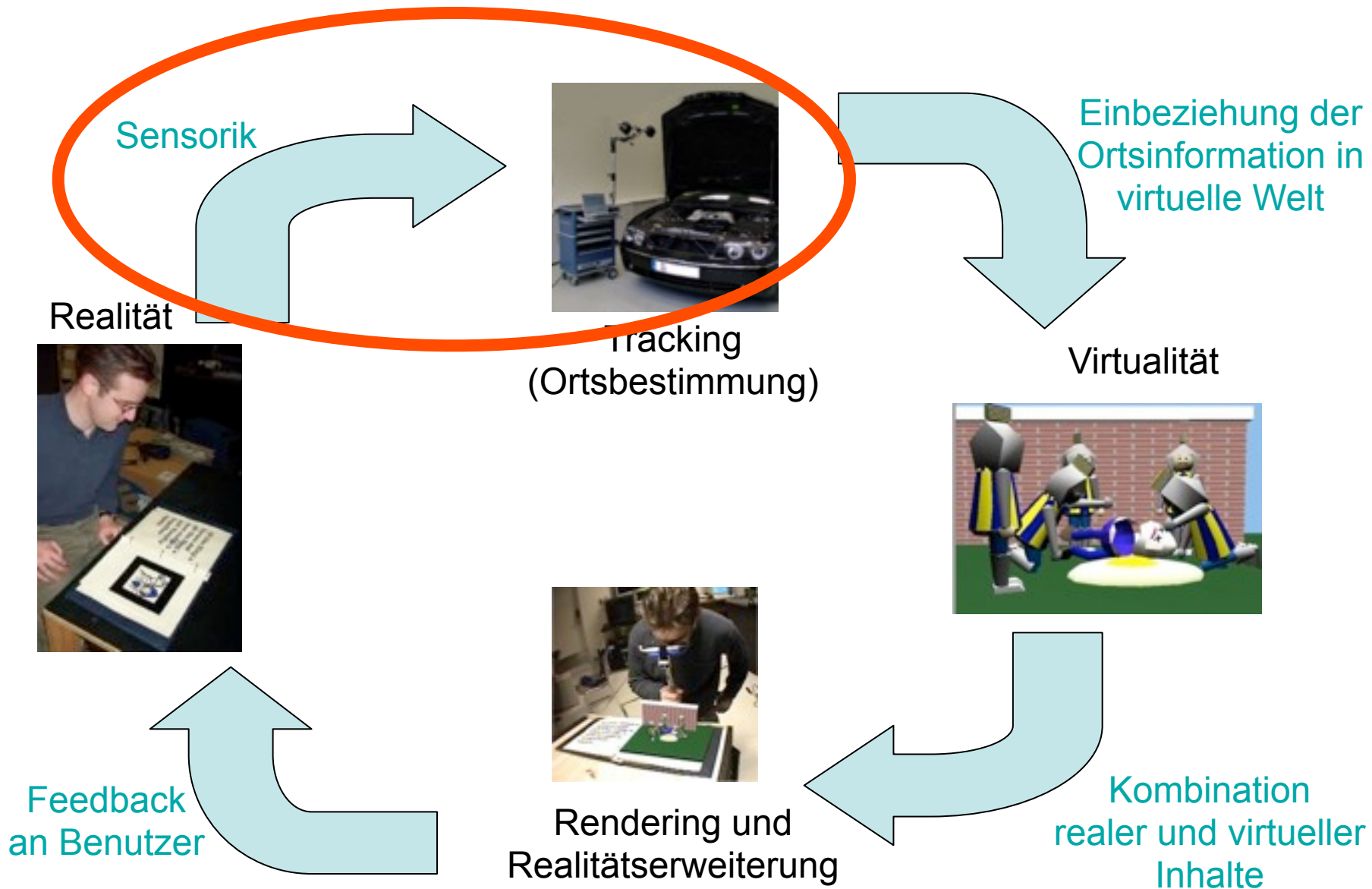


# Tracking, Teil 1: Einführung

Vorlesung „Augmented Reality“

Prof. Dr. Andreas Butz

# Ein Generisches AR-System



# Überblick

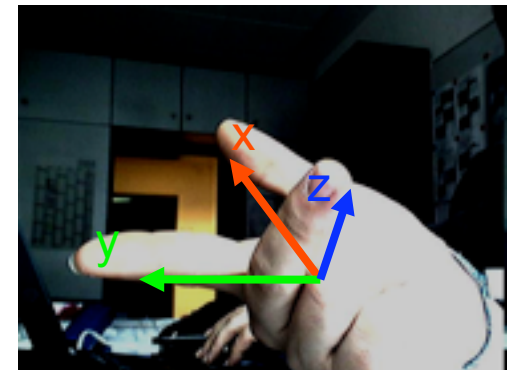
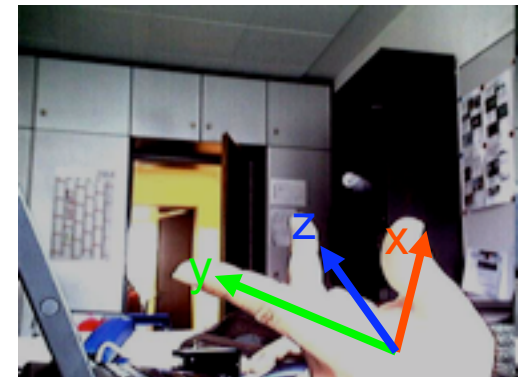
- Was ist Tracking?
- Koordinaten im 3D-Raum
- Transformationen im 2D-Raum
- Transformationen im 3D-Raum
- Darstellung von Rotationen im 3D-Raum
- Kombination von Transformationen
- Homogene Koordinaten
- Räumliche Beziehungen
- Mögliche Fehlerquellen beim Tracking

# Was ist Tracking?

- *Korrekte Registrierung* zwischen realen und virtuellen Objekten zwingend für AR
- *Kalibrierung* ist (offline-) Bestimmung aller unveränderlichen Parameter, z.B.
  - Position fester Sensoren
  - Feste Eigenschaften von Kameras (z.B. Brennweite)
- *Tracking* ist (online-) Bestimmung zeitlich variabler Parameter, meist Position von
  - Benutzern
  - Realen Objekten

# Koordinaten im 3D-Raum

- Kartesische Koordinaten
  - *Orthogonales* System
  - Koordinatenlinien sind Geraden mit konstantem Abstand
  - Achtung: *Einheit* beachten
- 2 Möglichkeiten (Obacht!):
  - *Linkshändiges* System
  - *Rechtshändiges* System
    - in Zukunft verwendet



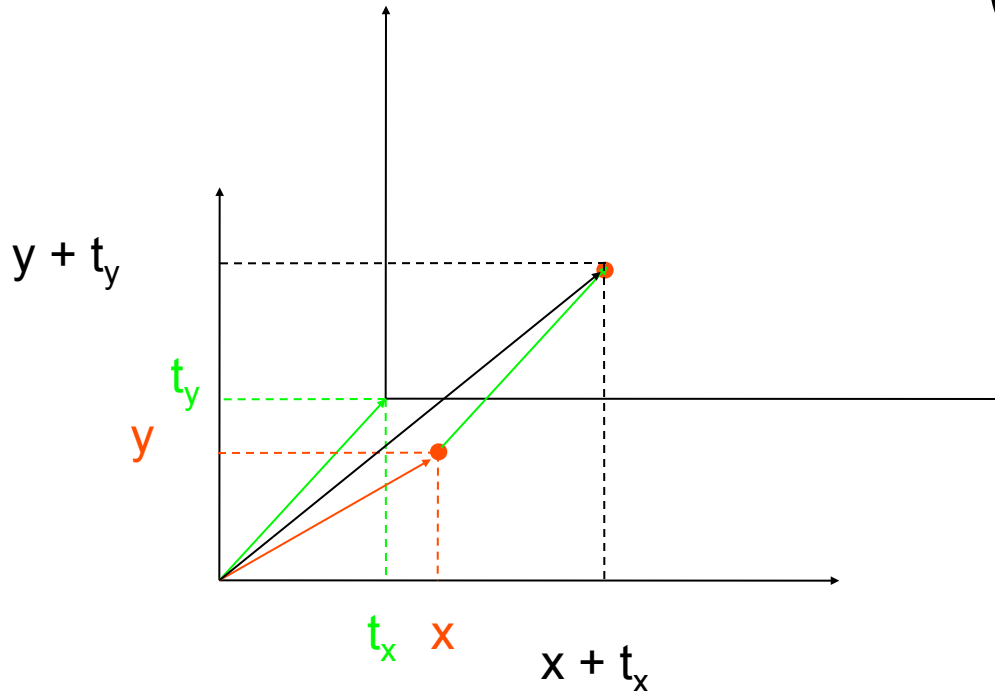
# Affine Transformationen

- Bewahrung von Parallelität
- Kein Bewahrung von Winkeln und Längen
- Arten:
  - Translation
  - Rotation
  - Skalierung
  - Scheren

# Transformationen im 2D-Raum (1)

- Punktdarstellung:  $p = \begin{pmatrix} x \\ y \end{pmatrix}$

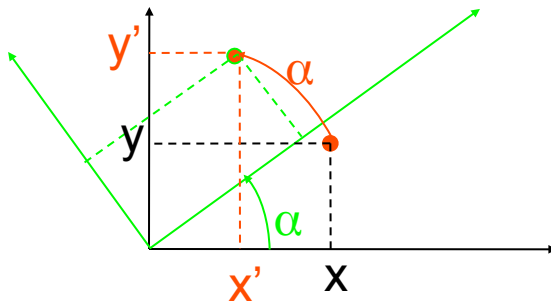
- Translation:  $p' = p + t = \begin{pmatrix} x \\ y \end{pmatrix} + \begin{pmatrix} t_x \\ t_y \end{pmatrix} = \begin{pmatrix} x + t_x \\ y + t_y \end{pmatrix}$



# Transformationen im 2D-Raum (2)

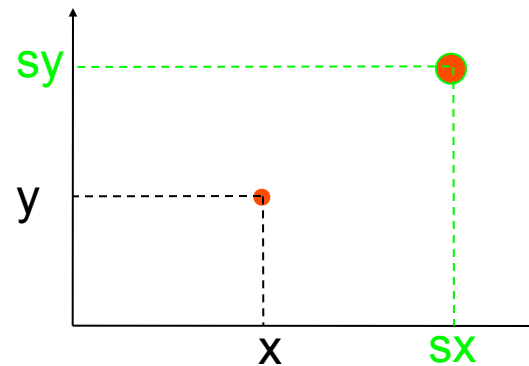
- Rotation:

$$p' = \begin{pmatrix} \cos \alpha \cdot x - \sin \alpha \cdot y \\ \sin \alpha \cdot x + \cos \alpha \cdot y \end{pmatrix} = \begin{pmatrix} \cos \alpha & -\sin \alpha \\ \sin \alpha & \cos \alpha \end{pmatrix} \cdot \begin{pmatrix} x \\ y \end{pmatrix} = \mathbf{R}_\alpha \cdot p$$



- Skalierung:

$$p' = s \cdot \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} sx \\ sy \end{pmatrix}$$





# Transformationen im 3D-Raum

- Punktdarstellung:

$$p = \begin{pmatrix} x \\ y \\ z \end{pmatrix}$$

- Translation:

$$p' = p + t = \begin{pmatrix} x \\ y \\ z \end{pmatrix} + \begin{pmatrix} t_x \\ t_y \\ t_z \end{pmatrix} = \begin{pmatrix} x + t_x \\ y + t_y \\ z + t_z \end{pmatrix}$$

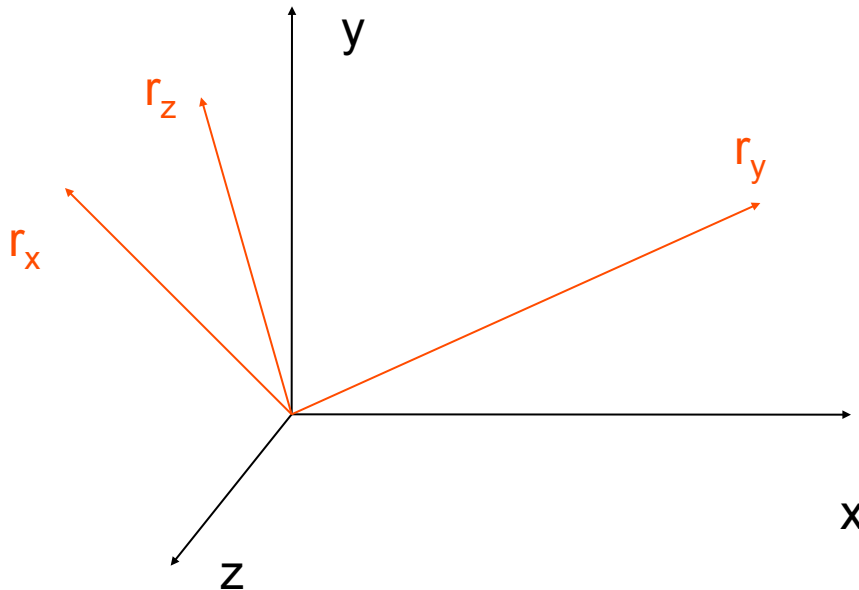
- Skalierung:

$$p' = s \cdot \begin{pmatrix} x \\ y \\ z \end{pmatrix} = \begin{pmatrix} sx \\ sy \\ sz \end{pmatrix}$$

# Rotationen im 3D-Raum (1)

- Matrixdarstellung:

$$p' = \mathbf{R} \cdot p = \begin{pmatrix} r_x^T \\ r_y^T \\ r_z^T \end{pmatrix} \cdot \begin{pmatrix} x \\ y \\ z \end{pmatrix} = \begin{pmatrix} r_x^1 & r_x^2 & r_x^3 \\ r_y^1 & r_y^2 & r_y^3 \\ r_z^1 & r_z^2 & r_z^3 \end{pmatrix} \cdot \begin{pmatrix} x \\ y \\ z \end{pmatrix} = \begin{pmatrix} r_x^1 x + r_x^2 y + r_x^3 z \\ r_y^1 x + r_y^2 y + r_y^3 z \\ r_z^1 x + r_z^2 y + r_z^3 z \end{pmatrix}$$



# Rotationen im 3D-Raum (2)

- Eigenschaften von Rotationsmatrizen:

- Orthonormal:

$$r_{i,j} \cdot r_{i,k} = \delta_{j,k}$$

- Also auch:

$$\mathbf{R}^{-1} = \mathbf{R}^T; \quad \mathbf{R}^T \mathbf{R} = \mathbf{I}$$

- Determinante +1, sonst (-1) *Rotoinversion*, d.h. Rotation gefolgt von Spiegelung (führt rechts- in linkshändiges System über)

- Alle orthonormalen 3x3 Matrizen mit Determinante +1 bilden die Gruppe  $SO(3)$

# Rotationen im 3D-Raum (3)

- Vorteile der Matrixdarstellung:
  - Recht intuitiv
  - Einfache, lineare Berechnungsvorschrift
  - Eindeutige Darstellung
- Probleme der Matrixdarstellung:
  - Matrix muss orthonormal sein (problematisch bei numerischen Instabilitäten) → hohe Redundanz in Darstellung, 9 Parameter für 3 Freiheitsgrade
  - Keine einfache Interpolation von Rotationen

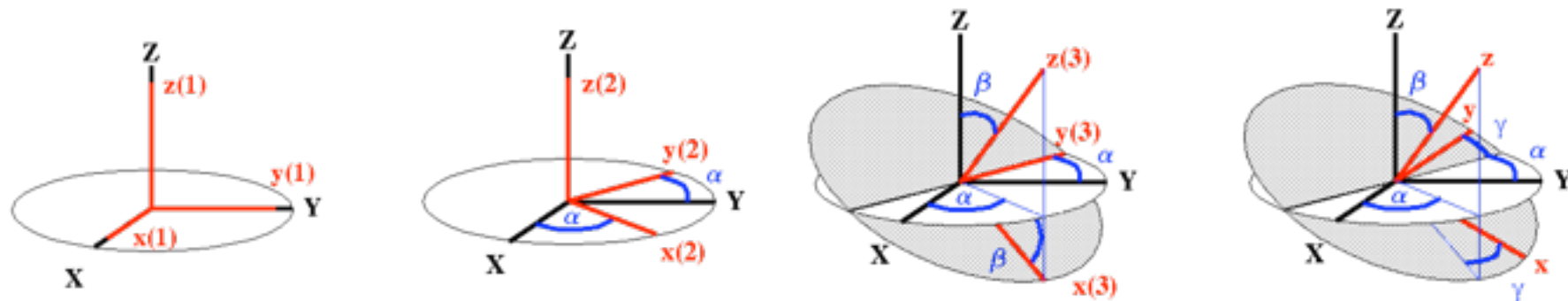
# Rotationen im 3D-Raum (4)

- Rotationsmatrizen:

$$\mathbf{R}_\alpha, \mathbf{R}_\beta, \mathbf{R}_\gamma = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos\alpha & -\sin\alpha \\ 0 & \sin\alpha & \cos\alpha \end{pmatrix} \begin{pmatrix} \cos\beta & 0 & -\sin\beta \\ 0 & 1 & 0 \\ \sin\beta & 0 & \cos\beta \end{pmatrix} \begin{pmatrix} \cos\gamma & -\sin\gamma & 0 \\ \sin\gamma & \cos\gamma & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

# Rotationen im 3D-Raum (5)

- Eulerwinkel:
  - Jede Rotation kann als Folge dreier Rotationen um drei Koordinatenachsen ausgedrückt werden
  - Häufigste Darstellung:  
 $\alpha$  um z-Achse, dann  $\beta$  um *neue* y-Achse, dann  $\gamma$  um *ganz neue* z-Achse

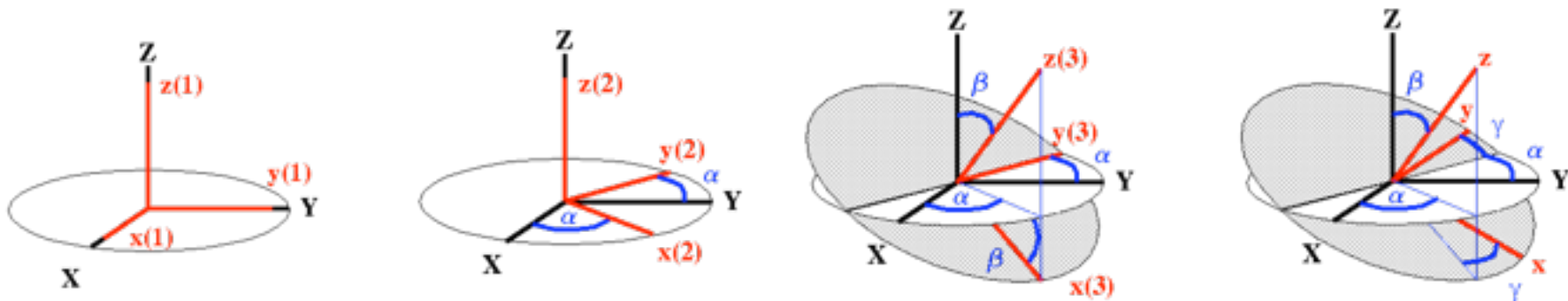


# Rotationen im 3D-Raum (6)

- Eulerwinkel in Matrixdarstellung:

$$\mathbf{R} = \mathbf{R}_\gamma \cdot \mathbf{R}_\beta \cdot \mathbf{R}_\alpha = \begin{pmatrix} \cos \alpha & \sin \alpha & 0 \\ -\sin \alpha & \cos \alpha & 0 \\ 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} \cos \beta & 0 & -\sin \beta \\ 0 & 1 & 0 \\ \sin \beta & 0 & \cos \beta \end{pmatrix} \cdot \begin{pmatrix} \cos \gamma & \sin \gamma & 0 \\ -\sin \gamma & \cos \gamma & 0 \\ 0 & 0 & 1 \end{pmatrix} =$$

$$= \begin{pmatrix} \cos \alpha \cos \beta \cos \gamma - \sin \alpha \sin \gamma & \sin \alpha \cos \beta \cos \gamma + \cos \alpha \sin \gamma & -\sin \beta \cos \gamma \\ -\cos \alpha \cos \beta \sin \gamma - \sin \alpha \cos \gamma & -\sin \alpha \cos \beta \sin \gamma + \cos \alpha \cos \gamma & \sin \beta \sin \gamma \\ \cos \alpha \sin \beta & \sin \alpha \sin \beta & \cos \beta \end{pmatrix}$$



# Rotationen im 3D-Raum (7)

- Einziger Vorteil der Eulerwinkel:
  - Häufig verwendet, v.a. in traditionellen Trackinganwendungen der Luftfahrt („Roll/Pitch/Yaw“ – Rollen/Neigen/Gieren)
- Nachteile der Eulerwinkel:
  - Darstellung alles andere als eindeutig (Welche Achsenfolge? Neue/alte Achsen als Bezug?)
  - Nur auf den ersten Blick intuitiv
  - Möglicher Verlust eines Freiheitsgrades, wenn nach einer Rotation eine „neue“ auf eine „alte“ Achse fällt (*Gimbal lock*, von „Gimbal“ = Kardanring bei Gyroskopen)
  - Teure Berechnung: Sinus/Kosinus



# Rotationen im 3D-Raum (8)

- Axis/Angle Darstellung:
  - Jede Rotation kann als Drehung um eine *bestimmte Achse  $x$*  und einen *bestimmten Winkel  $\varphi$*  dargestellt werden (Eulers Theorem)
  - Da  $\mathbf{R}x = x$ , ist  $x$  Eigenvektor zum Eigenwert 1 der äquivalenten Rotationsmatrix  $\mathbf{R}$  (1 ist einziger realer Eigenwert von  $\mathbf{R}$ ).
- Vorteile:
  - Intuitiv, weitgehend eindeutig
  - Geringe Redundanz (4 Parameter, 3 Freiheitsgrade, mit Normierung von  $x$  reichen 3 Parameter)
- Problem:
  - Kombination mehrerer Rotationen schwierig

# Rotationen im 3D-Raum (9)

- Quaternionen (Hamilton, 1843):
  - Hyperkomplexe Zahlen vom Rang 4, bestehend aus Skalar  $q_0$  und Vektor  $\mathbf{q}$
  - Mit Einheitsquaternionen können Rotationen im 3D ausgedrückt werden
  - Berechnung aus Axis/Angle  $\mathbf{x}/\varphi$ :

$$q_0 = \cos \frac{\varphi}{2}; \quad \mathbf{q} = \sin \frac{\varphi}{2} \mathbf{x}$$

- Vor-/Nachteile:
  - wie Axis/Angle, aber jetzt schnelle Kombination von Rotationen durch *Quaternionenmultiplikation*
  - Interpolation leicht möglich (→ Übung)

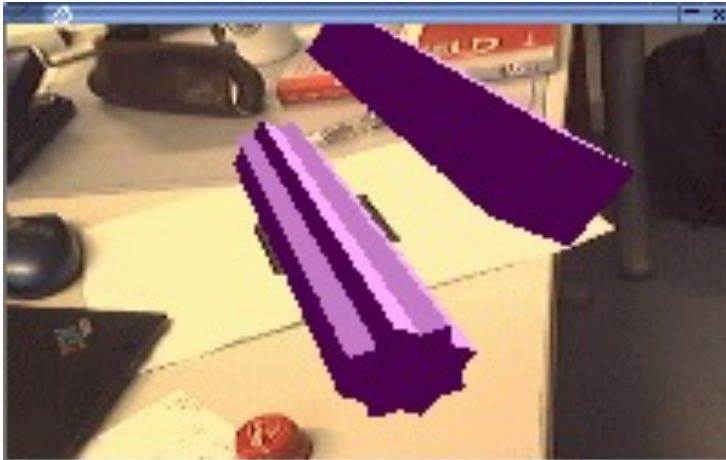
# Rotationen in 3D: Zusammenfassung

- Verschiedene Darstellungsmöglichkeiten
- Orthonormale Rotationsmatrizen und Quaternionen vorteilhaft
  - Eindeutige Darstellung
  - Leichte/effiziente Kombination mehrerer Rotationen
- Vorsicht bei Eulerwinkeln!

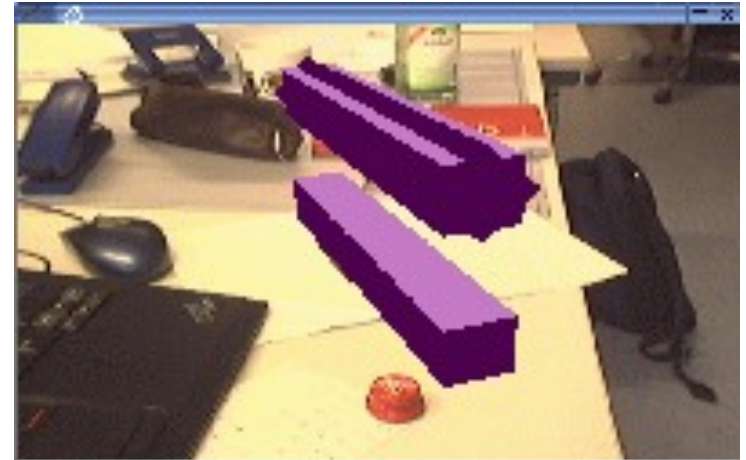
# Kombination von Transformationen

- Eigentlich kein Problem, aber...  
die Reihenfolge macht den Unterschied:

Zuerst Rotation, dann Translation:



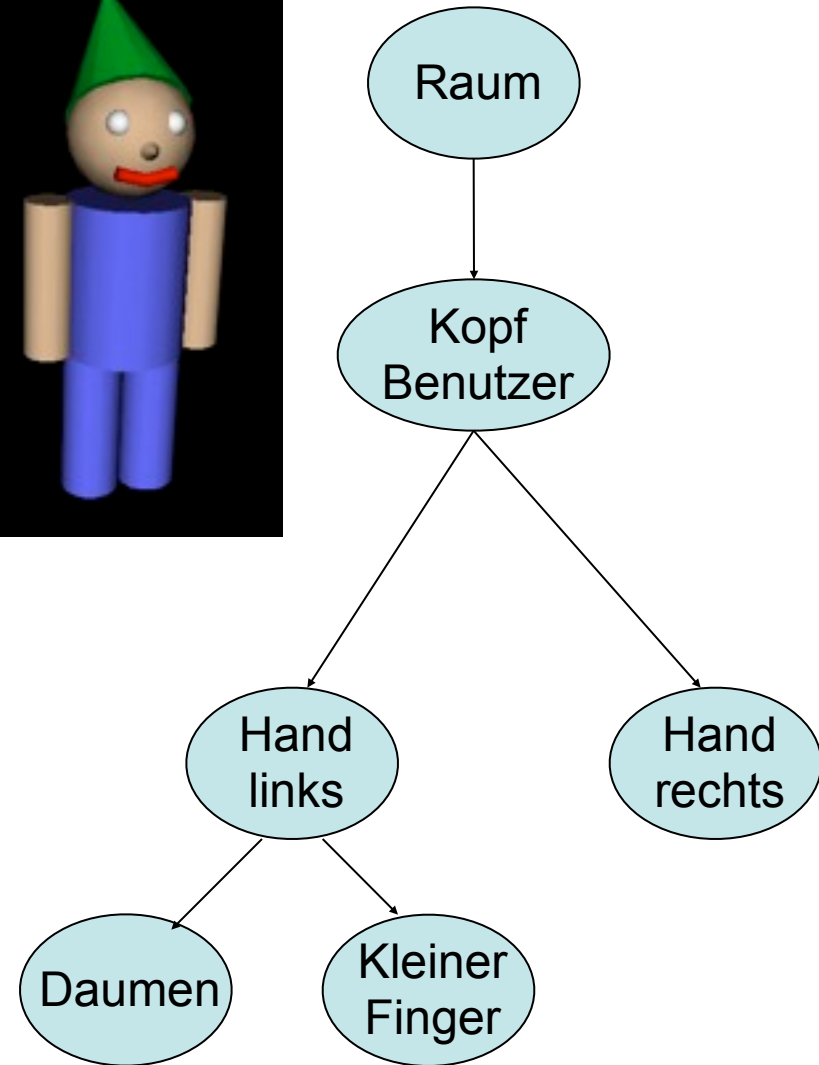
Zuerst Translation, dann Rotation:



- Inversion von Transformationen: Wechsel des Bezugssystems

# Szenengraph

- Hierarchische Kombination von Transformationen (Konzept aus Computergraphik)
- Beispiel: Benutzer ist in einem Raum, Kamera auf seinem Kopf wird getrackt. Aus dem Kamerabild wird zudem die Position der Hände des Benutzers bestimmt.
- Hauptvorteil: Gruppierung von Objekten



# Homogene Transformationsmatrizen (1)

- Wie kann man Kombination aus Translationen, Rotationen und Skalierungen als einzige Transformation darstellen?
- Rotationen und Skalierungen:
  - Darstellung als Matrix
  - Kombination: Produkt mehrerer Matrizen
- Translation:
  - Darstellung als Vektor
  - Kombination mehrerer Translationen: Summe der Vektoren
  - Kombination mit Rotations-/Skalierungsmatrix ??

# Homogene Transformationsmatrizen (2)

- Durch 4. Komponente („homogene Vektoren“) können auch Translationen als Matrix dargestellt werden:

$$p = \begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix}; \quad \mathbf{H}_t = \begin{pmatrix} 1 & 0 & 0 & t_x \\ 0 & 1 & 0 & t_y \\ 0 & 0 & 1 & t_z \\ 0 & 0 & 0 & 1 \end{pmatrix}; \quad \mathbf{H}_t \cdot p = \begin{pmatrix} x + t_x \\ x + t_y \\ x + t_z \\ 1 \end{pmatrix}$$

- Kombination mit Rotationen/Skalierungen durch einfaches Aufmultiplizieren:

$$\mathbf{H}_r = \begin{pmatrix} r_x^1 & r_y^1 & r_z^1 & 0 \\ r_x^2 & r_y^2 & r_z^2 & 0 \\ r_x^3 & r_y^3 & r_z^3 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}; \quad \mathbf{H}_s = \begin{pmatrix} s_x & 0 & 0 & 0 \\ 0 & s_y & 0 & 0 \\ 0 & 0 & s_z & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

# Homogene Transformationsmatrizen (3)

- Homogene Matrizen sind Standard in der Computergraphik (z.B. OpenGL)
- Weitere Berechnungen möglich:
  - Projektionen in 2D (perspektivisch/orthogonal)
  - Ebenen in 3D und Schnitte



# Räumliche Beziehungen

- Zurück zum Tracking:  
Bestimmung räumlicher Beziehungen
- Klassifikation:
  - Absolute vs. relative Ortsbestimmung
  - Ableitungen: Geschwindigkeit, Beschleunigung
- Am wichtigsten & häufigsten:  
Absolute Position und Orientierung im 3D-Raum

# Anforderungen an AR-Tracker

- Hohe Genauigkeit
- Geringe Latenzzeit
- Hohe Wiederholrate (min. 10 fps)
- Kleine Baugröße, v.a. von mobilen Bestandteilen
- Meist 6 DOF, absolute Messung
- Simultane Unterstützung mehrerer Objekte/  
Benutzer
- Niedriger Preis

→ Gesucht: eierlegende Tracking-Wollmilchsau.

# Mögliche Fehlerquellen (1)

- Latenz ist DIE Fehlerquelle
  - Jede Millisekunde führt max. zu 1mm, im Schnitt zu 1/3mm Registrierungsfehler
  - D.h.: ein System, das eine Genauigkeit von 1mm haben soll, darf eine *Gesamtlatenzzeit* von 1ms nicht überschreiten
- Ursachen der Latenz
  - Tracker
  - Bearbeitungszeit auf dem Host
  - Bildgenerierung
  - Diverse Displayeigenschaften

# Mögliche Fehlerquellen (2)

- Numerische Fehler bei der Akkumulation von Koordinatensystemen
- Optische Fehler: Kissenverzerrung, fehlendes Augentracking
- Trackermessfehler:
  - Statische Fehler (unveränderlich, können durch gute Kalibrierung behoben werden)
  - Jitter (Rauschen, kann nicht behoben werden)
  - Dynamische Fehler (abhängig von der Bewegung des Sensors und/oder getrackten Objekts)

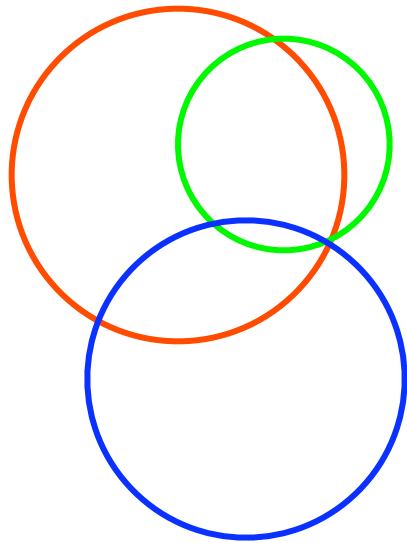
# Tracking, Teil 2: Methoden und Systeme

Vorlesung „Augmented Reality“  
Prof. Dr. Andreas Butz

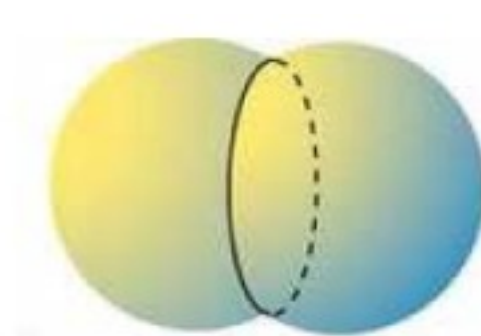
# Überblick

- Akustisches Tracking
- GPS etc.
- Inertialtracking
- Magnetisches Tracking
- Optisches Tracking (basierend auf Bildverstehen)
- Sensorfusion

# Akustisches Tracking: Prinzip

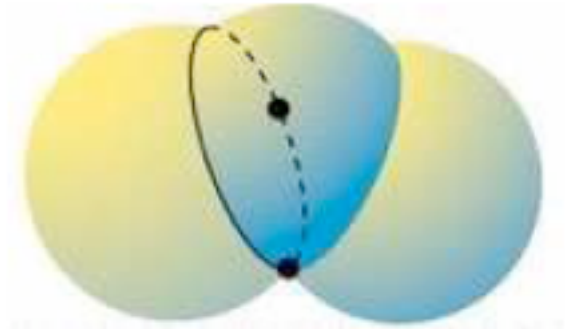


2D:  
3 Kreise bestimmen  
eindeutigen Punkt



[Bishop et al. 2001]

3D:  
2 Kugeln bestimmen  
eindeutigen Kreis



[Bishop et al. 2001]

3D:  
3 Kugeln bestimmen zwei  
Punkte  
→ Ausschluss eines Punkts  
durch geometrische  
Einschränkungen

# Akustisches Tracking: Rekonstruktion (1)

- 3 Kugeln mit Radius  $r_i$  und Zentrum  $(x_i, y_i, z_i)$

$$(x - x_0)^2 + (y - y_0)^2 + (z - z_0)^2 = r_0^2$$

$$(x - x_1)^2 + (y - y_1)^2 + (z - z_1)^2 = r_1^2$$

$$(x - x_2)^2 + (y - y_2)^2 + (z - z_2)^2 = r_2^2$$

- Trick: Koordinatensystem verschieben

- Kugel 0 im Ursprung  $(0,0,0)$

- Kugel 1 an Position  $(1,0,0)$

- Kugel 2 an Position  $(0,1,0)$

→ „Kugelsystem“ kann in Labor-/Zielsystem zurücktransformiert werden



# Akustisches Tracking: Rekonstruktion (2)

- Vereinfachte Gleichungen:

$$x^2 + y^2 + z^2 = r_0^2$$

$$(x-1)^2 + y^2 + z^2 = r_1^2$$

$$x^2 + (y-1)^2 + z^2 = r_2^2$$

- Lösung:

$$x = \frac{r_0^2 - r_1^2 + 1}{2}$$

$$y = \frac{r_0^2 - r_2^2 + 1}{2}$$

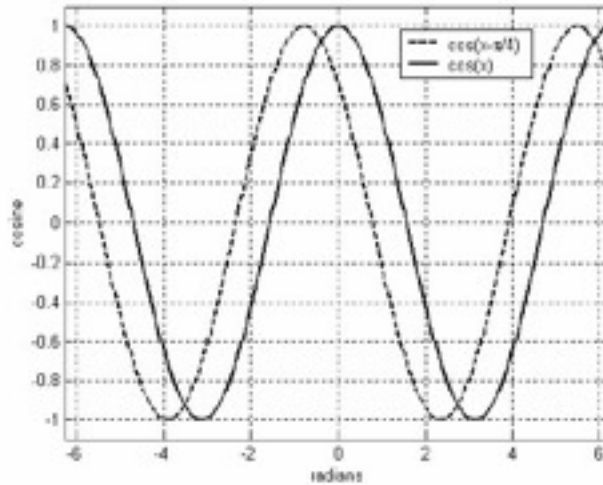
$$z = \pm \sqrt{r_0^2 - x^2 - y^2}$$

# Akustisches Tracking: Laufzeitmessung

- Positionsabschätzung durch Kenntnis der Schallgeschwindigkeit:  $d[m] = v[m/s] \times t[s]$
- Triggern des Senders nötig (per Funk oder Lichtsignalen)
- Schallgeschwindigkeit hängt ab von
  - Temperatur
  - Luftdruck
  - (Medium !!)
- Störanfällig durch Reflektionen
- Multi-Objektbetrieb: verschiedene Frequenzen und/oder Round-Robin-Verfahren

# Akustisches Tracking: Phasenkohärenz

- Phasenverschiebung durch Laufzeit



[Bishop et al. 2001]

- Problem: Verschiebung  $\Phi$  nicht unterscheidbar von  $(\Phi + 2n\pi)$ , dadurch max. Verschiebung zwischen zwei Messungen z.B. bei 40 KHz:

$$\delta [m] = \frac{c [m/s]}{f [Hz]} = \frac{331 [m/s]}{40 [kHz]} = 8,275 [mm]$$

# Akustisches Tracking: Systeme



[Sutherland 1968]



Intersense IS-900  
[Intersense Inc. 2005]

# Akustisches Tracking: Diskussion

## Vorteile:

- Recht einfache Berechenbarkeit
- Kleine Bauform

## Nachteile:

- Sichtverbindung („Line of Sight“) zum Sender nötig
- Fehlerakkumulation bei Phasenkohärenzverfahren
- Abhängig von variablen Umweltfaktoren (Temperatur, Druck, Feuchtigkeit)
- Nur Positionsbestimmung

# GPS: Geschichte

ca. 1940: **LORAN** (Long Range Aid to Navigation) als erstes System zur Schiffsnavigation durch Laufzeitmessung von Radiosignalen (Nachfolger noch heute im Betrieb)

1960: **Transit** (US Navy) als erstes satellitengestütztes Navigationssystem v.a. für U-Boote

1967: **Timation** (US Navy) als erstes System mit Atomuhren auf Satelliten

1972: **System 612B** (US Air Force) nutzt Pseudozufallssignale zur 3D-Lokalisierung

1974-1979: **GPS NAVSTAR** (US Air Force) kombiniert Ideen, Testphase

1980-1994: Aufbauphase, erster „erfolgreicher Test“ im ersten Golfkrieg

Seit 1982: **GLONASS** (**GLO**bal'naya **NA**vigatsionannaya **S**putnikovaya **S**istema) als sowjetische Variante von GPS

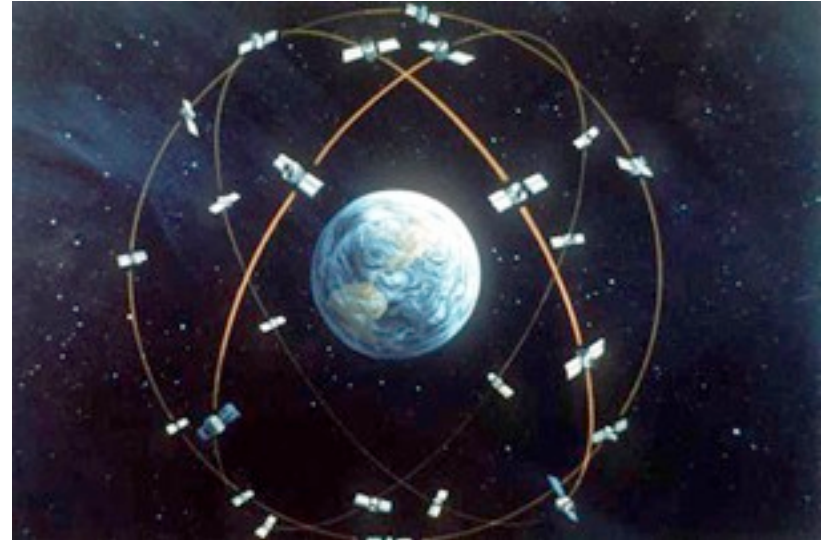
ca. 1985: Erste kommerzielle Anwendungen

2000: Aufhebung der künstlichen Störung von GPS für zivile Empfänger

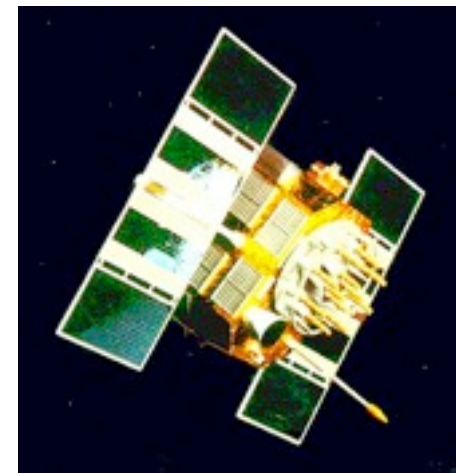
2004: EU-Beschluss zum Bau von GALILEO

# GPS: Grundprinzip

- 24 Satelliten auf 6 Ebenen in 20km Höhe
- Empfänger bekommt Pseudo-Zufallssignale
  - Zeitstempel
  - Position des Satelliten
- Laufzeit der Signale als Basis für Trilateration
- Genauigkeit:  $> 10\text{m}$



[US DoD]



[NASA]

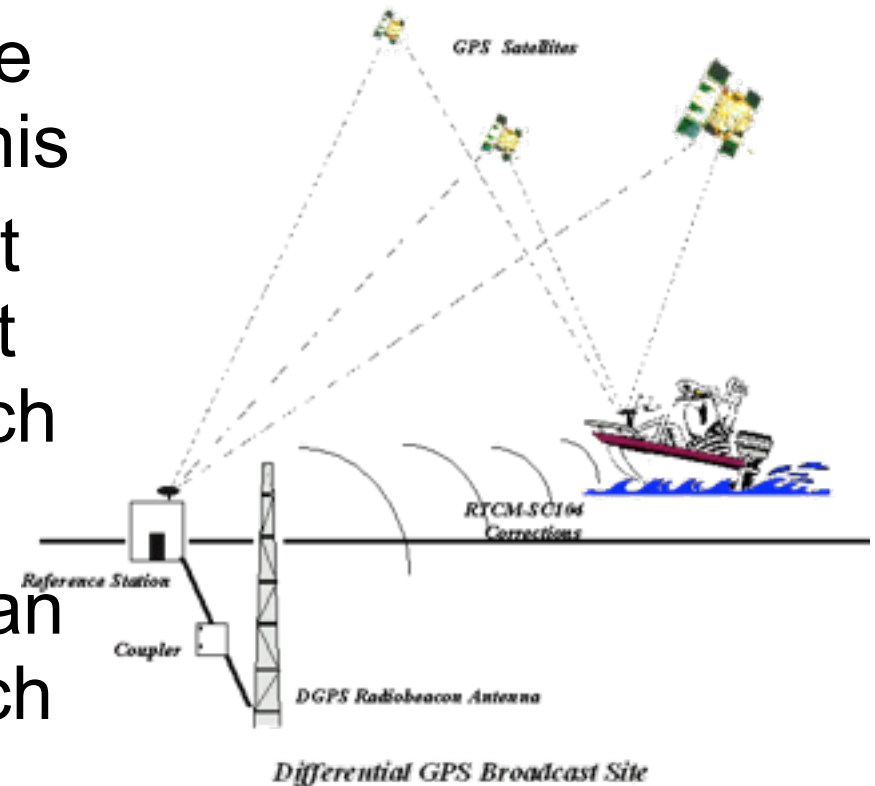
# GPS: Zeitprobleme

- Satelliten haben (hochgenaue) Atomuhren
- Empfänger hat (billige) Quarzuhr
- Drift  $d$  der Empfängeruhr als zusätzliche Unbekannte in Positionsgleichung
- Somit:
  - 4 Unbekannte ( $X, Y, Z, d$ )
  - 4 Laufzeitmessungen
  - 4 Satelliten nötig
- Wahl der Satelliten (bei mehr als 4) ist Optimierungsproblem



# Differential GPS (DGPS)

- Störungen in Atmosphäre verfälschen GPS-Ergebnis
- Idee: Referenzstation mit bekanntem Ort empfängt Signale und kann dadurch Störungen bestimmen
- Übermittlung der Werte an (mobile) Empfänger durch UKW-Funk oder GSM/UMTS
- Genauigkeit: 1-5m



# Realtime Kinematic DGPS

- Idee: Bestimmung Phasenverschiebung
- Sowohl Referenzempfänger als auch mobiler Empfänger bestimmen Phase des Trägersignals
- Sehr aufwendig und teuer
- Früher v.a. offline (Vermessungsaufgaben)
- Genauigkeit bis auf Zentimeter
- Hohe Wiederholrate (20 Hz)

# Referenzsignale für (RTK) DGPS

- Eigene Referenzstation
  - Sehr teuer
  - Beschränkte Reichweite (z.B. 2km um Referenzstation)
- Gemeinsame Referenzstationen
  - Abodienste
  - Zusätzliche Möglichkeiten

# Landesvermessungsämter: SAPOS

- In Bayern: 36 Referenzstationen
- Vernetzung der Referenzstationen (auch anderer Bundesländer), dadurch geringere Anzahl von Referenzstationen nötig

<http://www.sapos.de/>



[LVA Bayern 2005]

# Dienste & Korrektursignalverbreitung

- EPS: Echtzeitpositionierungsservice
  - DGPS-Korrekturdaten (1-3m Genauigkeit)
  - Verbreitung über Langwelle und UKW/RDS
  - Pauschale Gebühr 150,- €
- HEPS: Hochpräziser Echtzeitpositionierungsservice
  - RTK-DGPS-Korrekturdaten (1-5cm Genauigkeit)
  - Empfang über GSM oder 2m-Funk
  - Gebühr: 0,10 € / Minute (Bayern)
- GPPS/GHPS: Geodätisch-präziser Positionierungsservice/ Geodätisch-hochpräziser Positionierungsservice
  - Postprocessing, Datenübermittlung im Internet
  - Genauigkeit bis auf mm
  - Gebühr: 0,20 – 0,80 € / Minute (Bayern)

# GPS: Produkte



Trimble DGPS RTK Reference Station



Trimble DGPS RTK Receiver Board



Garmin eTrex



Garmin geko 101,  
4,8 cm x 9,9 cm x 2,4 cm  
88g

PenmapGPS-  
RTK Receiver /  
w Tablet PC



# GPS: Diskussion

## Vorteile:

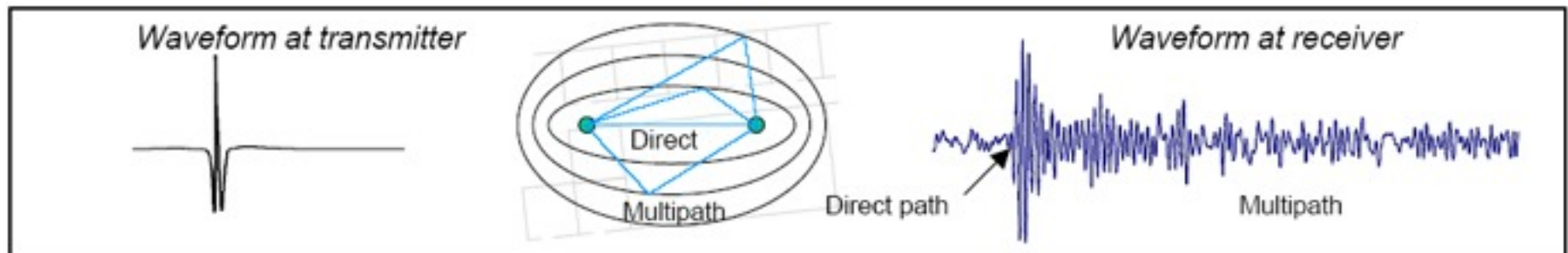
- Weltweit verfügbar
- Für Anwendungen im Freien ausreichend genau (zumindest bei DGPS)
- Günstige Empfänger (ab 100 €)

## Nachteile:

- Sichtverbindung zu Satelliten nötig
- Nur Bestimmung der Position, keine Orientierung
- Meist geringe Updaterate (ca. 1 Hz)
- Höhere Genauigkeit unverhältnismäßig teuer

# Ultra Wideband (UWB)

- GPS geht nur mit freier Sicht zu Satelliten
- Konventionelle RF-Technologien mit Signalstärkenmessung und Triangulation sind stark störungsanfällig
  - Multipathprobleme
  - Dämpfung durch Umgebung beeinflusst
- UWB sendet keine Sinuswellen, sondern kurze codierte Impulse (in der Fourierzerlegung tauchen also alle Frequenzen auf)
- Genauigkeit 10 cm; Reichweite 600 m<sup>2</sup> (bei 4 Sensoren)



[Shreve & Kell, 2001]



# Inertialtracking: Grundlagen

- Grundprinzip: Trägheit von Massen
- Sensoren sind unabhängig von Infrastruktur
- Aber: nur relative Messungen möglich

# Inertialtracking: Accelerometer

- Messung der Kraft, die durch eine Beschleunigung auf eine Masse wirkt:

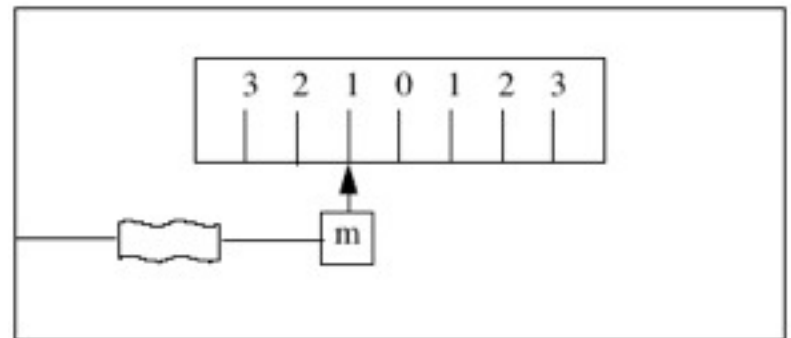
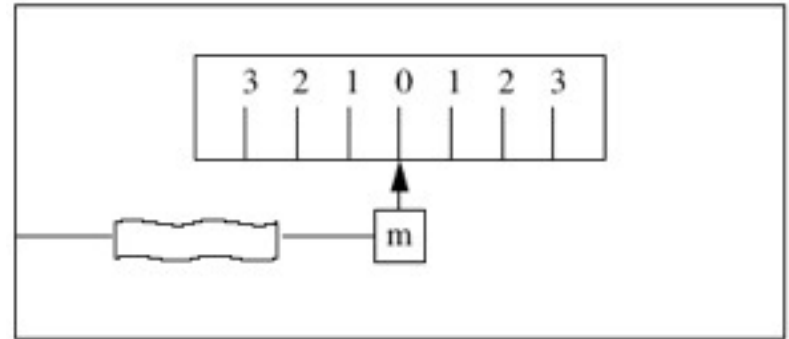
$$F = ma$$

- Masse wird an Feder gehängt, Messung von Stauchung/Streckung durch Potentiometer

- Bestimmung der absoluten Positionsänderung durch

Integration: 
$$p = \iint a dt^2$$

- Sehr günstig
- Problem: Fehlerakkumulation

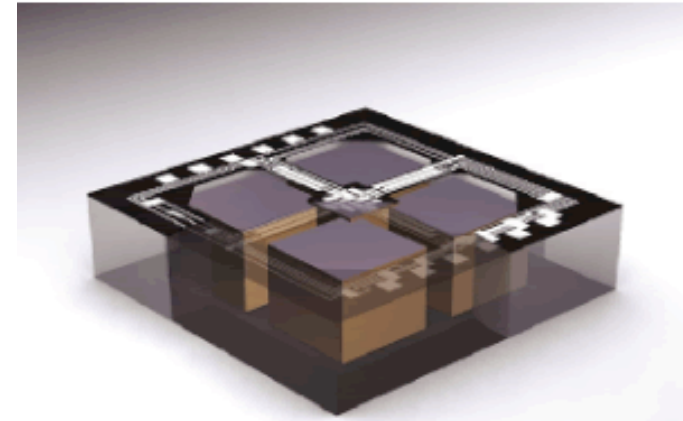


[Bishop et al. 2001]

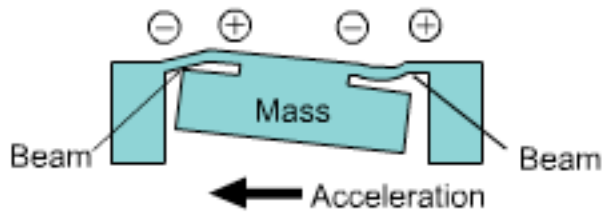
# Acceleration sensor

- Built from piezo elements and weights
- Integrated circuit

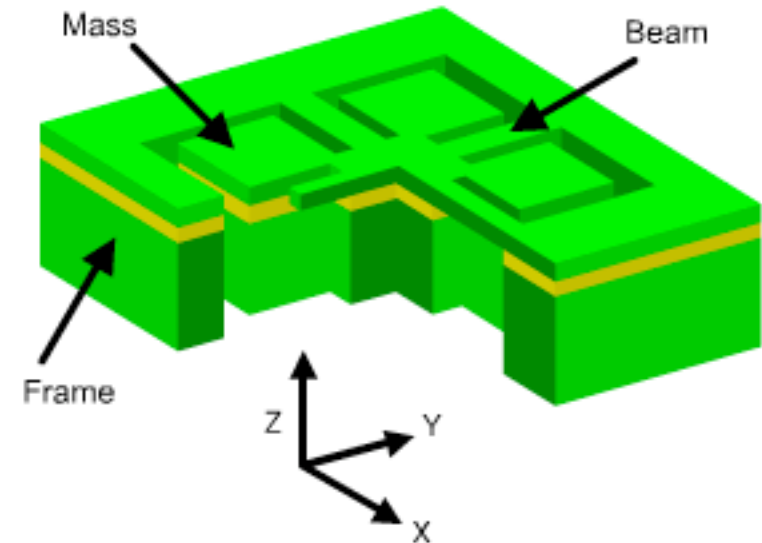
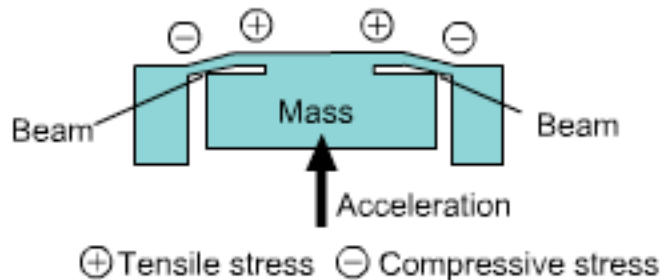
Figure 1 MEMS sensor chip in the GS3



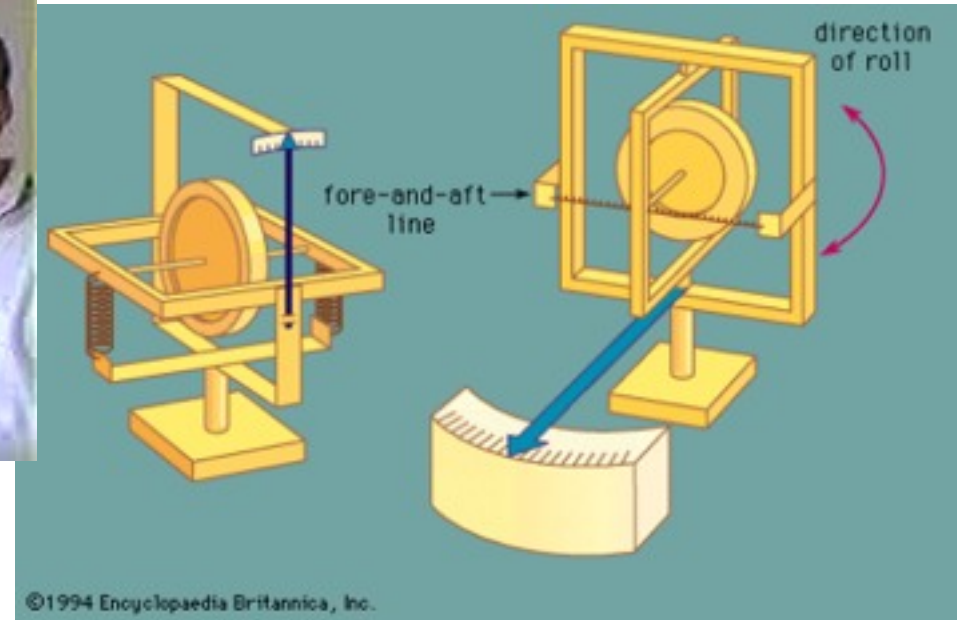
■ Detection of acceleration in the X-axis (Y-axis) direction



■ Detection of acceleration in the Z-axis direction



# Inertialtracking: Gyroskop

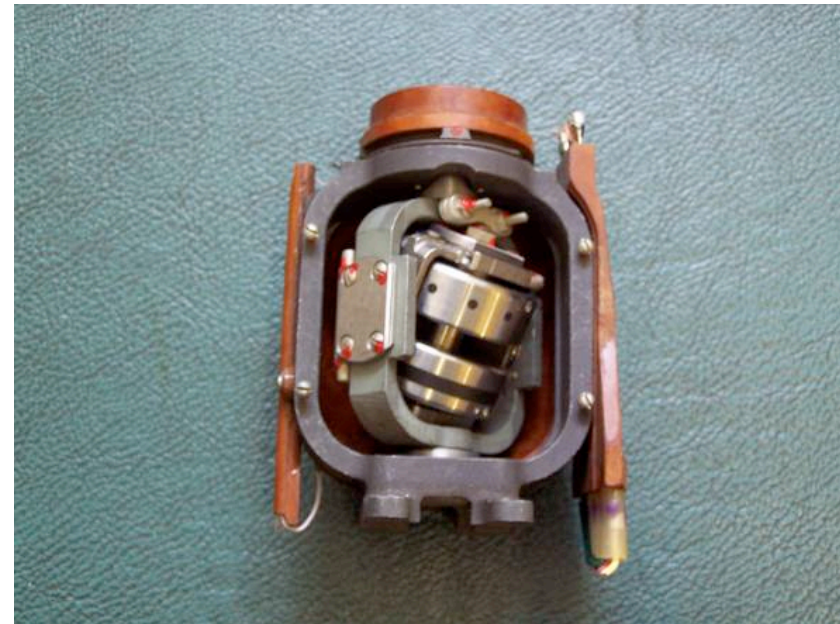
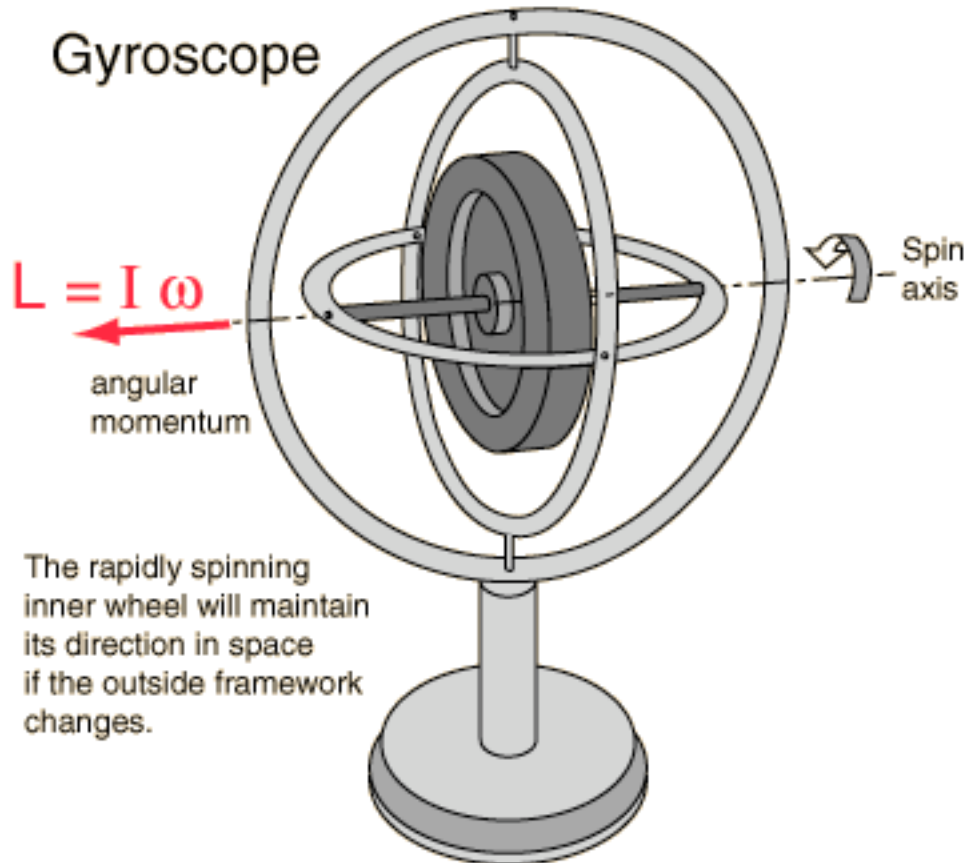


- Grundprinzip: Bewahrung des Winkelmoments drehender Körper, dadurch Messung der Winkelgeschwindigkeit
- Einfachste Art: Messung der Präzessionskraft
- Einfache Integration zur Berechnung absoluter Orientierung

# Gyroscopes

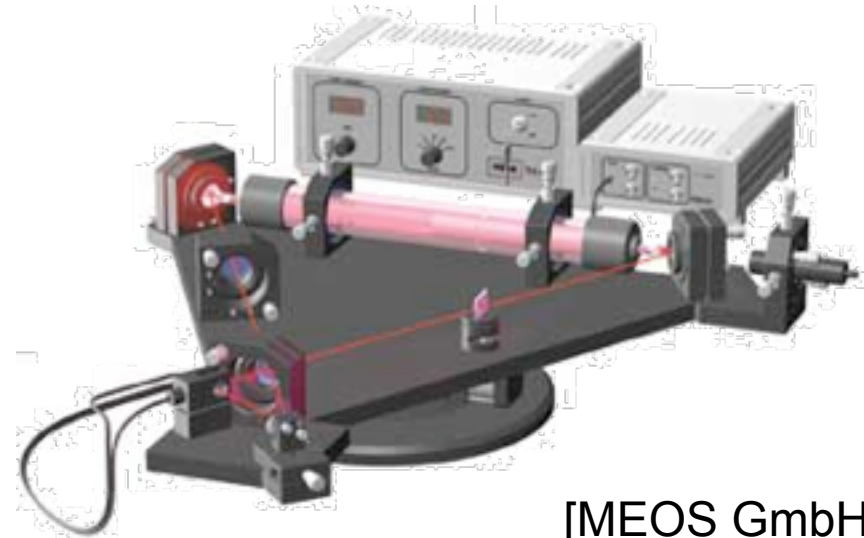
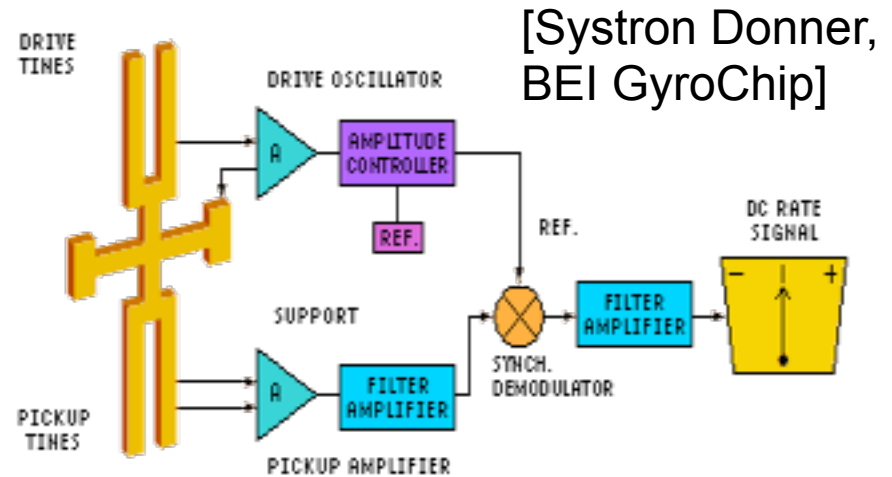


Gyroscope



# Inertialtracking: Weitere Gyroskope

- Vibrationsgyroskop:
  - Vibrierende (Piezo-) Elemente erfahren bei Drehung Korrioliskraft
  - Kann besonders klein gebaut werden
- Lasergyroskop
  - Licht wird im Kreis geschickt
  - Bei Drehung kann Phasenverschiebung gemessen werden



# Inertialtracking: Produkte



[BEI GYROCHIP Model AQRS,  
Automotive-Bereich]



[Intersense Intertrax<sup>2</sup>,  
Spiele]



[BEI SGP50,  
<math>0.005^\circ/\text{sec}/\text{square root of Hz}</math>]



[Honeywell Accelerometers]

# Inertialtracking: Diskussion

## Vorteile:

- Braucht keine Infrastruktur („self-contained“)
- Kleine Bauart möglich
- i.A. sehr günstig
- Können in großer Zahl in Umgebung gebracht werden

## Nachteile:

- Nur relative Messungen
- Fehlerakkumulation durch Integration (Drift)
- Je Sensor kann nur ein Freiheitsgrad gemessen werden
- Kombination mehrerer Sensoren nötig



# Magnetisches Tracking: Grundlagen

- Von Spulen erzeugte magnetische Felder hängen kubisch von Entfernung zur Spule ab
- Veränderliche Felder induzieren Spannung in einer Spule, durch drei orthogonale Spulen kann also 3D-Position/Orientierung bestimmt werden
- Zwei Arten:
  - Niederfrequenter Wechselstrom (AC): leitende Materialien in der Umgebung beeinflussen Feldstärke
  - Pulsierender Gleichstrom (DC): nur durch ferromagnetische Materialien beeinflusst

# Magnetisches Tracking: Produkte



[Ascension MicroBird, DC]



[Polhemus FastTrak, AC]



[Ascension Flock of Birds, DC]

# Magnetisches Tracking: Diskussion

## Vorteile:

- Sehr kleine Sensoren (Stecknadelkopf)
- Keine Sichtverbindung nötig
- Sehr hohe Updateraten
- Geringe Latenzzeit

## Nachteile:

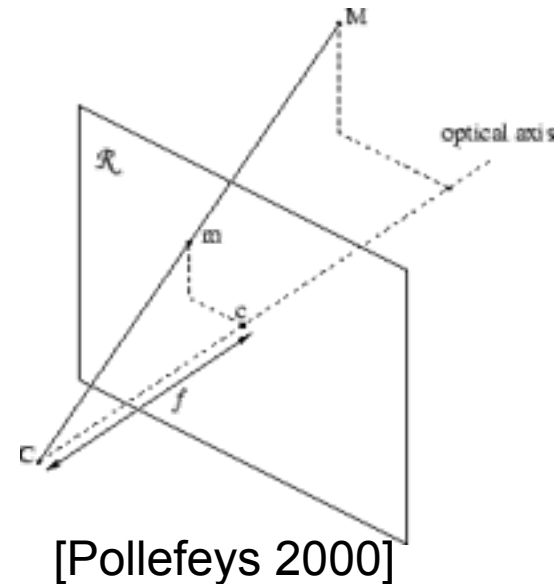
- Sehr kleine Reichweite (Magnetfeld nimmt kubisch ab)
- Starke Verzerrungen durch leitende/ferromagnetische Materialien
- Kabelverbindung zum Sensor nötig

# Optisches Tracking: Grundlage

- Menschen orientieren sich v.a. durch den Sehsinn
- **Idee:** Analyse von Videokamerabildern um Position von Objekten festzustellen
- Zwei Grundarten:
  - **Inside-out:** Kamera ist auf getracktem Objekt
  - **Outside-in:** Feste Kamera beobachtet Objekt(e)
  - Wesentlicher Tradeoff: Positions- vs. Orientierungsgenauigkeit

# Optisches Tracking: Lochkameramodell

- Einfachstes Kameramodell, auch in der Praxis gut geeignet
- Projektion 3D-Punkt  $(x,y,z)$  auf 2D-Punkt  $(u,v)$
- Verfeinerung des Modells durch Scheren, radiale/tangentiale Verzerrungen



$$\begin{pmatrix} u \\ v \end{pmatrix} = \begin{pmatrix} f_x \frac{x}{z} \\ f_y \frac{y}{z} \end{pmatrix}$$

# Optisches Tracking: 2D-3D-Rekonstruktion

- Bildanalyse ergibt 2D-Position mehrerer bekannter 3D-Punkte
- Ziel: Bestimmung Position und Orientierung der Kamera bzw. von Objekten im 3D-Raum
- Mehrere Algorithmen:
  - Tradeoff Rechenzeit vs. Parameter Kameramodell und Genauigkeit
  - Initialisierungsproblem: bei guter Anfangsschätzung von Parametern (z.B. durch Daten aus letztem Frame) geht Berechnung schneller
  - Offlinebestimmung ***intrinsischer Kameraparameter*** (z.B. Brennweite, CCD-Pixelgröße),  
Onlinebestimmung ***extrinsischer Kameraparameter*** (Position und Orientierung)

# Optisches Tracking: Features (Merkmale)

- Einfach:  
künstliche Features („Marker“)
  - Sehr robust
  - Vereinfachte Bildverarbeitung
  - Aber: Umgebung wird zugepflastert
  - Features können auch „aktiv“ sein (z.B. IR-Dioden)
- Schwierig:  
natürliche Features
  - Bsp.: Kanten im Bild, auffällige Texturen, leuchtende Farbflecken
  - i.a. sehr komplex, brandheißes Forschungsthema
  - Hauptproblem: Initialisierung



# Optisches Tracking: Systeme



[AR Toolkit]



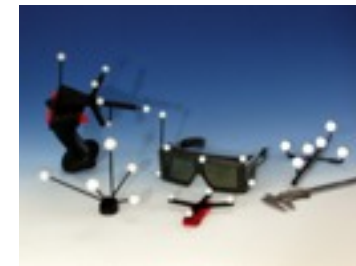
[UNC HiBall]



[Northern Digital Polaris/  
Optotrak Certus]



[AR Solutions, Garching  
b. München]



[ART dTrack, Weilheim]



# Optisches Tracking: Diskussion

## Vorteile:

- i.A. hohe Genauigkeit
- Prinzipiell unbeschränkte Reichweite
- Mobiler Einsatz möglich
- Großes Potenzial durch Moore's Law

## Nachteile:

- Anbringung von Markern nötig (bei derzeitigen Produkten)
- Recht hohe Latenzzeit
- Sichtverbindung nötig
- Anfällig für Jitter

# Weitere Trackingarten

- Mechanisches Tracking
  - Feste Verbindung Tracker zu Objekt
  - Sinnvoll v.a. in Robotikanwendungen
  - Eingabegeräte: Sensable Phantom
- Kompass
  - Global anwendbar
  - Fehleranfällig
  - Elektronischer Kompass basiert auf durch das Erdmagnetfeld induzierten Frequenzänderungen in einem Schwingkreis (pro Freiheitsgrad)

[Kuka 2005]



[Sensable 2005]



[Rickly Hydrological Company 2005]



# Sensorfusion

## ***Problem:***

Es gibt keinen perfekten Tracker.

## ***Idee:***

Kombination mehrerer Sensoren zum Ausgleich von Defiziten

## ***Varianten:***

- Integration mehrerer Sensoren in ein Hardwaregerät, Sensorfusion überwiegend in Hardware gelöst
- Kombination mehrerer Sensoren in Software

# Arten der Sensorfusion

## ***Komplementär:***

- Sensoren hängen nicht voneinander ab
- Bsp.: Kombination GPS mit elektronischem Kompass

## ***Im Wettbewerb:***

- Mehrere Sensoren bestimmen gleiche räumliche Beziehung
- Nutzen: Fehlerminimierung

## ***Kooperativ:***

- Sensoren versorgen sich gegenseitig mit (Teil-) Informationen
- Bsp.: Stereo-Vision, Stabilisierung optisches Tracking durch Gyroskop

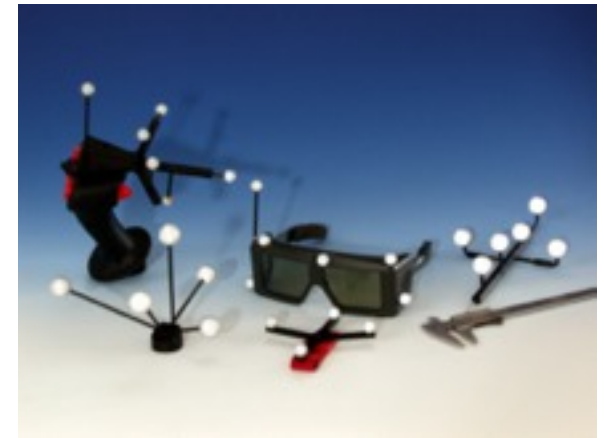
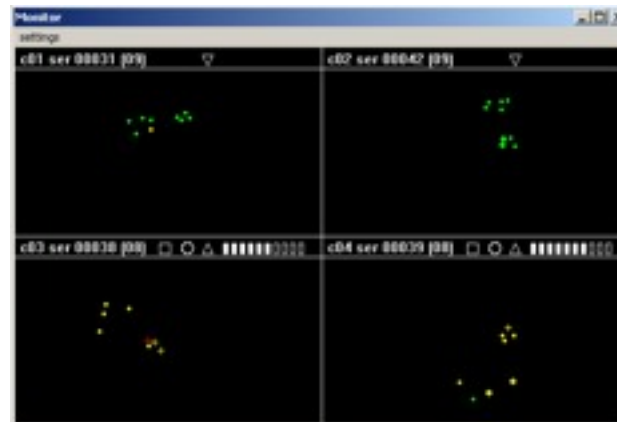
# Sensorfusion: Integrierte Geräte

- Bsp.: XSens MTi
  - 3 Piezo-Gyroskope
  - 3 Accelerometer
  - 3 Magnetometer (Erdmagnetfeldreferenz)
  - Temperaturmessung
  - Ausgabe: absolute 3D-Orientierung
- Bsp.: Intersense IS-600
  - Ultraschall für 3D-Position
  - 3 Accelerometer und 3 Gyroskope für Orientierung
  - Ausgabe: absolute 6DOF-Position



# Sensorfusion: Softwarekombination

- Bsp.: ART dTrack
  - Kameras geben 2D-Position von retroreflektiven Kugeln
  - Software rekonstruiert daraus 6DOF-Position von eindeutigen Kugelanzordnungen („Bodies“)
  - Kosten: > 30.000 €



# Dead Reckoning

- Gegeben:
  - Zuletzt gemessene Geschwindigkeit
  - Zuletzt beobachtete Bewegungsrichtung
  - Vergangene Zeit
- Schätze daraus die aktuelle Position

