

Erfassung der 3D-Umgebungsgeometrie

- Systematisch angewendet, wird Entfernungsmessung zu: Abbildung der Umgebungsgeometrie in 3D
- Unverzichtbar für Navigation mobiler Systeme (Kollisionsvermeidung! auch Lokalisierung, Objekterkennung)

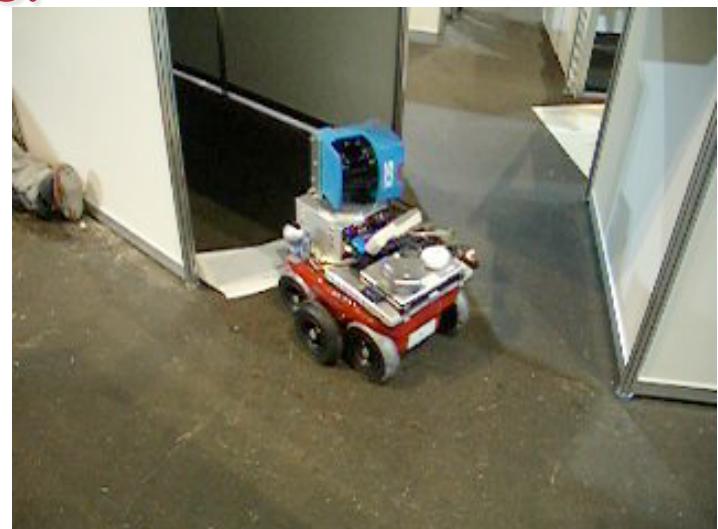
Direkte Realisierung: 3D-Laserscanner



+



=



Kurt3D (AIS, UOS)

Univ. Hannover

Nachteil: Messung braucht Zeit (sec.-Bereich)
Problem bei Dynamik (Eigen-, Umgebungs-dynamik)

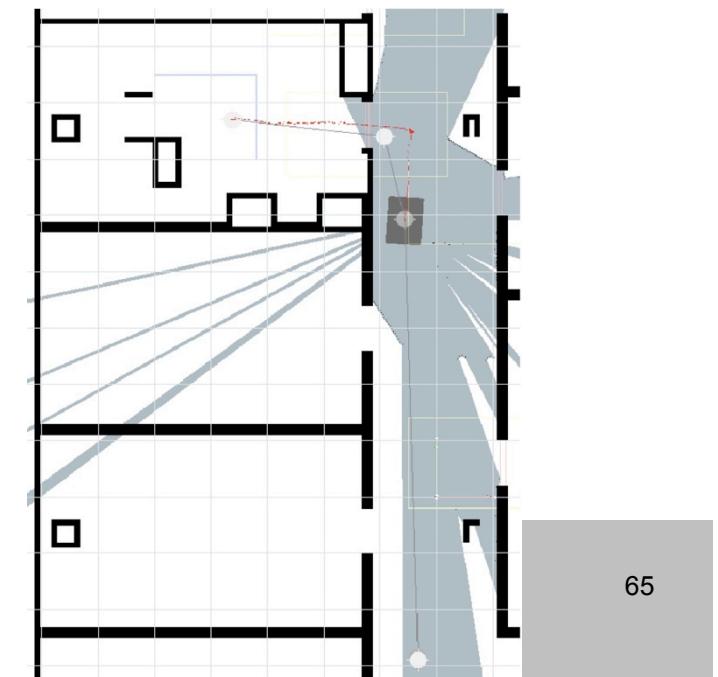
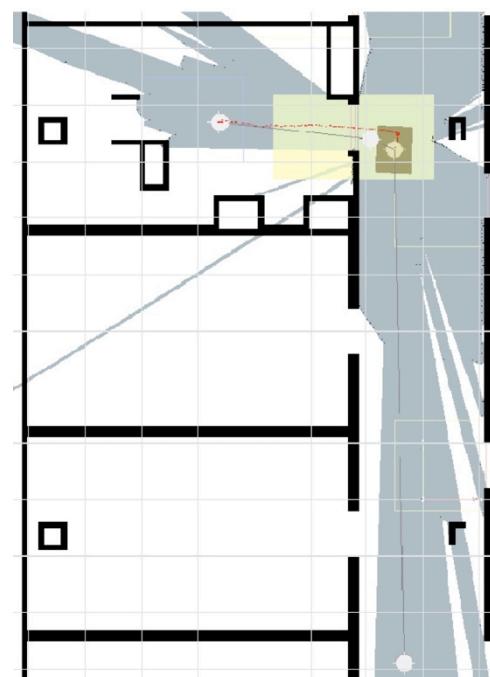
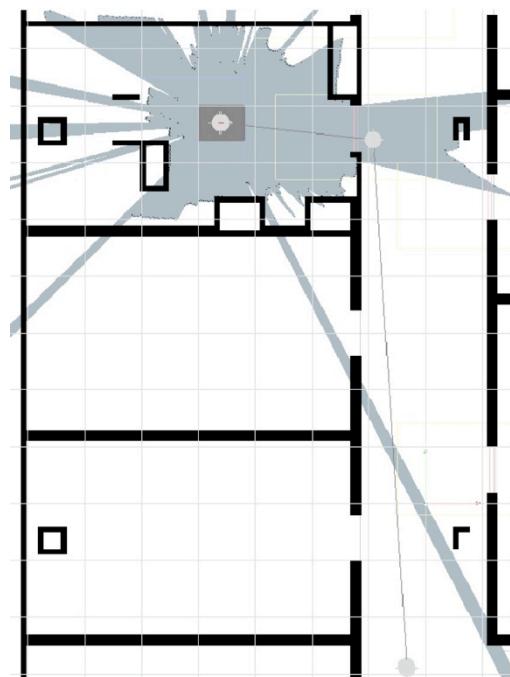
3D-Daten zur Kollisionsvermeidung



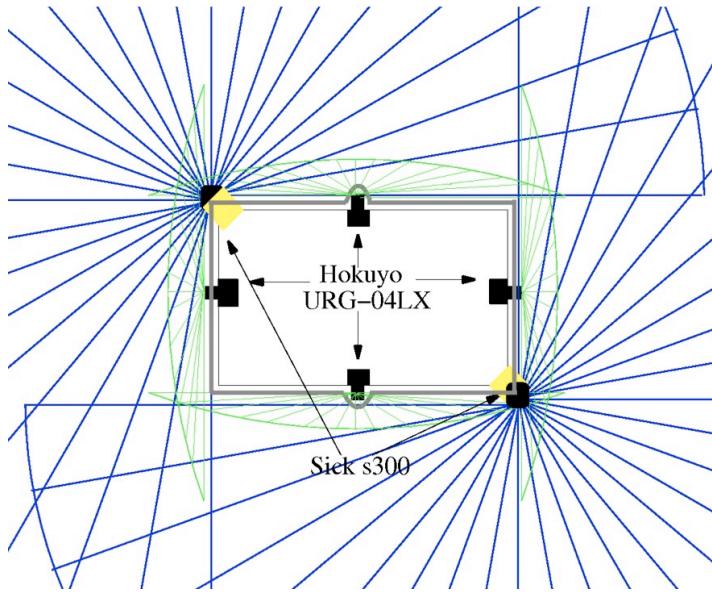
Virtuelle Entfernungsscans: Vorher



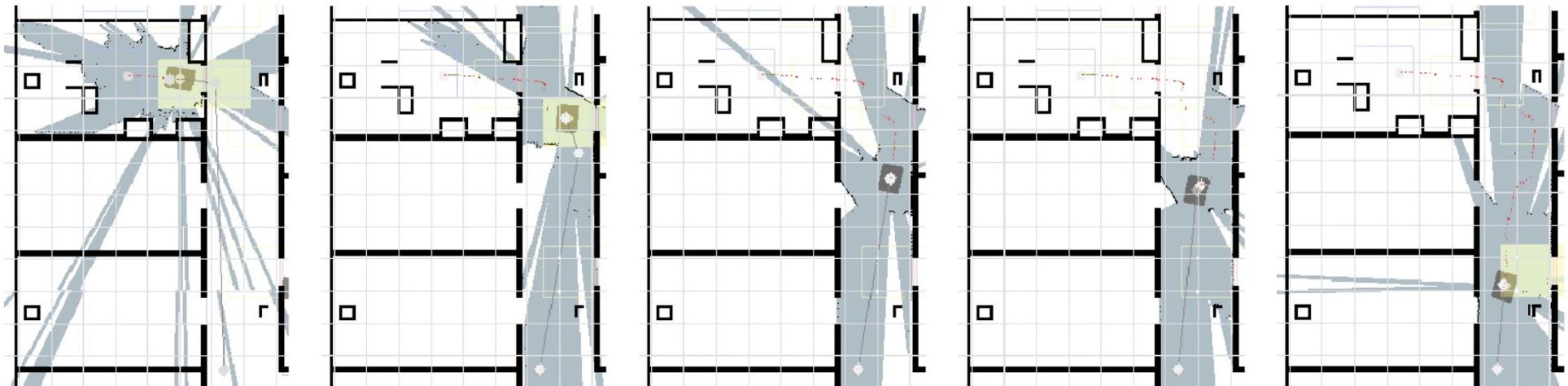
Laserscanner
(hier SICK S300 mit
Öffnungswinkel 270°) misst
nur in seiner Scanebene



Virtuelle Entfernungsscans: Nachher



- Fest schräg aufwärts montierte Scanner erfassen bei Bewegung Hindernisse in 3D im Nahbereich
 - Projektion ihrer Daten in die Scan-Ebene der S300 ergibt die 2D-Projektion des 3D-Freiraums
- Online-Messung ohne Zeitverzug!



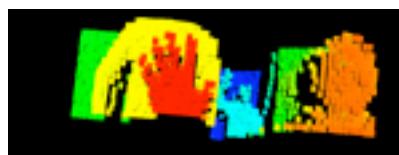
3D-Kameras

Gleichzeitige Aufnahme Textur und Tiefe (moduliertes IR-Licht)

PMD
Technologies
Siegen



Detector	1/2" global shutter PMD sensor
Pixel Dimensions	40 µm x 40 µm
Resolution	160 (h) x 120 (v) QQVGA
Optical Fill Factor	30 %
Receiver Optics	f = 12 mm; f/# = 1.2; C-mount
Max. Range	7.5 m
Z-Resolution	> 6 mm
Max. Field of View	40°
Illumination Power	approximately 4W optical
Wavelength	850 nm
Frame Rate (3D)	up to 10 fps
Digital Interface	IEEE 1394a, Ethernet (IEEE 802.3u), RS 232
Data Format	12-bit single phase or 16-bit 3D plus 16-bit grayscale



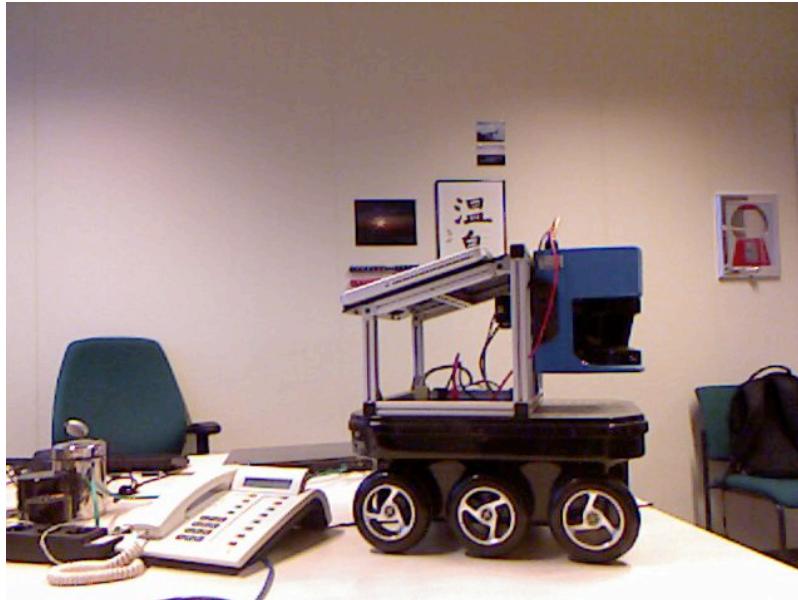
Grauwert- und Tiefenbild

CSEM SwissRanger, Zürich



Number of pixels (ROI option)	Up to 124 x 160
Depth resolution	Down to 5 mm
Wavelength of illumination	870 nm
Illumination power	720 mW optical
Maximum range	7.5 m
Frame rate [†]	Up to 30 fps
Diagonal field of view (HxW)	+/- 30° (+/- 21° x +/- 23°)
Interface	USB 2.0
Connector	Mini USB Type-B
Power supply	+ 12 V / 1.5 A DC
Power consumption	18 W max.
Lens	f= 8 mm, F/# = 1.4, M 12x0.5
Dimensions (mm)	135 (W) x 45 (H) x 32 (D)
Weight	0.2 kg

RGB-D-Kameras: Kinect und andere



- erzeugt „farbige Punktwolken“ (3D-Punktwolke überlagert mit Texturwert)
- Öffnung $57^\circ \times 43^\circ$
- 30Hz Framerate
- relativ hohes Rauschen



Microsoft Kinect



ASUS Xtion Pro



2.6 Kameras und Kameramodelle

... reguläre (CMOS, CCD) Digitalkameras aller möglichen Bauarten von 15–15.000€

Vorteile

- „flächiger“ Sensor mit hoher Auflösung
- (potenziell) hohe Frequenz i.Vgl. zu Roboterbewegung
- „natürlicher“ Sensor aus Sicht von Nutzern

Probleme

- hohe Datenrate bei viel Redundanz
- 3D-Umgebungsgeometrie muss aus 2D-Repräsentation extrahiert werden
- Beleuchtungsabhängigkeit
 - Bei Dunkelheit ausleuchten (Energieaufwand!)
 - Bei Varianz extreme Verarbeitungsprobleme

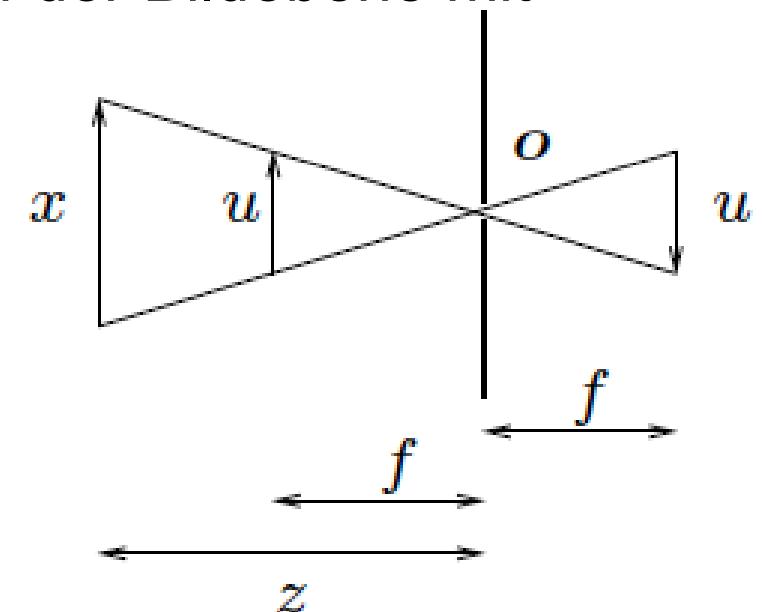
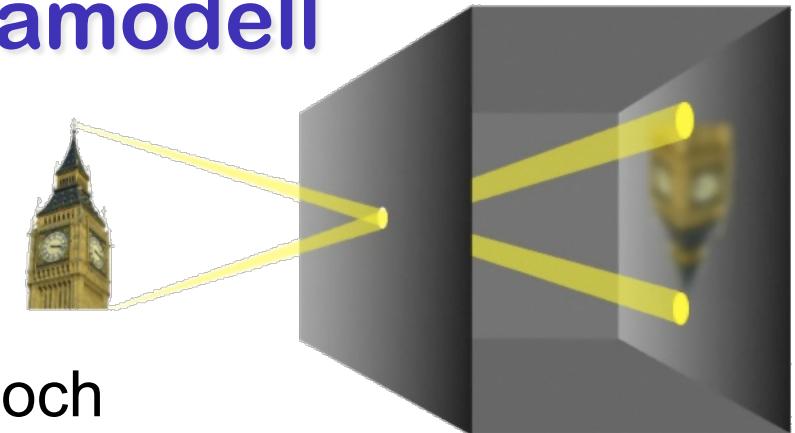


Das Lochkameramodell

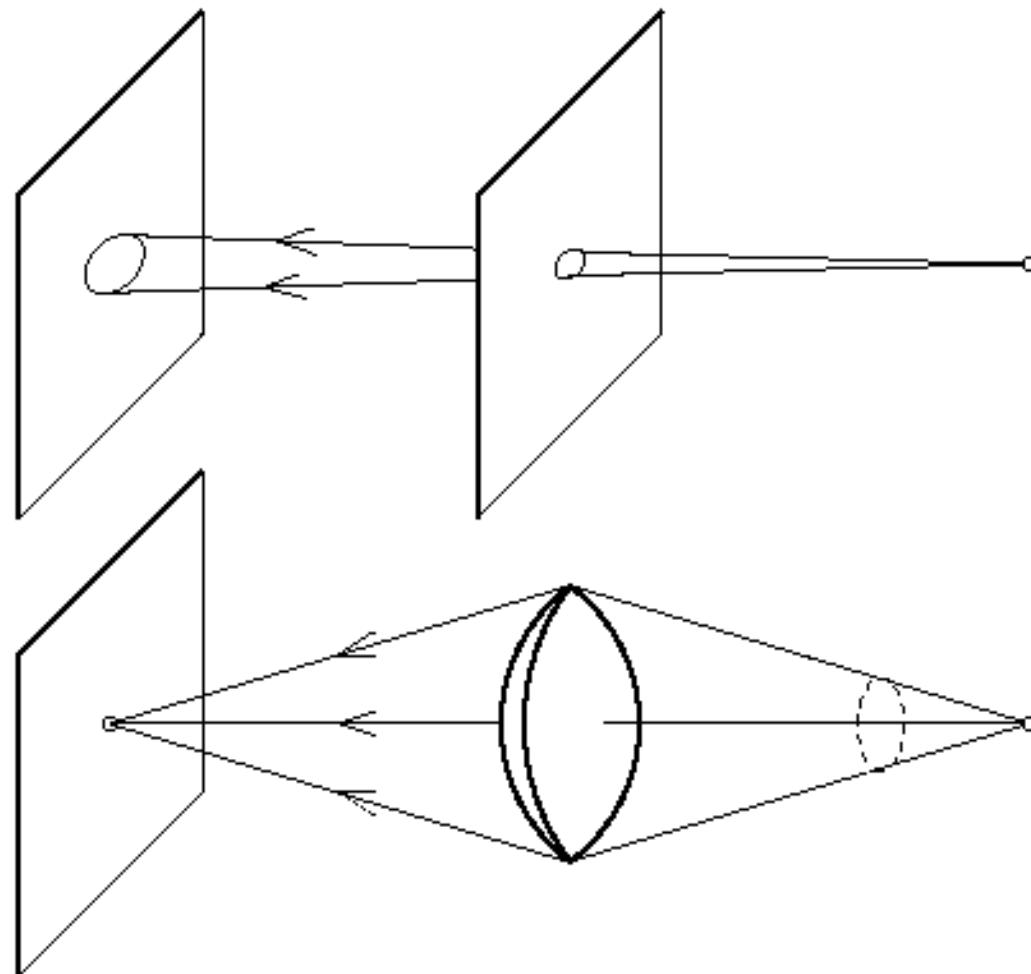
- Definiere Kamerakoordinatensystem ${}^c x, {}^c y, {}^c z$ (3D) durch Bildebene und optische Achse
- Für Koordinatenursprung im Kameraloch o gilt (Strahlensatz): Gegeben Brennweite f , wird Raumpunkt $({}^c x, {}^c y, {}^c z)$ abgebildet auf Punkt (u, v) auf der Bildebene mit

$$\frac{-f}{{}^c z} = \frac{u}{{}^c x} = \frac{v}{{}^c y}$$

**Ein Punkt im Raum wird eindeutig abgebildet.
Ein Bildpunkt entspricht nicht eindeutig einem Raumpunkt!**



Warum baut man dann Kameras mit Linsen?



→ Weil Kameras lichtempfindlich sein und scharf abbilden sollen!

Aber: Linsenfehler!

- Verzeichnungen
 - chromatische Aberration
 - ...
- Kamerakalibrierung

Reale Optiken sind komplizierter ...

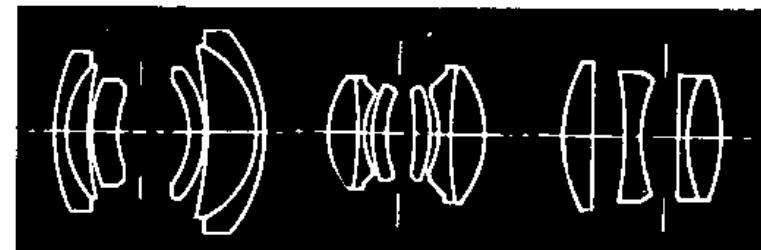


Abbildung in homogenen Koordinaten

Aus der Projektiven Geometrie wissen wir (s. A.1.3 im Buch):

**Transformation des n D-Raums in $(n+1)$ D
homogene Koordinaten erleichtert das Rechnen**

(alle Abbildungen als lineare Matrix-Operationen)

Lineare Abbildungsgleichung

... für homogene Darstellung
von Punkt $p = ({}^c x, {}^c y, {}^c z)$
im Raum
(im Kamera-Koordinatensystem)
in homogene Bildkoordinaten

$$\begin{pmatrix} u' \\ v' \\ w' \end{pmatrix} = \begin{pmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} \cdot \begin{pmatrix} {}^c x \\ {}^c y \\ {}^c z \\ 1 \end{pmatrix}$$

Rücktransformation in 2D-Bildebene: $u = u' / w'$, $v = v' / w'$

Abbildung durch reale Kameras

Intrinsische Kameraparameter

- Optische Achse schneidet Bildebene in (u_0, v_0)
 - Brennweite ausgedrückt in Pixelzahl k_x, k_y horizontal & vertikal: $\alpha_x = f/k_x$ und $\alpha_y = f/k_y$
 - Reale Pixelanordnung ist nicht notwendig ideal rechteckig: Scherung s des Bildes
- Homogene Abbildungsmatrix wird zu

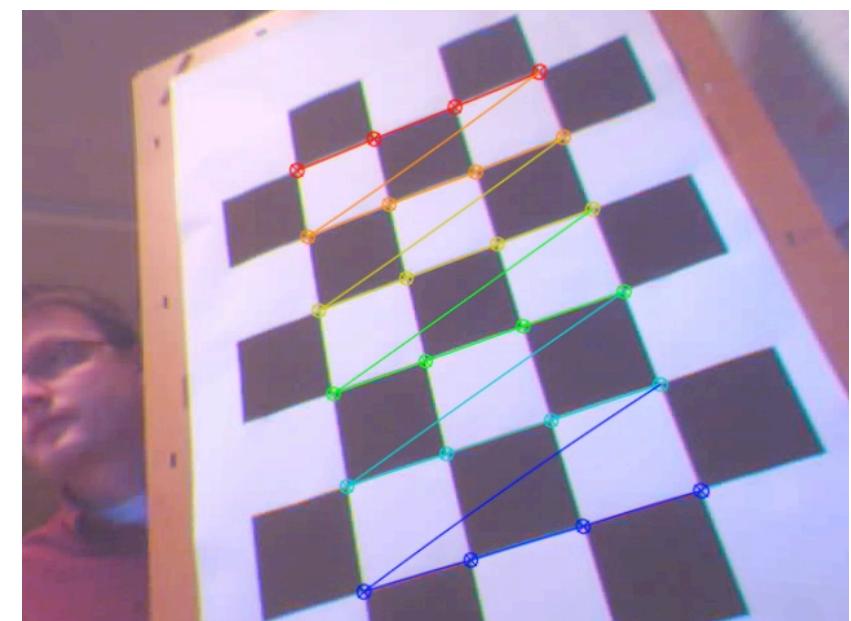
$$\begin{pmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} \xrightarrow{\quad} \begin{pmatrix} \alpha_x & s & u_0 & 0 \\ 0 & \alpha_y & v_0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix}$$

Intrinsische Kameraparameter bekannt und/oder bei Kamerakalibrierung zu bestimmen

Kamerakalibrierung

... bestimmt die intrinsischen Kameraparameter (soweit unbekannt/unsicher) und die optischen Verzeichnungen

Keine zwei Webcams sind gleich!



Transformation in Weltkoordinaten

Meist ist das Kamerakoordinatensystem nicht das, auf welches sich die Umgebungswahrnehmung beziehen soll (andere: Roboterkoordinaten, Flurkoordinaten, ...)

→ Transformation durch Rotation und Translation

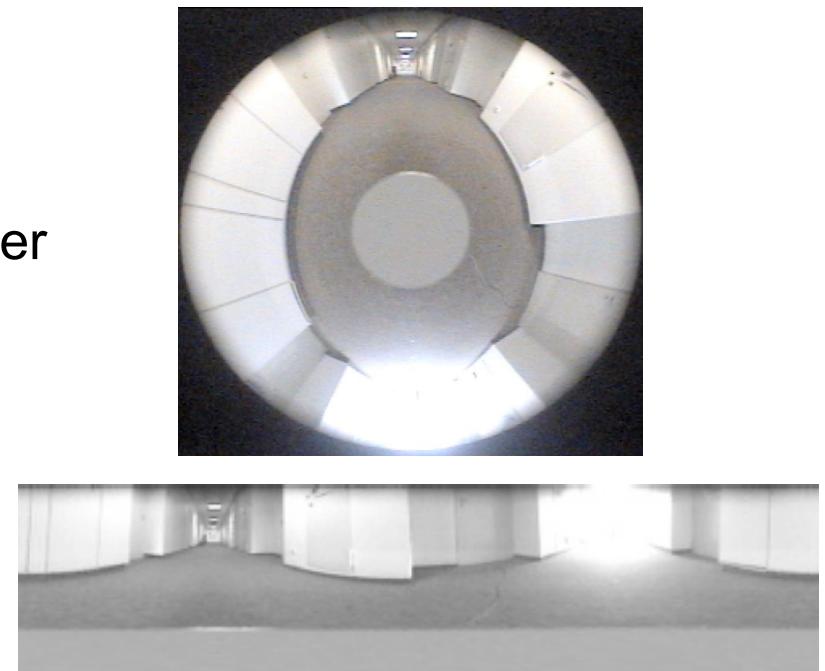
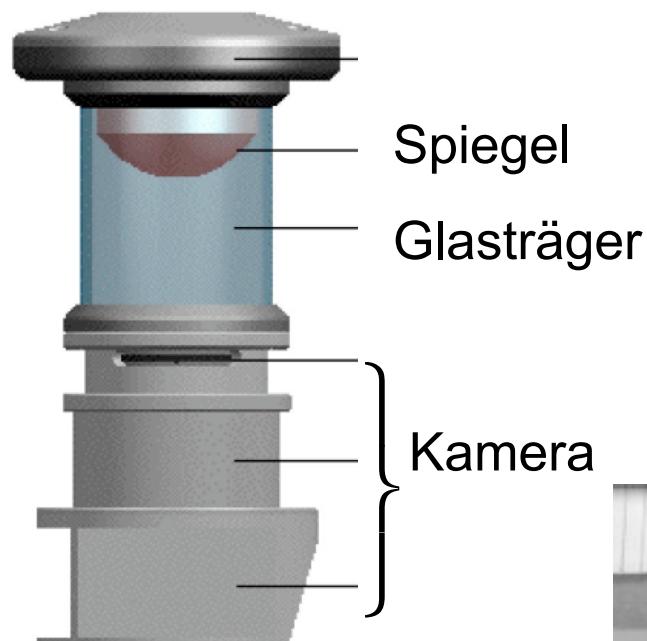
$$\begin{pmatrix} {}^c x \\ {}^c y \\ {}^c z \\ 1 \end{pmatrix} = \begin{pmatrix} \mathbf{R} & \mathbf{t} \\ 0 & 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} {}^w x \\ {}^w y \\ {}^w z \\ 1 \end{pmatrix}$$

R Rotationsmatrix (3x3), **t** Translationsvektor (1x3)

Omnidirektionale Kameras

zum Beispiel eine **katadioptrische** Kamera (Optik hat Linsen und Spiegel)

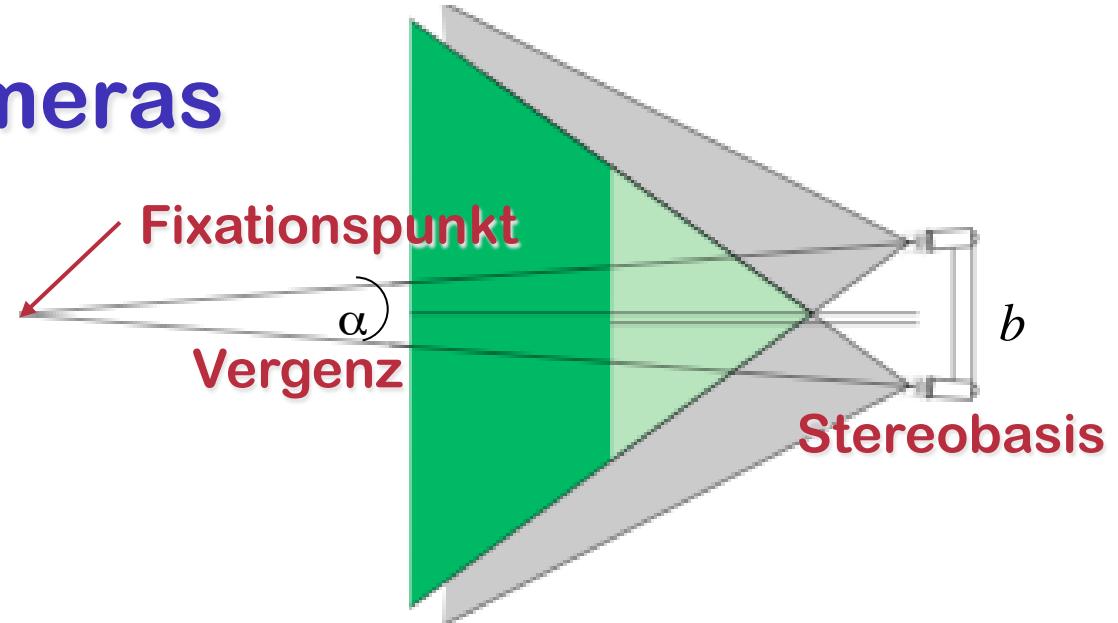
- Abbildung einer 360°-Rundumsicht durch Blick senkrecht in sphärischen, konischen, ... Rundumspiegel
- Vertikale bestimbar durch Variation der Spiegelgeometrie
- Entzerrung durch Rückrechnung möglich (aber oft nicht nötig)



Stereo-Kameras

Idee:

Errechne 3D-Szene aus Disparität korrespondierender Bildpunkte



Probleme

- korrespondierende Bildpunkte finden (oft *matching* unmöglich)
- Präzise Kalibrierung erforderlich für präzise 3D-Rekonstruktion (daher Problem mit *stereo from motion!*)
- Rechenaufwand beträchtlich



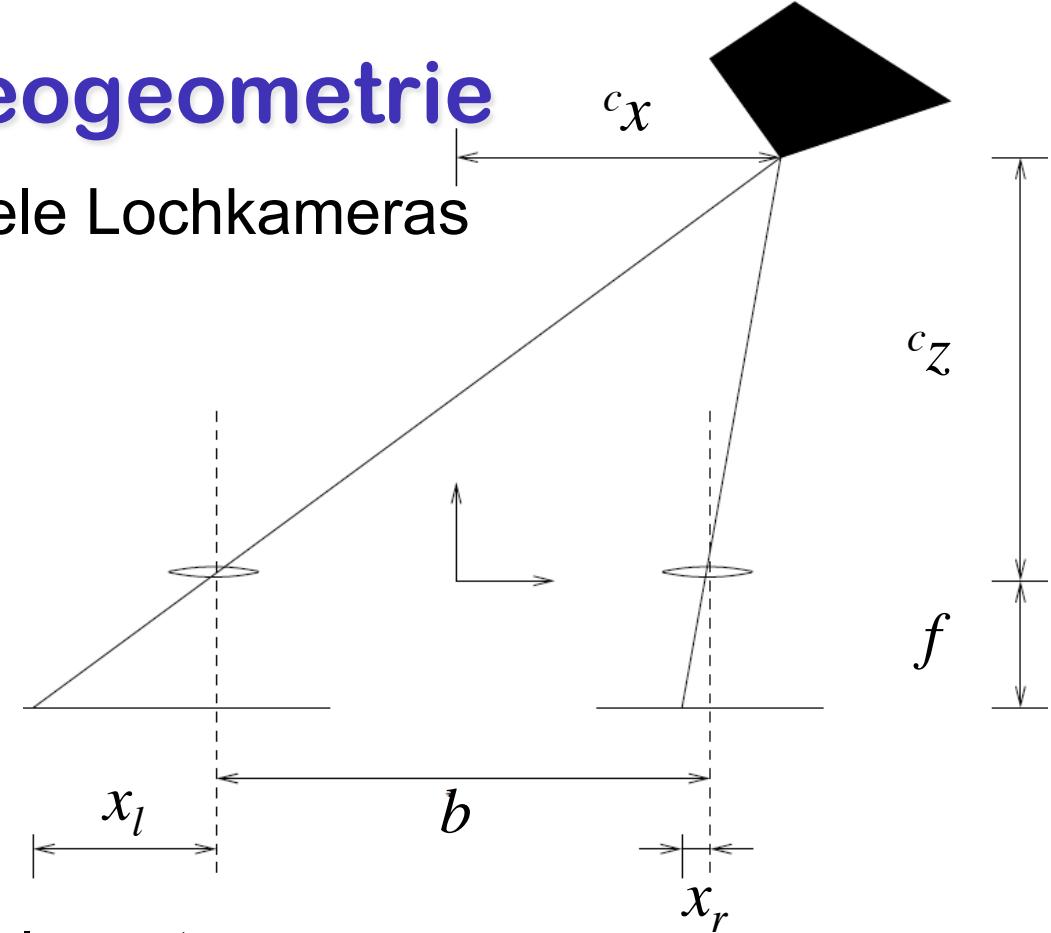
Kanonische Stereogeometrie

- idealisiertes Modell: Parallele Lochkameras
- Nach Strahlensatz:

$$\frac{^c z}{b} = \frac{f}{x_l - x_r}$$

- mit **Disparität**
 $d = x_l - x_r$: $^c z = \frac{b \cdot f}{d}$

- vgl. Triangulations-Entfernungsmeßung, Folie 49



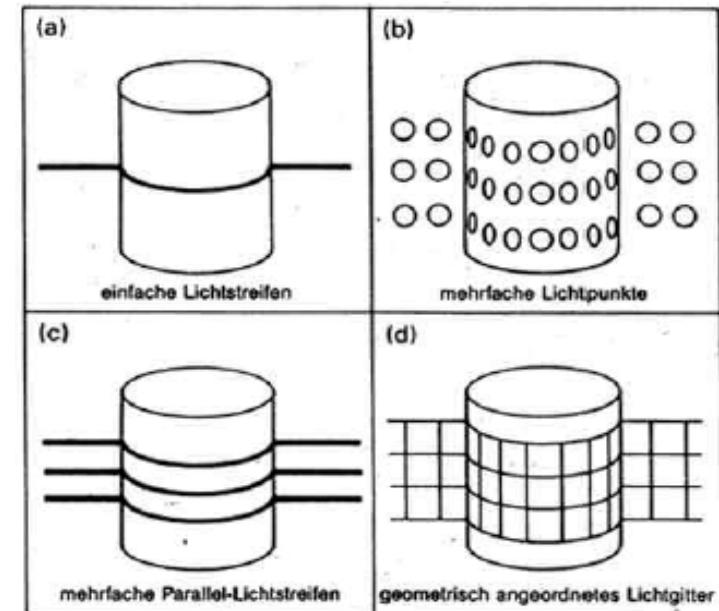
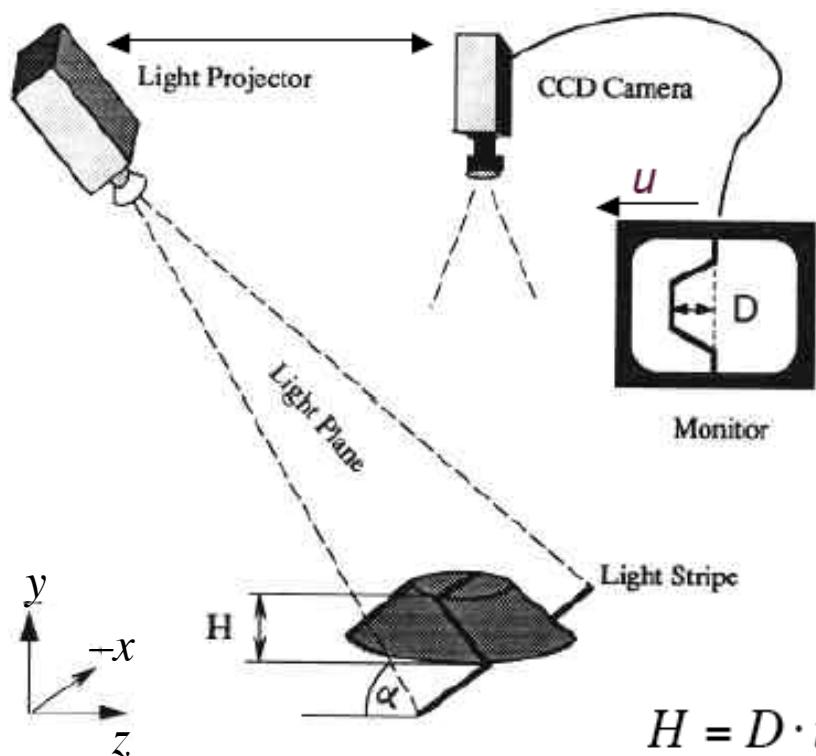
- Für $^c x$ und $^c y$ nochmal Strahlensatz:

$$\frac{^c x + \frac{b}{2}}{^c z} = \frac{x_l}{f} \Leftrightarrow ^c x = \frac{b}{2} \left(\frac{x_l + x_r}{d} \right)$$

- für $y_l = y = y_r$: $^c y = \frac{b \cdot y}{d}$

Strukturiertes Licht

- Projiziere bekanntes Muster in die Szene
- Nimm (mit Parallaxe!) die Szene auf
- Aus Deformation des Musters errechne Geometrie



Standardmethode in der
industriellen Bildverarbeitung

Ende Sensorik

- Ordentliche Roboterkontrollprogramme brauchen gute Sensordaten
- Selbst der beste Sensor liefert unperfekte Daten; clevere Algorithmen müssen damit umgehen
- Technische Sensoren gibt es in großer Vielfalt; sie werden immer weiter perfektioniert
- Unsere eigenen Arbeiten sind Laserscanner-lastig
- Bildverarbeitung gibt's an der UOS bei Gunther Heidemann (CogSci) und manchmal in Spezialvorlesungen in der AG Wissensbasierte Systeme