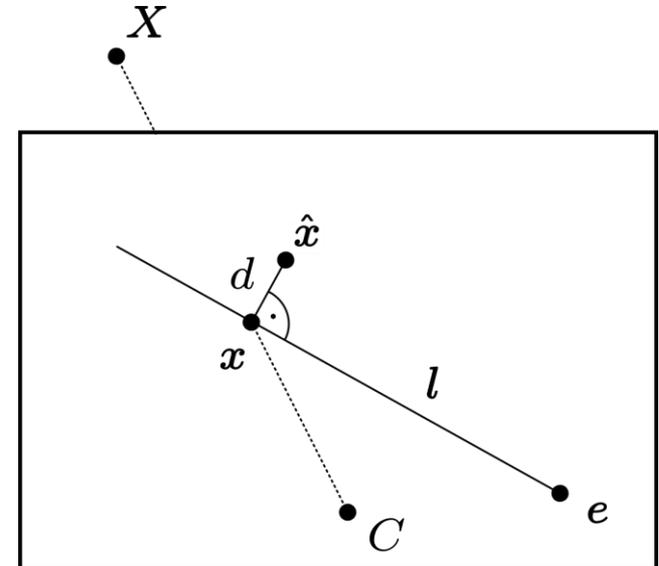
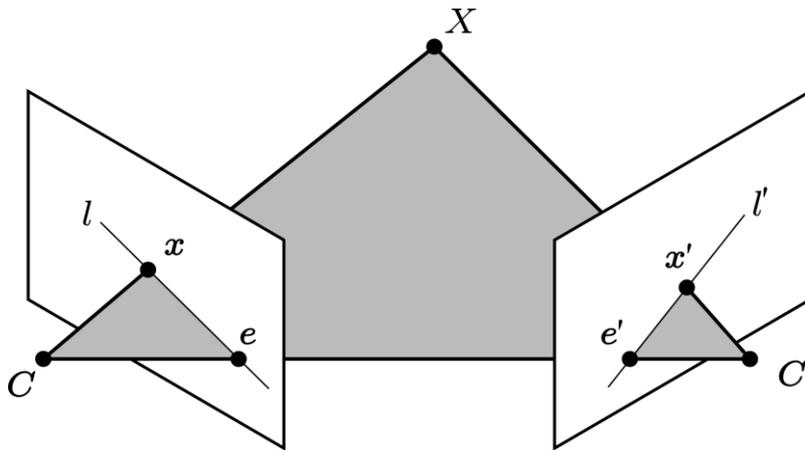
Projektionsmatrix

$$P = K \cdot R \cdot [E_3 \mid -\tilde{C}]$$

Abbildung von Punkten (homogene Koordinaten)

$$b_k \cdot [b_x \quad b_y \quad 1]^T = P \cdot [B_x \quad B_y \quad B_z \quad 1]^T$$

Grundlagen Epipolargeometrie



[HZ08, Abbildungen 9.1, 12.3]

Epipolarbedingung $x'^T F x = 0$ in der Regel nicht erfüllt.

Triangulation

Lösungsmethoden

- Minimierung Reprojektionsfehler
- exakte Lösung bei Gaußschem Rauschen möglich
- lineare Triangulation einfach und genau
- Lösung über Singulärwertzerlegung [HZ08, Anhang A5.3]

$$AX = \begin{bmatrix} x_x P_3 - P_1 \\ x_y P_3 - P_2 \\ x'_x P'_3 - P'_1 \\ x'_y P'_3 - P'_2 \end{bmatrix} X = [U \mid D \mid V] X = 0$$

Triangulation

Untersuchte Verfahren

- Jacobi-Rotation

$$A_{i+1} = J_{pq}^T \cdot A_i \cdot J_{pq}$$

$$V = \prod J_{pq}$$

- Inverse Iteration

$$X = \lim_{n \rightarrow \infty} \left[(A^T A)^{-n} v \right] = \left(\frac{1}{\lambda_4} \right)^n a_4 w_4$$

$$v = a_1 w_1 + a_2 w_2 + a_3 w_3 + a_4 w_4$$
$$\lambda_1 > \lambda_2 > \lambda_3 > \lambda_4$$

- Lineare Gleichungen

$$X = [x \ y \ z \ 1]^T$$

$$Y = [x \ y \ z]^T$$

$$AX = 0 \rightarrow CY = c$$

Triangulation

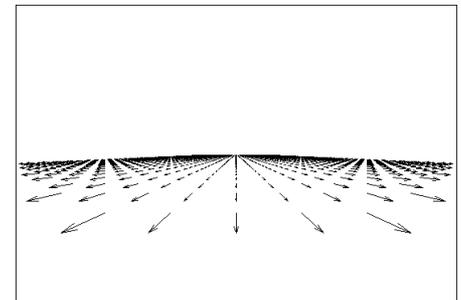
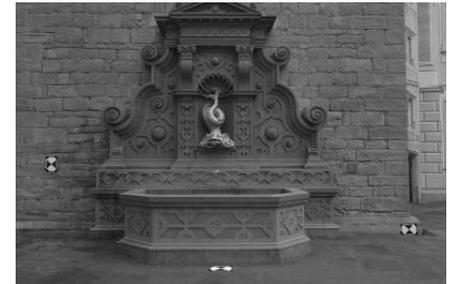
Testszenen

Voraussetzungen

- 3D-Punkte
- Projektionsmatrizen für jede Aufnahme
- Bilder zu jeder Aufnahme

Verwendete Szenen

- Gang [HZ08]
- Brunnen [SHVG+08]
- Synthetische Szenen (Vorwärts- und Seitwärtsbewegung)



Triangulation

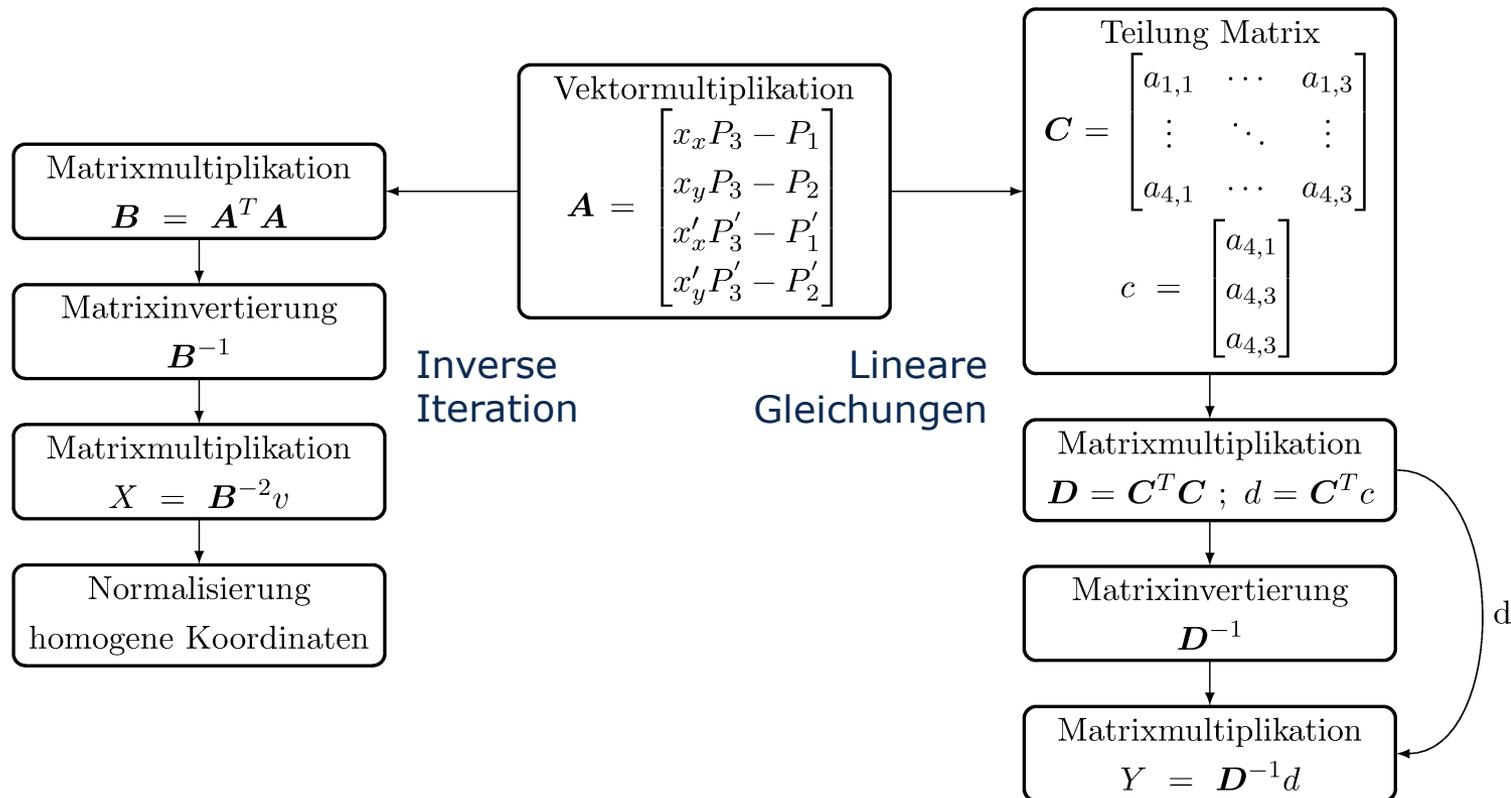
Wertbasierte Auswahl

- Kriterien
 - Reprojektionsfehler
 - Abstände Originalpunkte ↔ triangulierte Punkte
- Gewichtung der Punkte anhand der Entfernung vom Kamerazentrum

Verfahren (Iterationen)	Gang	Brunnen	Vorwärts- bewegung	Seitwärts- bewegung	geom. Mittel
SVD-Funktion	+	++	++	+++	++
Inverse Iteration (2)	+	++	++	+++	++
Lineare Gleichungen	+	++	++	++	+
Jacobi Rotation (10)	---	--	--	--	--
Jacobi Rotation (40)	--	-	-	-	-

Triangulation

Architekturbasierte Auswahl



Implementierung

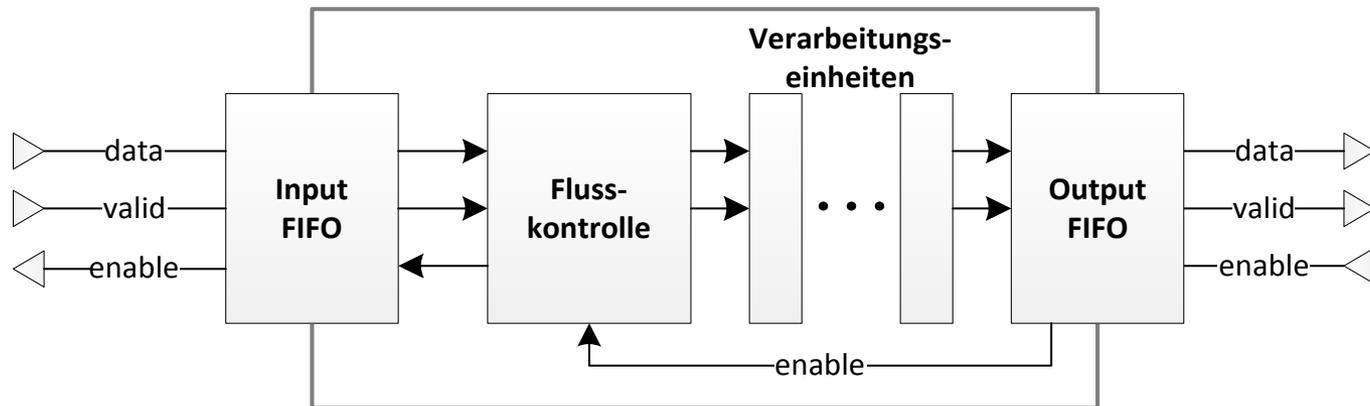
Anforderungen

- Zielarchitektur: Xilinx Virtex-5
- Echtzeitanforderungen
 - Sensorgröße: 1 Megapixel
 - 25 frames per second
 - Taktfrequenz: 125 MHz
 - ↳ 5 Takte pro trianguliertem Punkt
- Berücksichtigung Matrixgröße und Initialisierung
 - spalten- bzw. zeilenweise Berechnung
 - Takte für den Austausch der Projektionsmatrizen
 - ↳ 4 Takte pro trianguliertem Punkt

Implementierung

Anforderungen Interface

- FlexWAFE Interface [CLSW⁺09]
- backpressure kompensiert durch FIFO



Implementierung

Zahlenformat

- Untersuchungen zur Genauigkeit auf Basis von IEEE 754 Single (float)
- Festkommaformat: weniger Ressourcen und geringere Latenz
- Orientierung an DSP48E Slices für Multiply-Add Einheiten

Wortbreite Faktor 1 [Bit]	Wortbreite Faktor 2 [Bit]	Anzahl DSP48E- Slices
18	25	1
25	35	2
35	42	4

Eingangsbreiten Multiplikation [Xil10, Tabelle 4-2]

Implementierung

Zahlenformat

- Anpassung an Kamerasystem und Translation
 - Datenbasis:
 - normalisierte Projektionsmatrizen
 - 3D-Punktwolke
 - MATLAB ermittelt Maximalwerte
 - C-Programm zur Simulation der Wortbreiten
- ↳ ansteigende Wortbreiten während der Rechnung
- ↳ Maximalwerte + Wortbreite
= Anzahl der Nachkommastellen
- ↳ generische Nachkommastellen über Package-Konstante

Implementierung

Zahlenformat

- Gaußverfahren mit Festkommaformat zu ungenau
- Matrixinvertierung über Adjunkte

$$A^{-1} = \frac{1}{\det(A)} \operatorname{adj}(A)$$

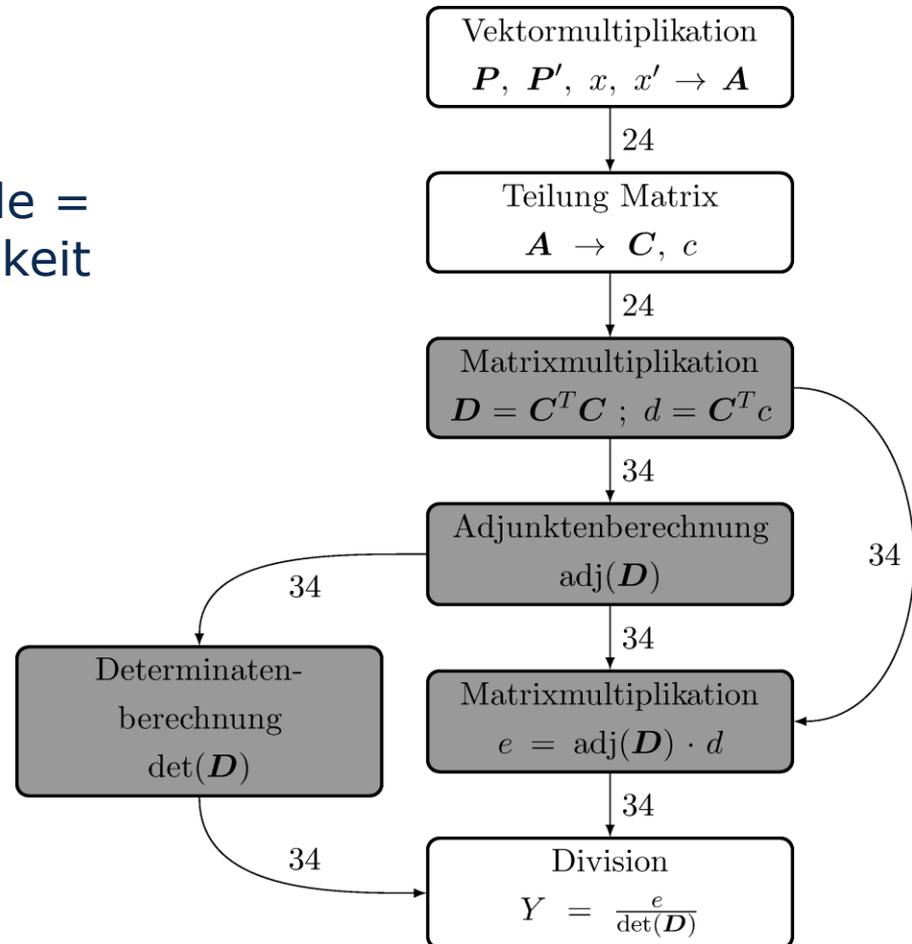
$$A^{-1} = \begin{pmatrix} a & b & c \\ d & e & f \\ g & h & i \end{pmatrix}^{-1} = \frac{1}{\det(A)} \begin{pmatrix} ei - fh & ch - bi & bf - ce \\ fg - di & ai - cg & cd - af \\ dh - eg & bg - ah & ae - bd \end{pmatrix}$$

$$\det(A) = a \cdot (ei - fh) + b \cdot (fg - di) + c \cdot (hd - eg)$$

Implementierung

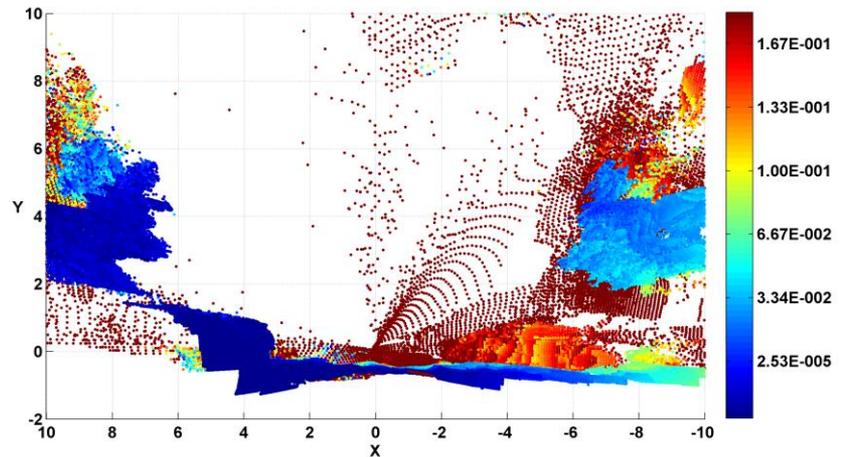
Übersicht Pipeline

- Wortbreite an Kanten
- grau hinterlegte Module = intern höhere Genauigkeit von 48 Bit



Ergebnisse

Genauigkeit



Kamerasicht
Angaben in Metern

- Genauigkeitsverluste durch Festkommaformat
- bis 10 Meter im Bereich von wenigen Zentimetern

Ergebnisse

Hardwareressourcen

- $\frac{3}{4}$ Ressourcen im Vergleich zu Gleitkomma eingespart (Gleitkommaeinheiten aus Xilinx Core Generator)
- weniger als 2,5% der Slices in FPGA XC5VSX240T
- kritische Ressource: 68 DSP48E-Slices
- ohne Anpassungen auf Virtex-6 oder Virtex-7 einsetzbar
- Taktfrequenz $\leq 250\text{MHz}$ durch kurze Pfade

Ergebnisse

Verarbeitungsgeschwindigkeit

		Programm / Hardware		
Punktdaten	Verfahren	MATLAB	C++ [ope11]	FPGA
Brunnen	Lin. Gleichungen	29.424	80.161	62.500.000
	SVD	29.366	44.491	-
Parkplatz	Lin. Gleichungen	29.499	109.155	62.500.000
	SVD	29.467	46.384	-

Punkttriangulationen / Sekunde

- Steigerung triangulierter Punkte durch Entfernung von Abstraktionsebenen
- FPGA bietet Ressourcen für größere Sensoren bzw. höhere Framerraten

Ausblick und Fazit

Ausblick

- weitere Untersuchungen zu Abweichungen über gesamte Verarbeitungskette
- Datensätze mit Gültigkeits- bzw. Qualitätsfaktor versehen

Fazit

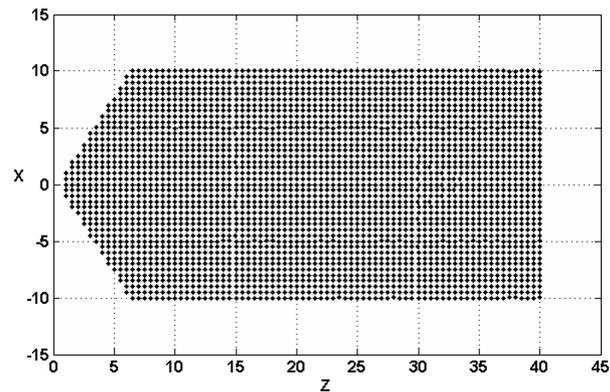
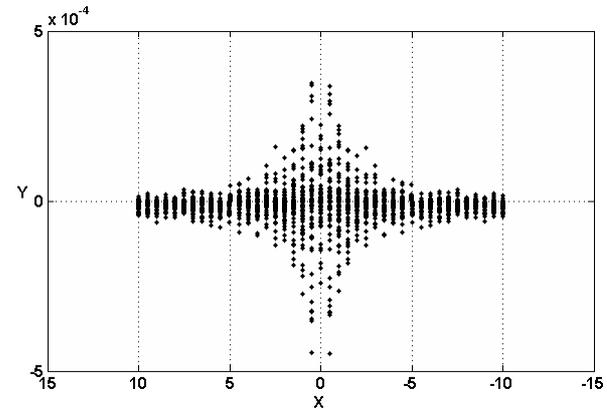
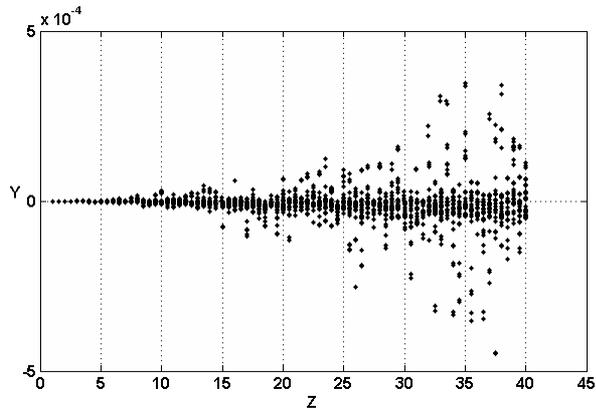
- Genauigkeit ausreichend für nahe Umfeldrekonstruktion (z.B. Parkszenarien)
- einfache Anpassbarkeit an andere Kamerasysteme
- Steigerung der Datenraten um mehr als zwei Größenordnungen

Quellen

- [CLSW⁺09] Carmo Lucas, A. do ; Sahlbach, H. ; Whitty, S.; Heithecker, S. ; Ernst, R. :
Application Development with the FlexWAFE Realtime Stream Processing Architecture
for FPGAs.
In: ACM Transactions on Embedded Computing Systems Special Issue on Congurable
Computing. Oktober 2009.
- [HZ08] Hartley, R. I. ; Zisserman, A.:
Multiple view geometry in computer vision.
Cambridge University Press, 2008.
<http://www.robots.ox.ac.uk/~vgg/hzbook/>
- [ope11] OpenCV.
<http://opencv.willowgarage.com>, 2011.
- [SHVG⁺08] Strecha, C. ; Hansen, W. von ; Van Gool, L. ; Fua, P. ; Thoennessen, U.:
On benchmarking camera calibration and multi-view stereo for high resolution imagery.
In: IEEE Conference on Computer Vision and Pattern Recognition, 2008.
- [Xil10] Xilinx:
Virtex-5 FPGA XtremeDSP Design Considerations User Guide 3.4.
http://www.xilinx.com/support/documentation/user_guides/ug193.pdf
Juni 2010.

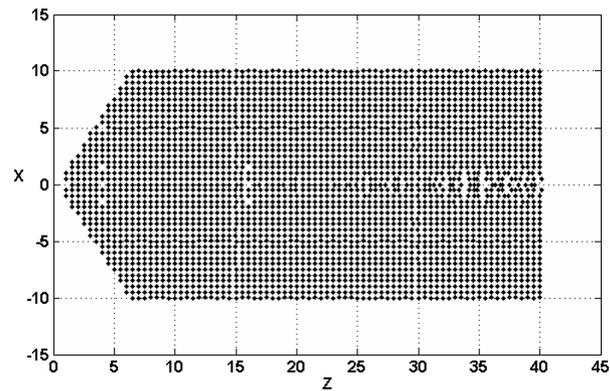
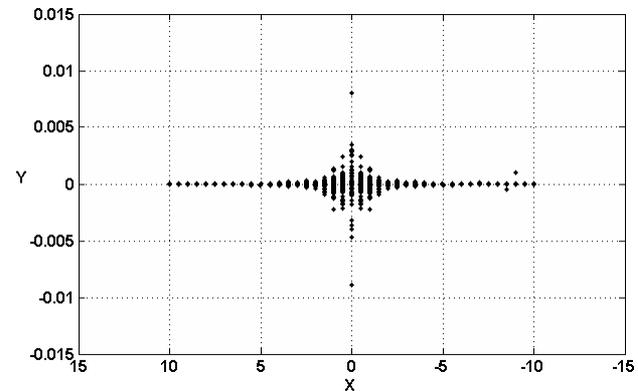
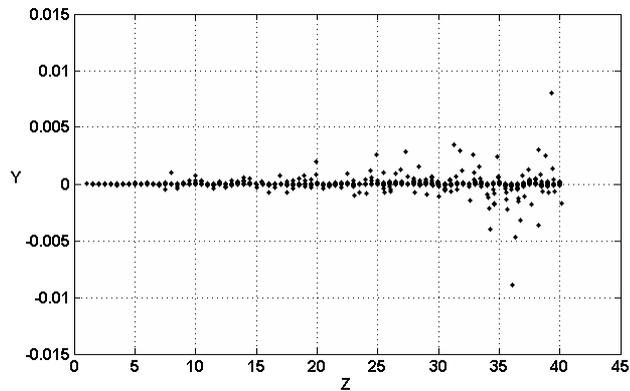
Triangulation

Singulärwertzerlegung



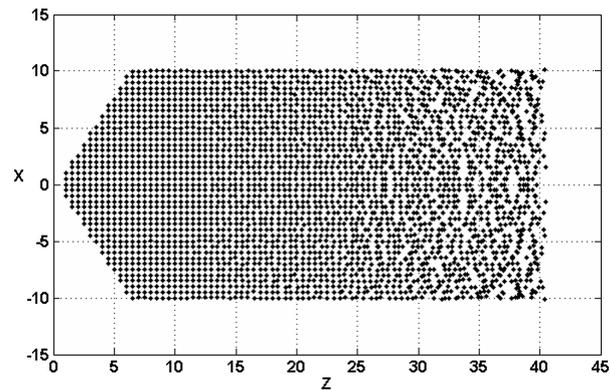
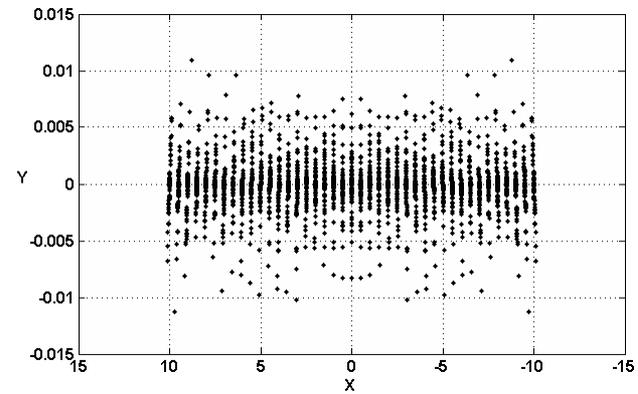
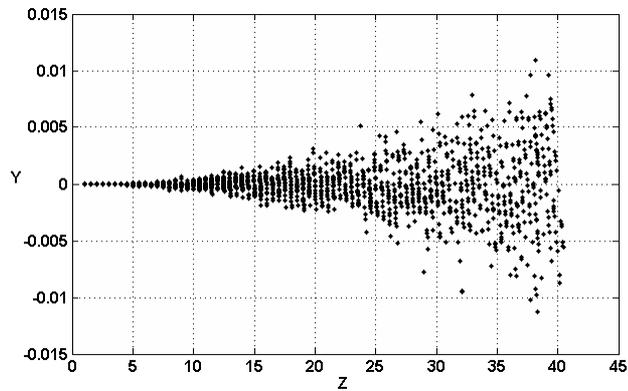
Triangulation

Inverse Iteration



Triangulation

Lineare Gleichungen



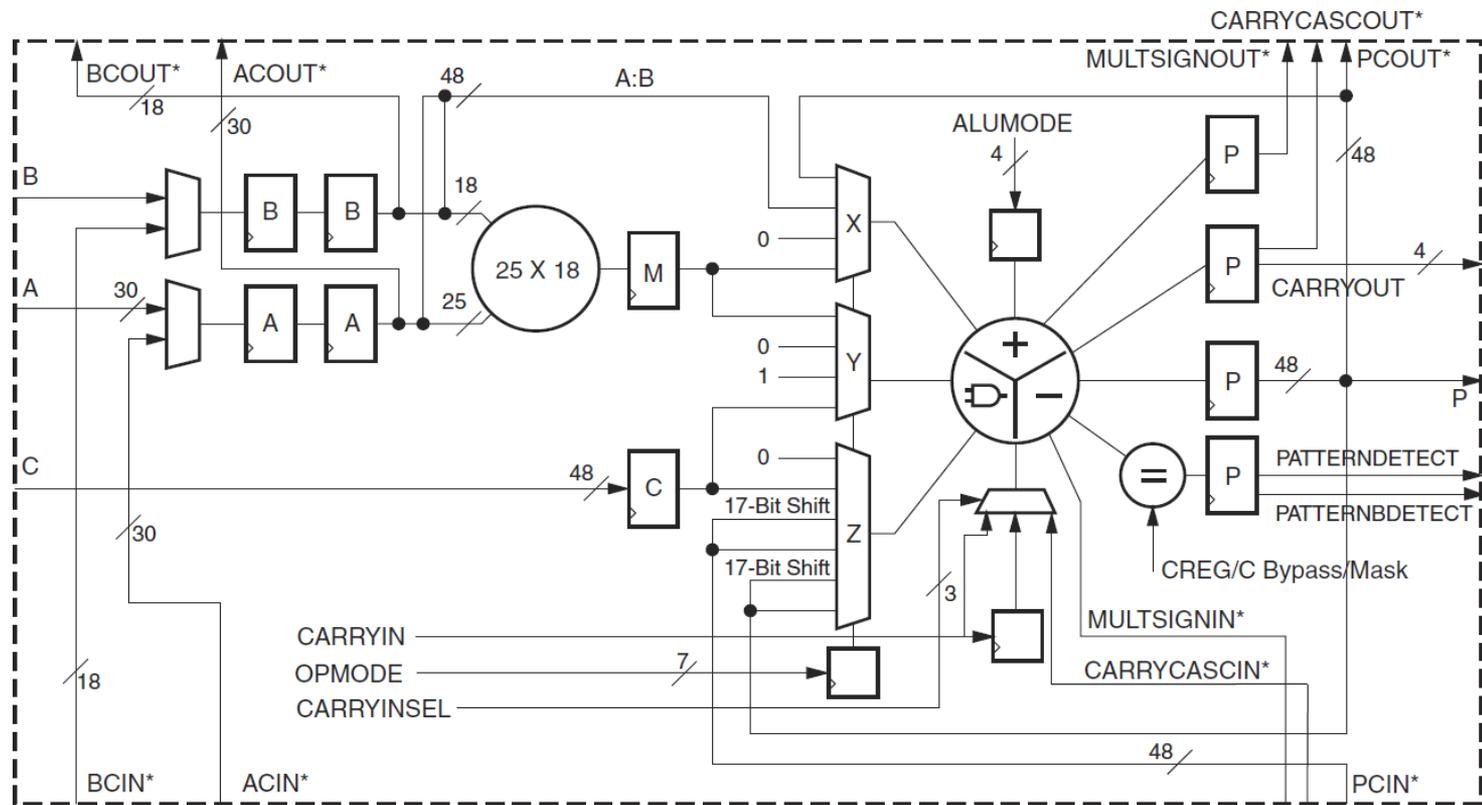
Triangulation

gewichtete Mittelwerte

Verfahren (Iterationen)	Gang	Brunnen	Vorwärts- bewegung	Seitwärts- bewegung	Geometrisches Mittel
SVD-Funktion	2,6915E-02	1,8425E-04	4,5336E-05	6,3360E-06	1,9427E-04
Inverse Iteration (2)	2,6913E-02	1,8424E-04	2,3589E-04	5,9546E-06	2,8889E-04
Lineare Gleichungen	2,9605E-02	1,8424E-04	2,8246E-03	1,7315E-04	1,2780E-03
Jacobi Rotation (10)	1,5456E+02	2,3645E+00	3,2569E+00	6,0905E+00	9,2273E+00
Jacobi Rotation (40)	2,5040E-01	4,0292E-01	1,5320E+00	2,4975E-01	4,4326E-01

Angaben in Radiant

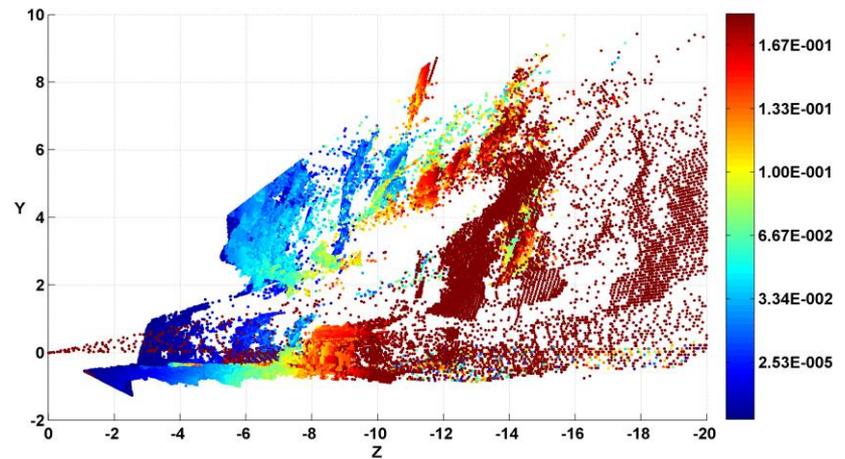
Implementierung DSP48E-Slice



[Xil10, Abbildung 1-1]

Ergebnisse

Genauigkeit



Seitenansicht
Angaben in Metern

Ergebnisse

Hardwareressourcen

	Gleitkomma Anzahl	Festkomma Anzahl	FPGA XC5VSX240T Anteil
Number of Slice LUTs	8646	1950	1%
Number used as Memory	1220	276	1%
Number of occupied Slices	3673	901	2%
Number of LUT Flip Flop paris used	11951	3129	-
Number with an unused Flip Flop	1785	191	6%
Numer with an unused LUT	3305	1179	37%
Number of fully used LUT-FF pairs	6861	1759	56%
Number of 36k FIFO used	7	3	1%
Number of 18k FIFO used	0	3	1%
Number of DSP48Es	90	68	6%