

Kognitive Systeme

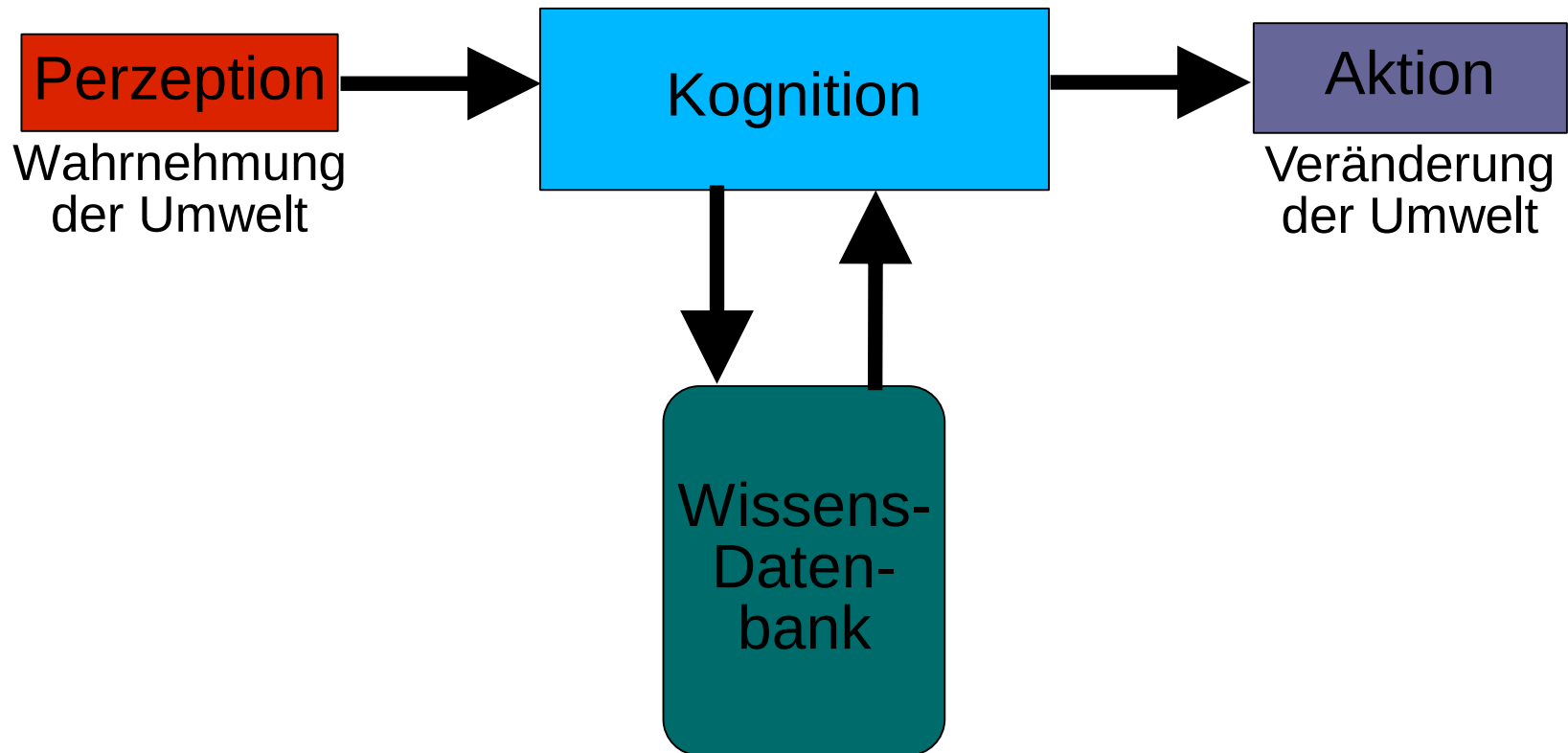
Grundlagen Bildverarbeitung

Mittwoch, 26. April 2017

Zunächst: kurze Wiederholung

Elemente eines Kognitiven Systems

Ingenieurmäßiger Ansatz:



Elemente eines kognitiven Systems:

- **Perzeption:** Aufnahme der Umwelt durch Sensoren (Tasten, Hören, Sehen,...)
- **Repräsentation von Umwelt:** Repräsentation beobachteter Situationen, Zusammenhänge, Strukturen, Handlungen und Reaktionen darauf
- **Kognition:** Auf den Repräsentationen arbeitende Algorithmen
 - Deduktion
 - Induktion
 - Lernen
- **Aktion:** Ausführende Komponente (Textausgabe, Robotersystem,...)

Klassifikation von Sensoren nach ihrer Funktion:

Sensoren kognitiver Systeme



Interner Sensor:

Erfassung innerer Zustände

Bsp: Winkel, Momente, Kräfte, Temperatur, Geschwindigkeit

Externer Sensor:

Information über Zustand der Umwelt

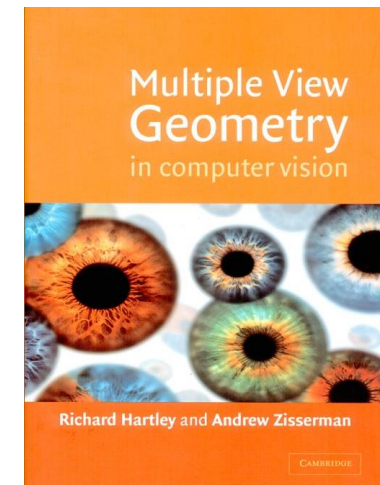
Bsp: Szenen, Situationen, Objekte, Relationen, Prädiktion, Absichten

Literatur:

Azad, Gockel, Dillmann
Computer Vision – Das Praxisbuch
Elektor Verlag
ISBN 3895761656



Hartley, Zisserman:
Multiple View Geometry in Computer Vision
Cambridge University Press
ISBN 0521540518



Literatur:

Bernd Jähne

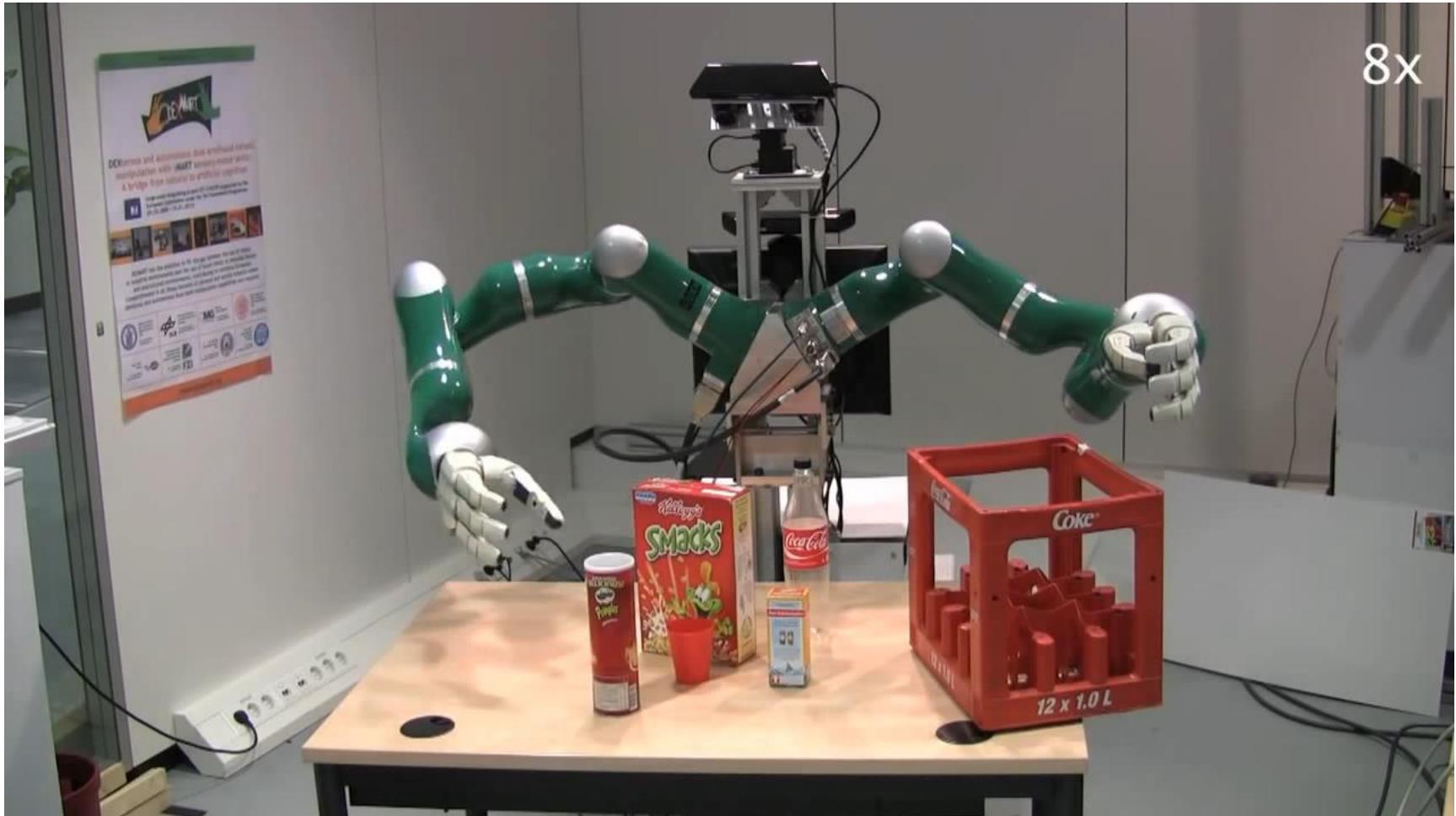
Digitale Bildverarbeitung

Springer Verlag

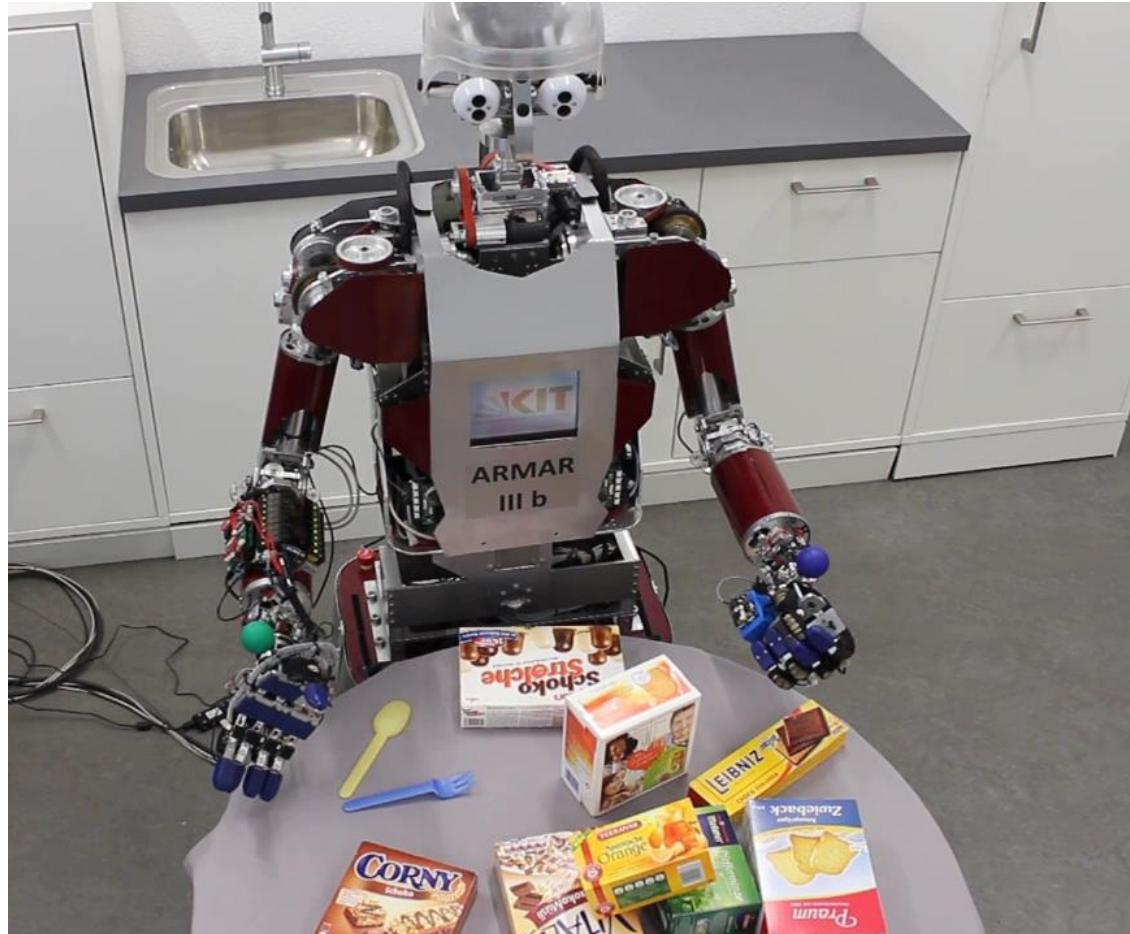
ISBN 3540412603



- Modelliert den visuellen Sinn des Menschen
- Basiert auf Halbleiter-Kameras und Laser-Scanner
- Erlaubt reichhaltige sensorische Information
 - Farbe
 - Merkmale
 - Strukturen
 - Relationen
 - Textur
 - Bewegungen, Handlungen
- Für den Menschen einfach nachvollziehbar



Kombination von maschinellern Sehen und Haptik mit Interaktionen



Initial object hypotheses

Generate **hypotheses** based on
Color, **Geometric primitives**
and **Saliency**

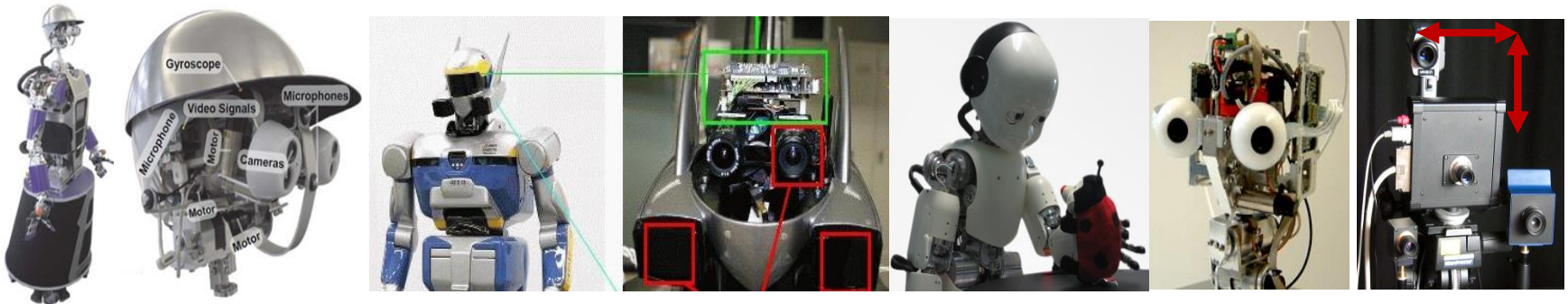
Hypothesis 49 is chosen
for verification by pushing



Probleme bei der visuellen Datenverarbeitung

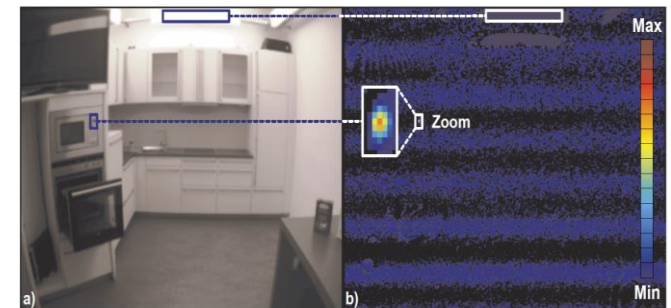
Humanoid robot embodiment restrictions

- Limited camera size
- Complex perturbations
- Extensive requirements



Environmental adverse conditions

- Lighting flickering
- Dynamic day lighting
- Complex materials/structures



Kritischer Aspekte zur visuellen Kognition

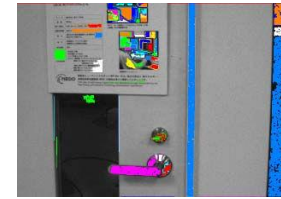
■ Humanoid robots

■ Self-localization

- Thompson et al. 2005
- Stasse et al. 2006

■ Enviromental object recognition

- Okada et al. 2008
- Yokoi et al. 2011



■ Computer vision

■ Radiometric calibration

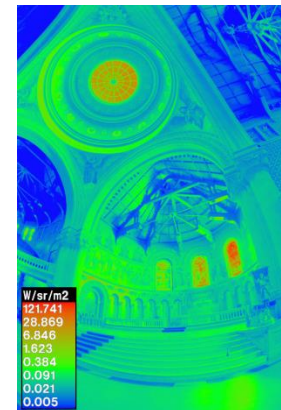
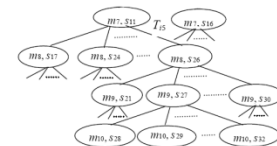
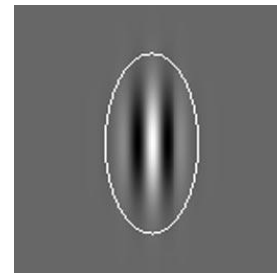
- Debevec et al. 2001
- Krawczyk et al. 2007

■ Edge extraction

- Grigorescu et al. 2003

■ Visual Feature Matching

- Fang Huang et al. 2010



Grundprinzip: Programmieren durch Vormachen

- A human demonstrates a task naturally, incorporating all implicit and explicit domain knowledge
- The demonstration is observed by sensors and robot task knowledge is generated from that data



Specific PbD
methodology



Beispiele sensorischer Bewegungserfassungssysteme

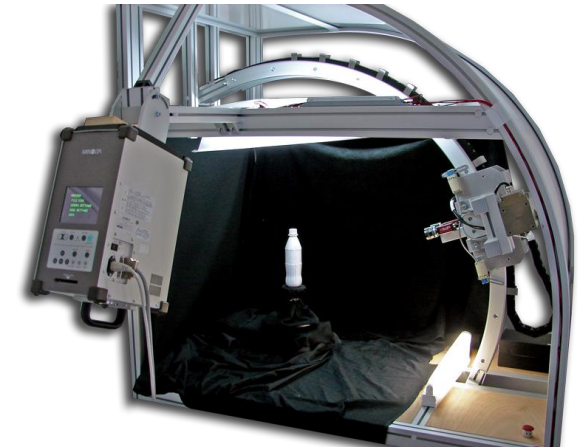
■ Object Modeling Center

- Generation of high-precision, textured 3D-models

■ PbD-Dome

- Observation of human during manipulation

■ Vicon Tracking System



Object Modeling Center



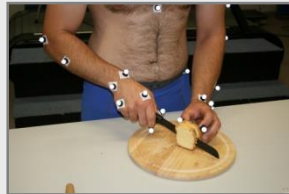
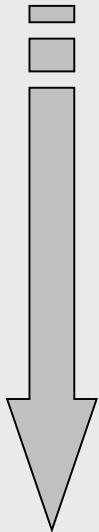
PbD-Dome



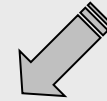
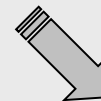
Vicon Tracking System

Bewegungserfassung und Aufzeichnung von Handlungen

Sequential Motion



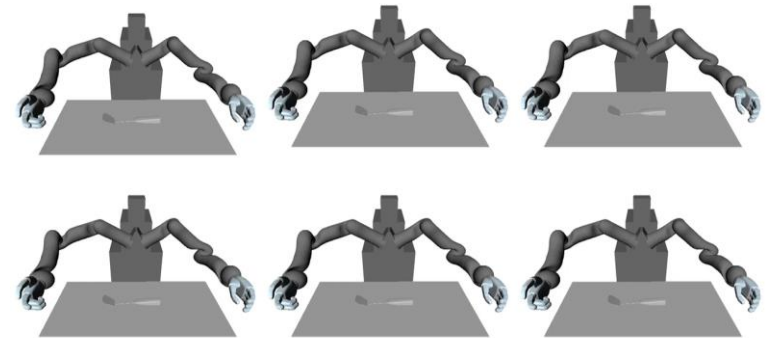
Parallel and coordinated Motion



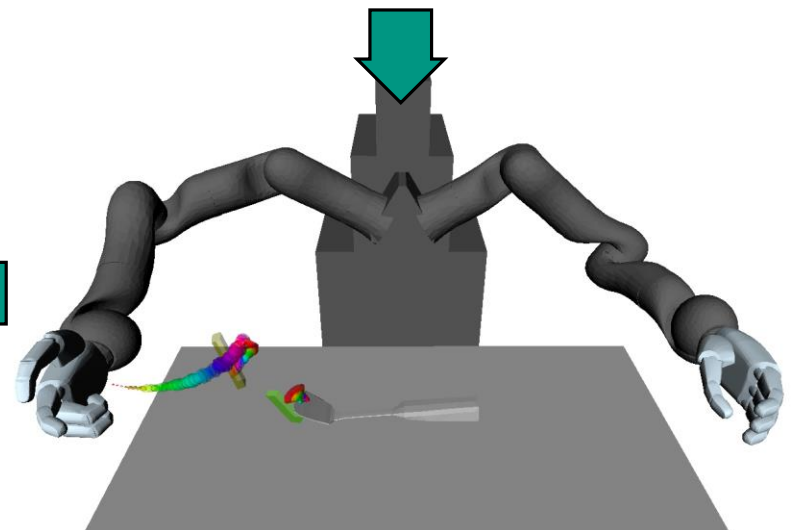
Aufzeichnung und Repräsentation von Handlungen



Human demonstrations



(Semi-)Automatic constraint selection

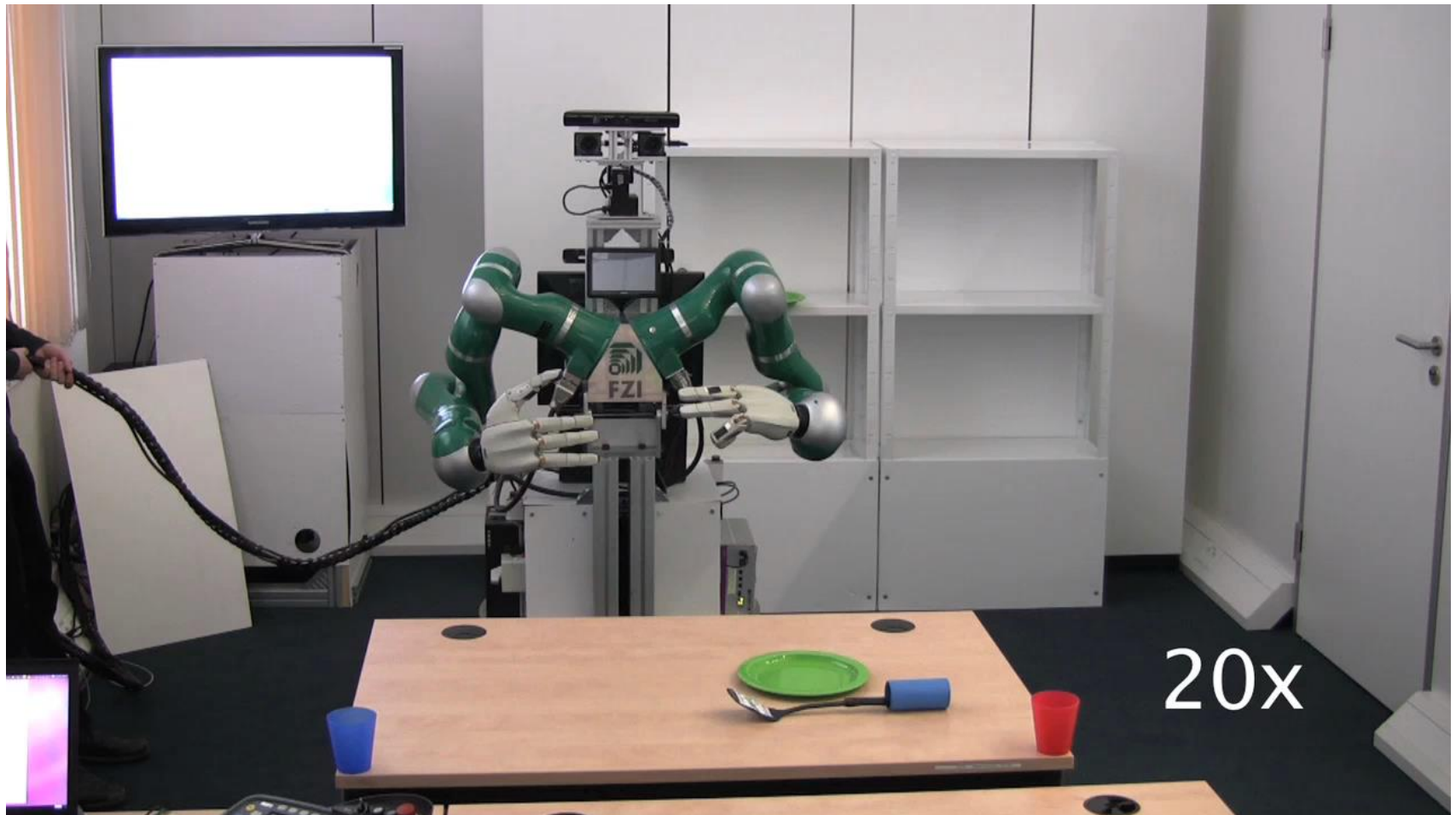


Automatic constraint optimization

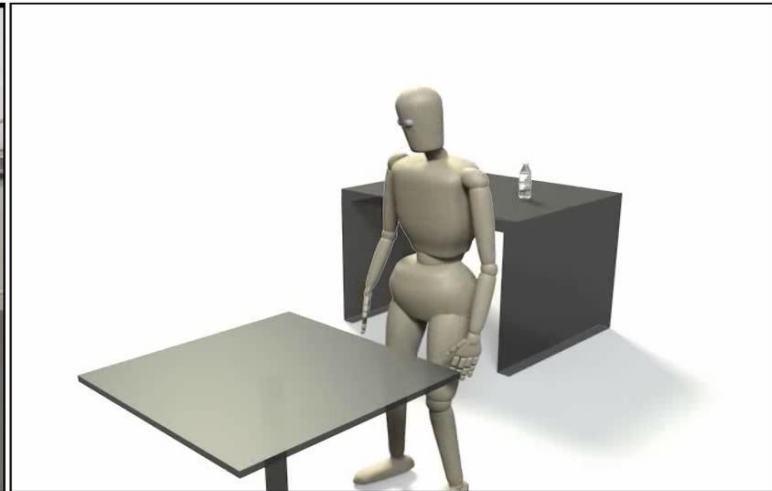
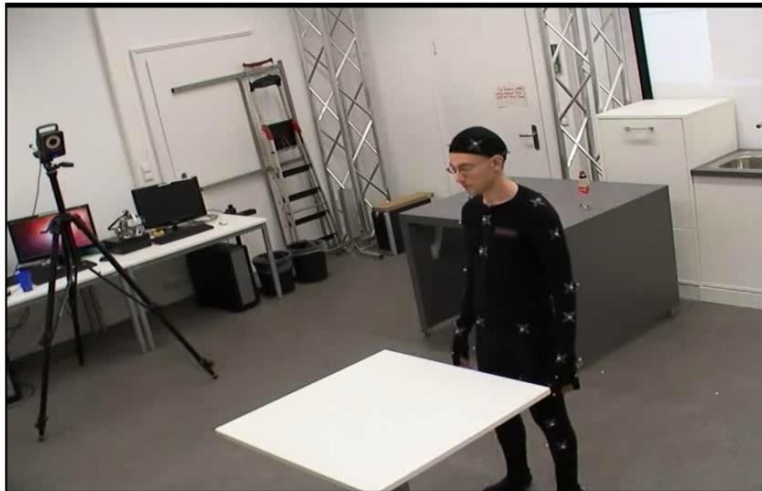


Execution using 2D-vision to localize objects

Ausführung beobachteter und gelernter Handlungen



Lernen von Ganzkörperbewegungen



Conversion of Human and Object Motions with the MMM Framework

<https://motion-database.humanoids.kit.edu/>

Lernen aus Beobachtungen

- Building a library of motion primitives
- Dynamic movement primitives (DMP) for discrete and periodic movements



Humanoids 2006, IJHR 2008, Humanoids 2007, ICRA 2009, Humanoids 2009, TRO 2010, Humanoids 2012, IROS 2013, RAS 2015



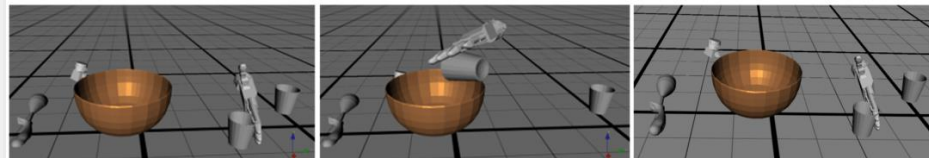
Verstehen von Handlungen

- Hierarchical segmentation approach which not only considers motion but also relevant objects
 - **Semantic** segmentation based on object relation
 - **Motion** segmentation based on trajectory characteristics

Human Demonstration



Converted Demonstration



Object-Relation Segmentation

No contact Cup in left hand No contact

Motion Characteristic Segmentation & Recognition

Grasp Lift Pour Place Retreat

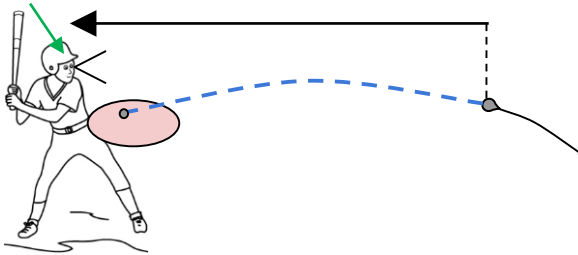
Observation, Representation and Segmentation of preparing batter

Based on Geometrical Simulation &
Object Contact Relation Changes

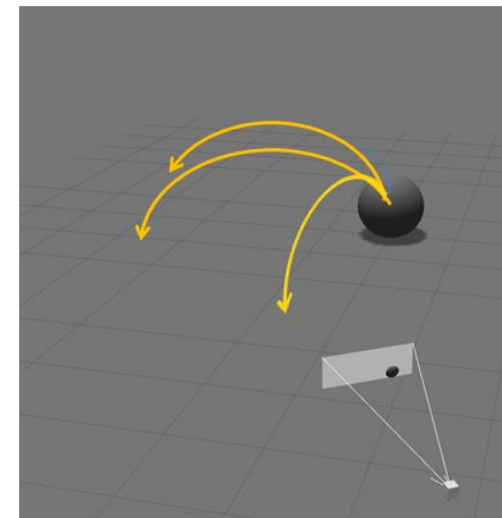
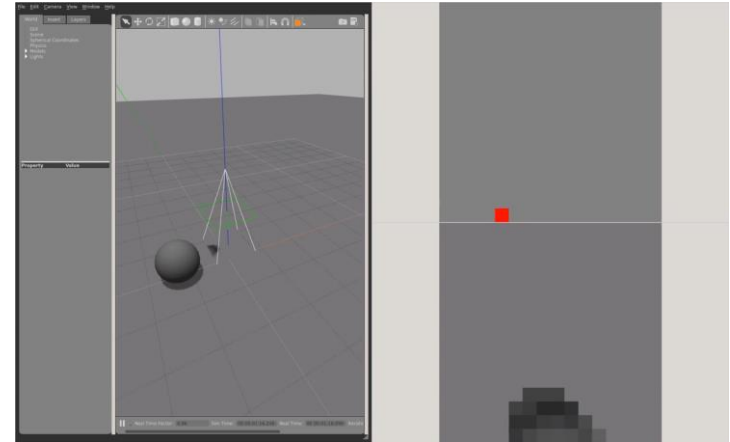
Lernen aus Beobachtungen



Beobachtung und Prädiktion

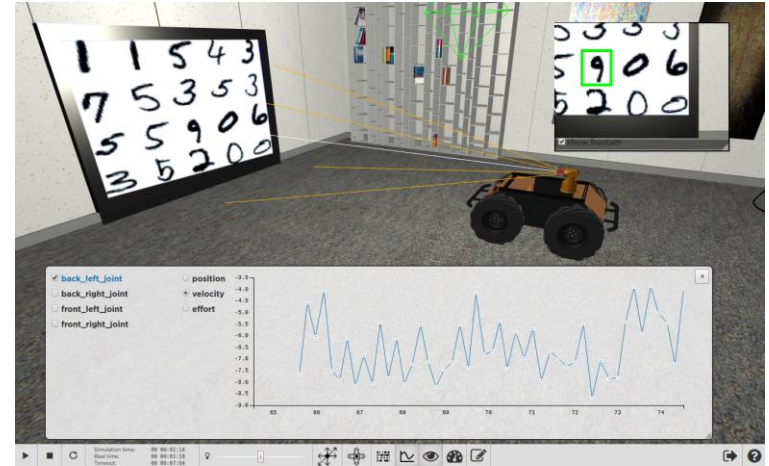


- “Where is the object and where will it be next?”
 - Detection of motion / Saliency
 - Optical flow
 - Localization
- Input data:
 - Address events (DVS-Camera)
- Output:
 - Prediction of future address events
 - 3D ball position
- Neural candidate:
 - Liquid State machines

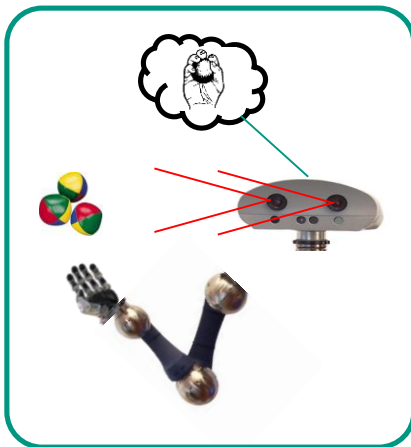


Visuelle Deutung und Objektklassifikation

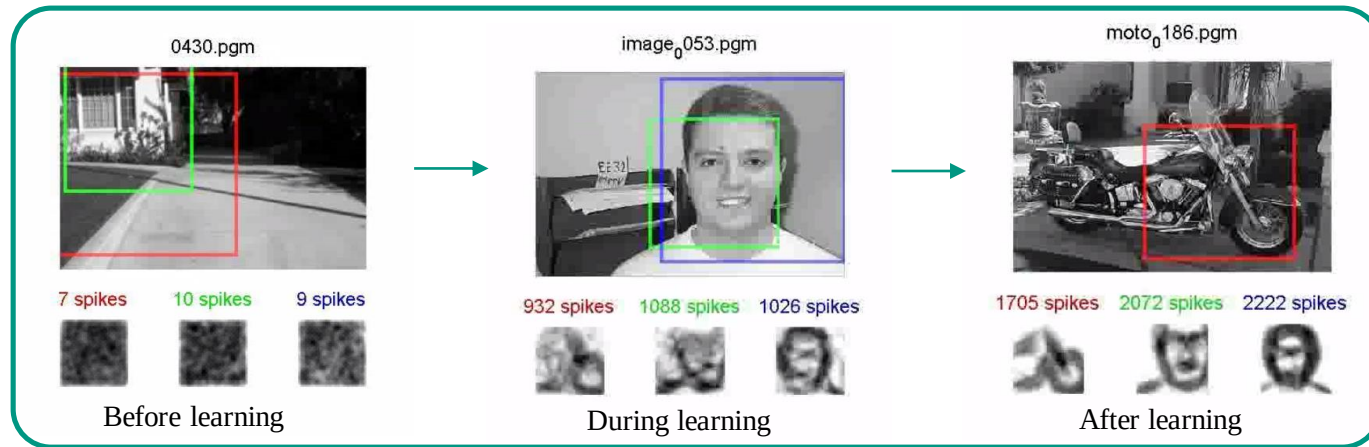
- “What is the object?”
 - Object recognition
 - Prediction of affordances
- Input data:
 - Camera image stream (RGB)
 - Address events (DVS)
- Output: Grasp type, Object name, Object location, ...
- Neural candidates:
 - Spiking Convolutional Restricted Boltzmann Machine
 - HMAX



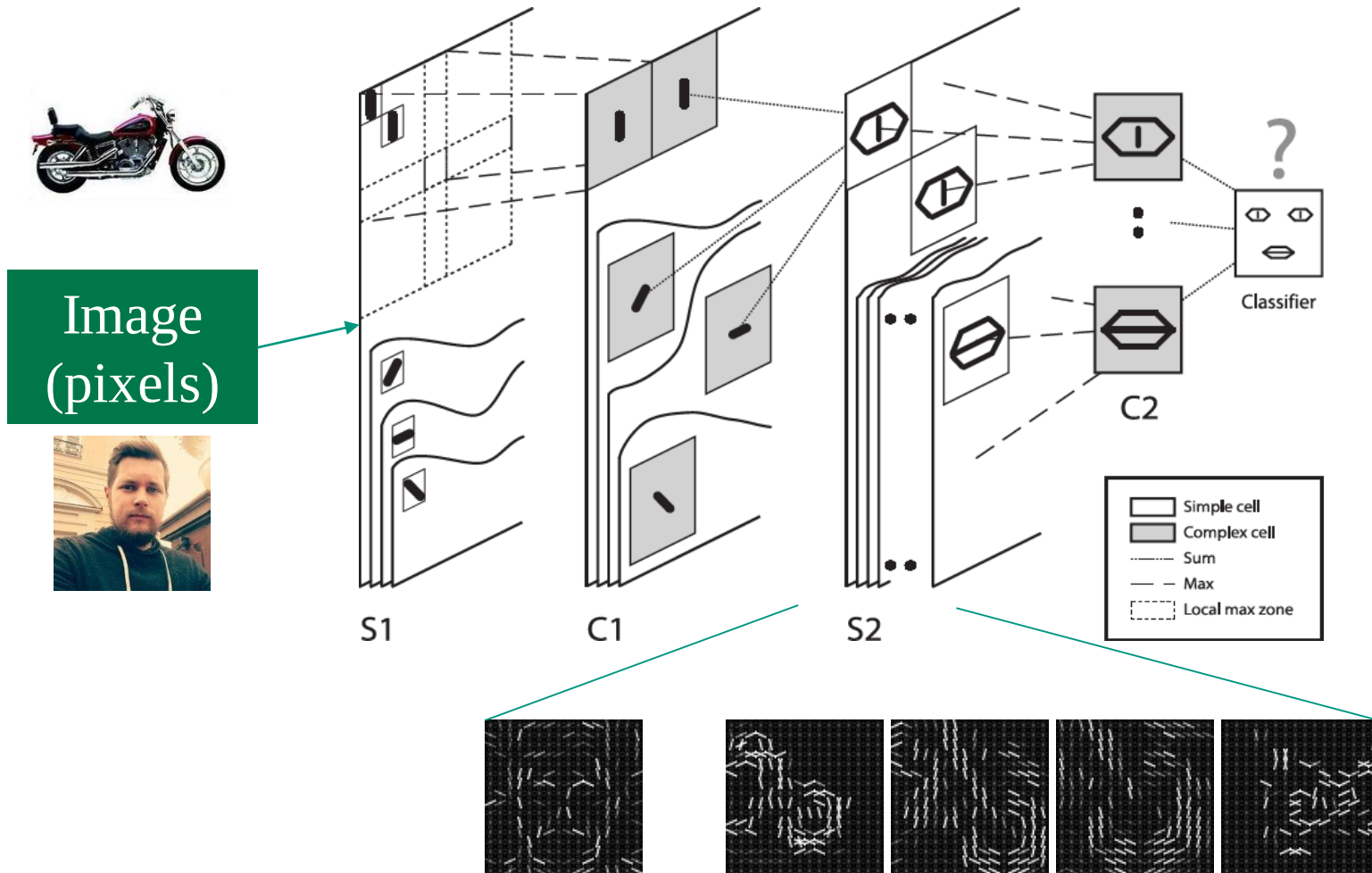
Desired outcome of recognition experiment



Masquilier T, Thorpe 2007

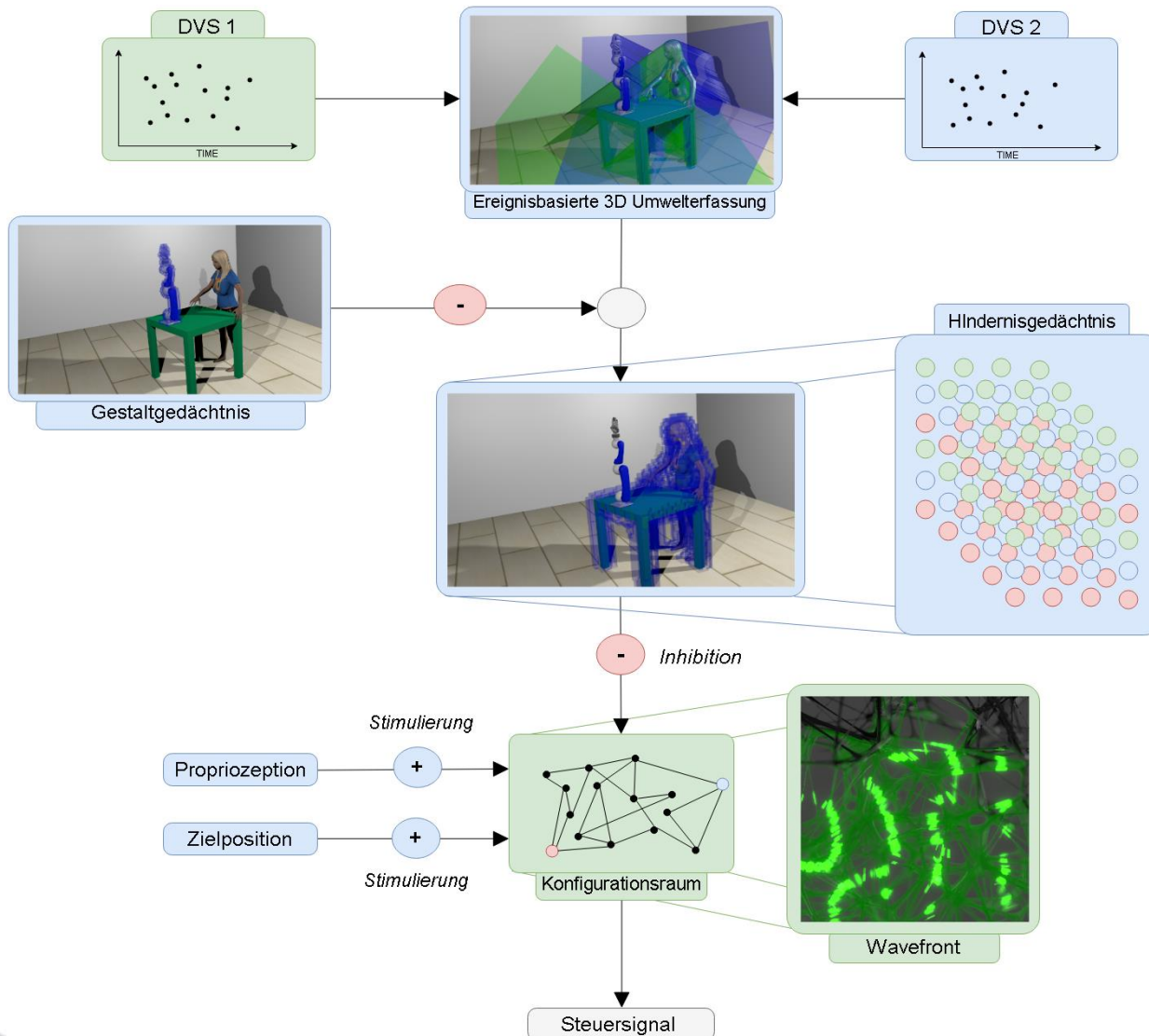


Lernen visueller Merkmale mit Hilfe von Faltungsnetzen (convolutional networks)



Architecture from Masquelier T, Thorpe 2007

Motion Planning



- 3D dynamic vision
- From stereoscopic DVS
- Active occupancy memory (grid)
- Sensing robot
- Sensing obstacles
- Wavefront planning and control in configuration space
- Occupancy memory works together with learnt body schema to prohibit planning wave traveling through configurations resulting in collision

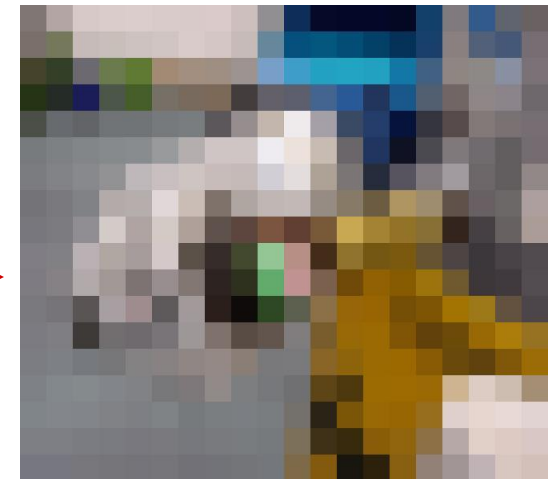
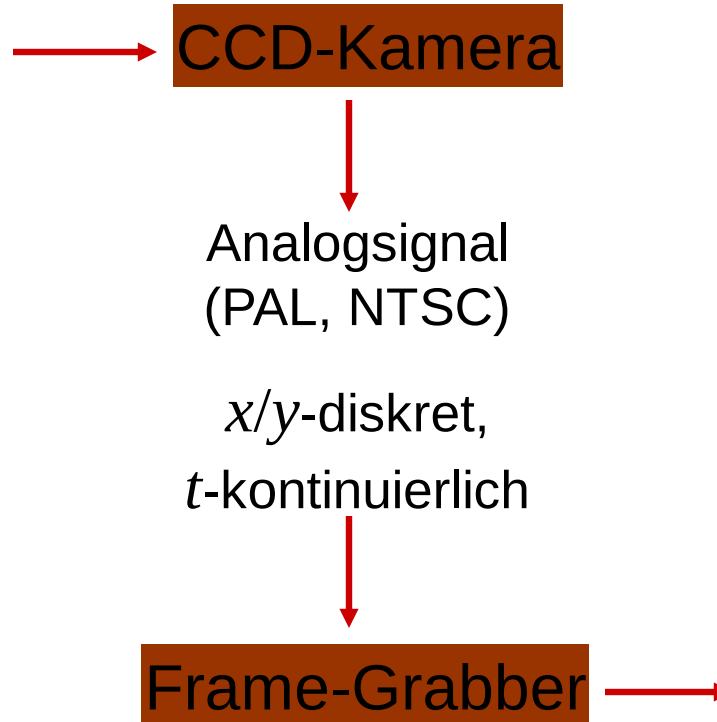
Heutige Vorlesung:

- Bilderzeugung
 - Analoge Kameras
 - Digitalel Kameras
- Bildrepräsentation
 - Graustufenbilder, Farbbilder, Pixelarrays
 - Konvertierung von Farbräumen
- Lochkameramodell
- Homogene Punktoperatoren (affine, nicht-affine)
- Farb – Histogramme zur Charakterisierung digitaler Bilder
- Automatische Bildverbesserung und Kontrastanpassung

Bilderzeugung: Analoge Kameras



x/y -kontinuierlich



x/y -diskret

- Heute werden oftmals Digitalkameras verwendet (meist CCD, aber auch CMOS)
- Anschluss erfolgt über:
 - Firewire (IEEE1394) (inzwischen weit verbreitet)
 - USB (hauptsächlich für Webcams)
 - Camera Link
 - ...
- Kameras liefern direkt digitalisierte Bilddaten
 - Format je nach Kamera und Modus unterschiedlich
 - Bei S/W-Kameras meist 8bit Graustufen
 - Bei Farbkameras entweder als Bayer-Pattern oder meist bereits konvertiert als RGB24, YUV422, etc.

- Kameras unterscheiden sich durch:
 - Bildqualität (Qualität CCD-Chip, aber auch grundlegend: Auswahl Linse/Objektiv)
 - Graustufen- oder Farbkamera
 - Auflösung (z.B. 640×480, 1024×768, ...)
 - Bei Farbkameras: Welche Farbkodierungen sind verfügbar? (8bit, 16bit, 24bit)
 - Wichtig je nach Anwendung:
Welche maximale Framerate ist bei welchem Bildformat noch verfügbar? (z.B. 15/30/60/120/200 Hz)

Bilderzeugung: Digitale Kameras 3

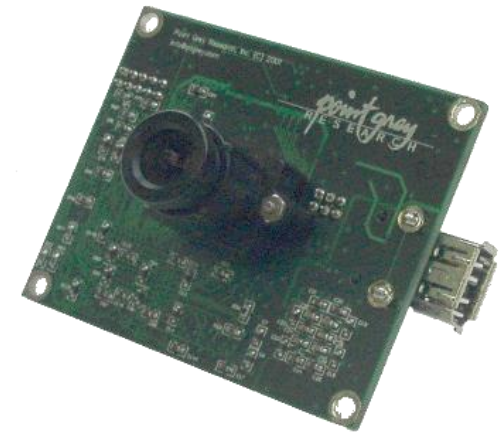
■ Beispiele



PointGrey Dragonfly Express
mit CS Mount



SAC Osiris I mit CS Mount



PointGrey Dragonfly
mit Microlens

■ Beispiele von Stereokamerasystemen mit fester Baseline



PointGrey Bumblebee mit Microlens

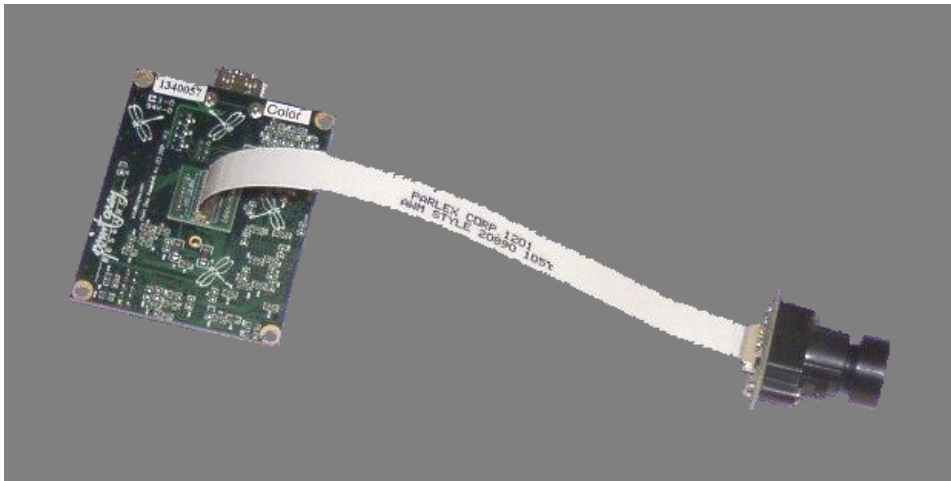


PointGrey Digiclops, drei Kameras, Microlens

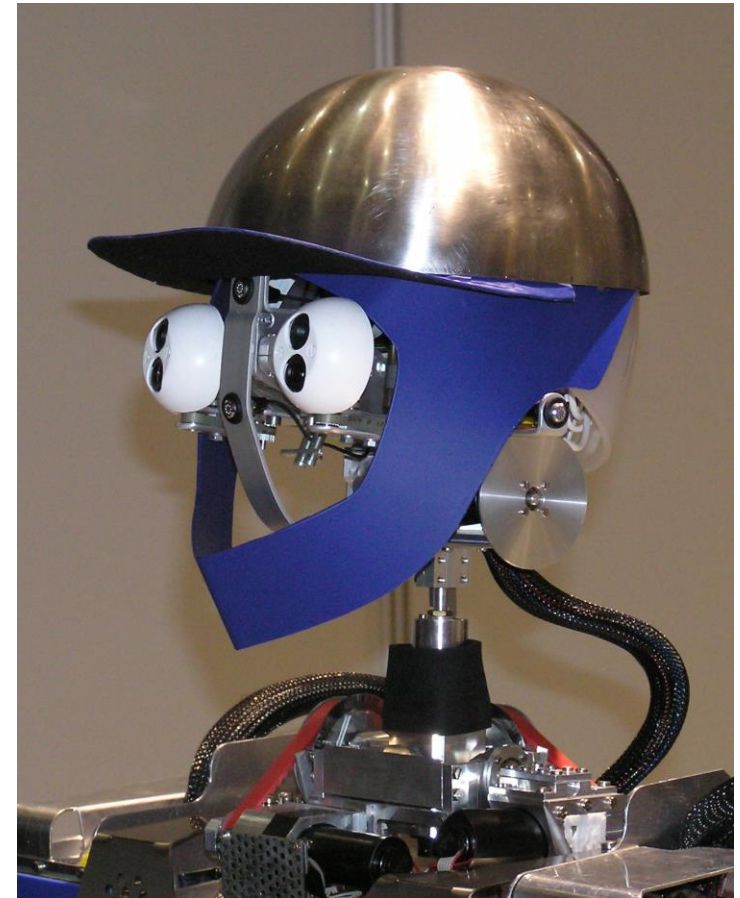


Videre STH-DCSG mit CS Mount

Kameras in ARMAR-III und ARMAR-IV



PointGrey Dragonfly Kameras, extended Version, wie in ARMAR II und ARMAR III verwendet



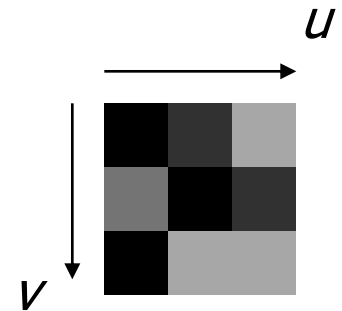
Hals und Kopf von ARMAR III, Augeneinheit mit vier Dragonfly Kameras

Bildrepräsentation 1

Monochrombild: Diskrete Bildfunktion

$$\text{Img} : [0..n-1] \times [0..m-1] \rightarrow [0..q]$$

$$(u, v) \mapsto \text{Img}(u, v)$$



Üblich:

$$q = 255$$

$$n = 640, m = 480 \text{ (VGA)}$$

oder $n = 768, m = 576 \text{ (PAL)}$

Bildrepräsentation 2

Farbbild:

- Viele verschiedene Farbmodelle für unterschiedliche Anwendungen
- Klassifikation nach erreichbarem Farbraum

Beispiele:

- S/W ,Grauwertstufen
- RGB-Modell: speziell für Monitore (Phosphor-Kristalle), sehr üblich

$$\text{Img}(u, v) \in \mathbf{R}^3 = (r, g, b)^T$$

- HSI (Hue, Saturation, Intensity): speziell für Farbsegmentierung
- CIE: physikalisch (Wellenlänge)
- CMYK-Modell: Farbdrucker (subtraktive Farbmischung)
- YIQ: Fernsehermodell

$$\text{Img} : [0..n-1] \times [0..m-1] \rightarrow \\ [0..R] \times [0..G] \times [0..B]$$

$$(u, v) \mapsto \text{Img}(u, v) = (r, g, b)$$

- Additive Farbmischung
- Drei Farbwerte: Rot, Grün, Blau
oft: $256 \times 256 \times 256$ Nuancen
($R=G=B=255$, 8 Bit, „RGB24“)
=16,8 Mio. Farben
- Oft verwendet von Kamera-Treibern

Bildrepräsentation 4: HSI-Modell

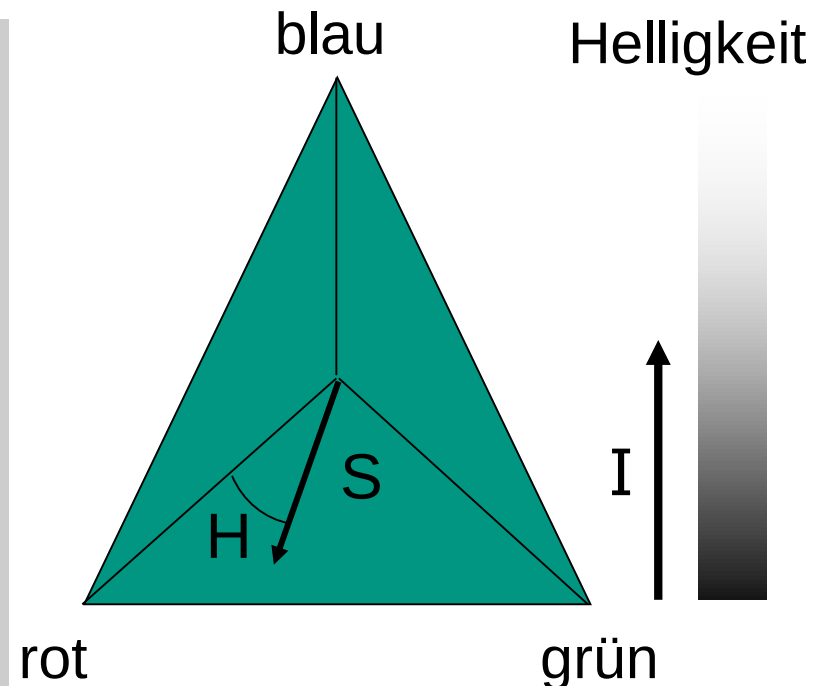
- Hue (Farbnuance), Saturation (Sättigung), Intensity (Helligkeit)
- Trennt Helligkeit vom Farbwert $\Rightarrow$ unempfindlich gegen Beleuchtungsänderungen
- Umrechnung von RGB nach HSI (falls $R = G = B$, dann ist H undefiniert; falls $R = G = B = 0$, dann ist S undefiniert):

$$c = \arccos \frac{2R - G - B}{2\sqrt{(R - G)^2 + (R - B)(G - B)}}$$

$$H = \begin{cases} c & \text{falls } B < G \\ 360^\circ - c & \text{sonst} \end{cases}$$

$$S = 1 - \frac{3}{R + G + B} \min(R, G, B)$$

$$I = \frac{1}{3}(R + G + B)$$

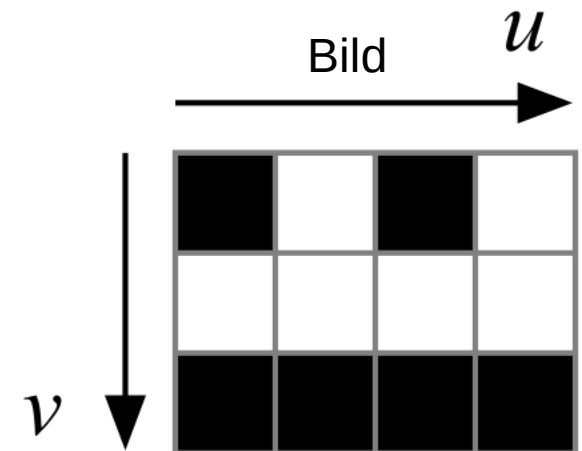
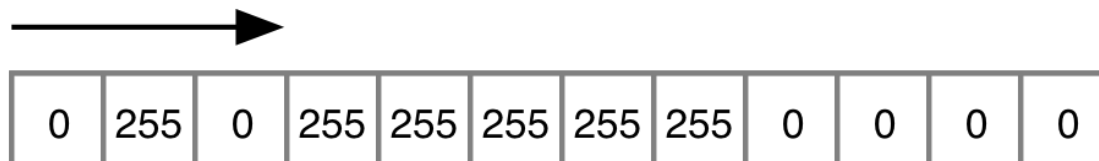


■ Hinterlegung eines 8bit Graustufen-Bildes im Speicher

- Pixel werden zeilenweise, von oben links nach unten rechts, linear abgelegt (Achtung: z.B. bei Bitmaps von unten links nach oben rechts)

■ Graustufen-Kodierung:

- Ein Byte pro Pixel
- 0 schwarz, 255 weiß, dazwischen Graustufen

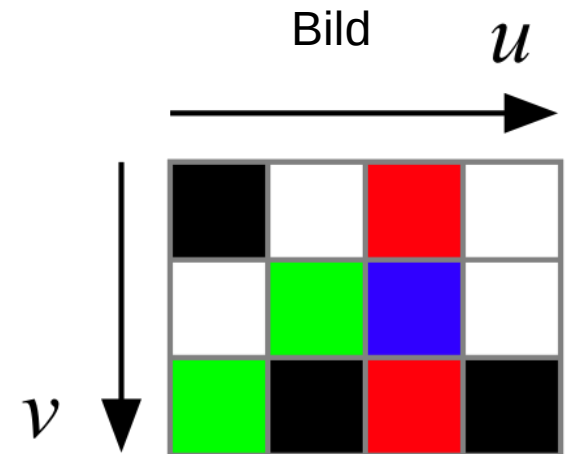
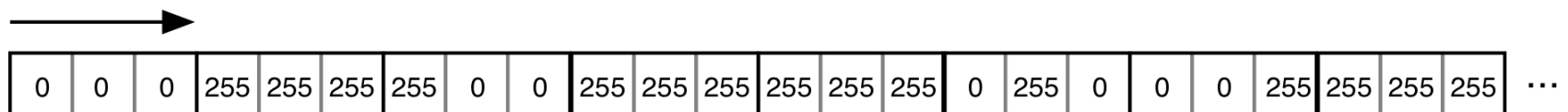


zeilenweise „aufgerollt“

Hinterlegung im Speicher

■ Hinterlegung eines RGB24 Farbbildes im Speicher

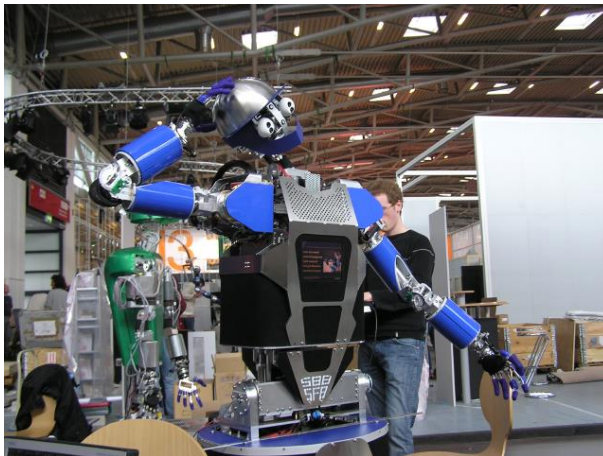
- Pixel werden zeilenweise, wie beim Graustufen-Bild, abgelegt
- Farbkodierung:
 - Drei Bytes pro Pixel
 - Für jeden Kanal gilt:
0 minimale, 255 maximale
Intensität; dazwischen Nuancen



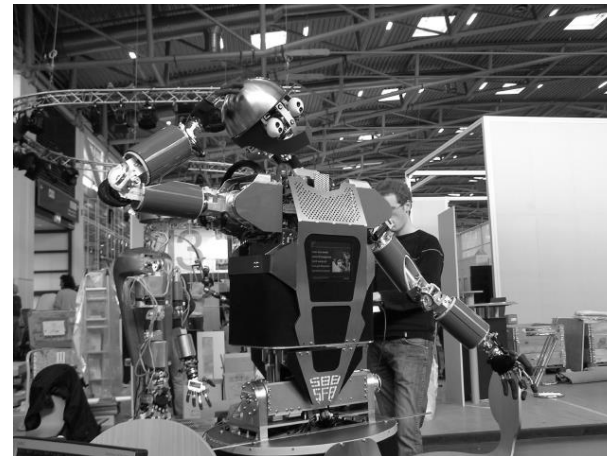
zeilenweise „aufgerollt“

Hinterlegung im Speicher

- Transformation von RGB24 nach 8bit Graustufen
 - Eine Möglichkeit: $g = (R + G + B) / 3$
 - Aber: Menschliches Auge ist am empfindlichsten gegenüber der Farbe Grün, ...
 - Üblicherweise wird deshalb verwendet:
 $g = 0.299 \cdot R + 0.587 \cdot G + 0.114 \cdot B$



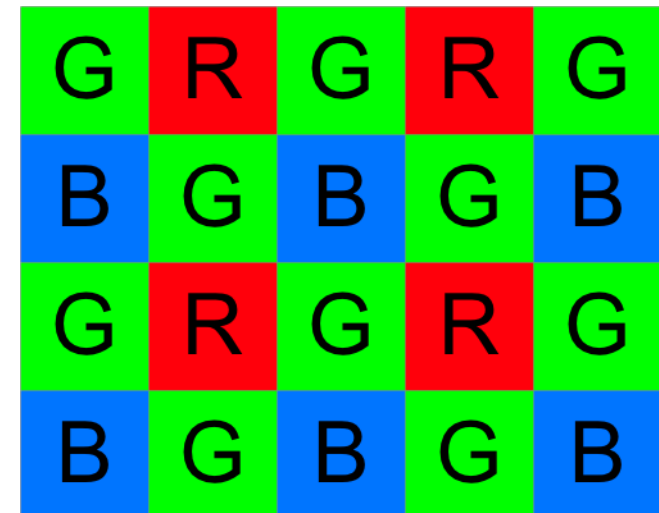
Farbbild



Graustufenbild

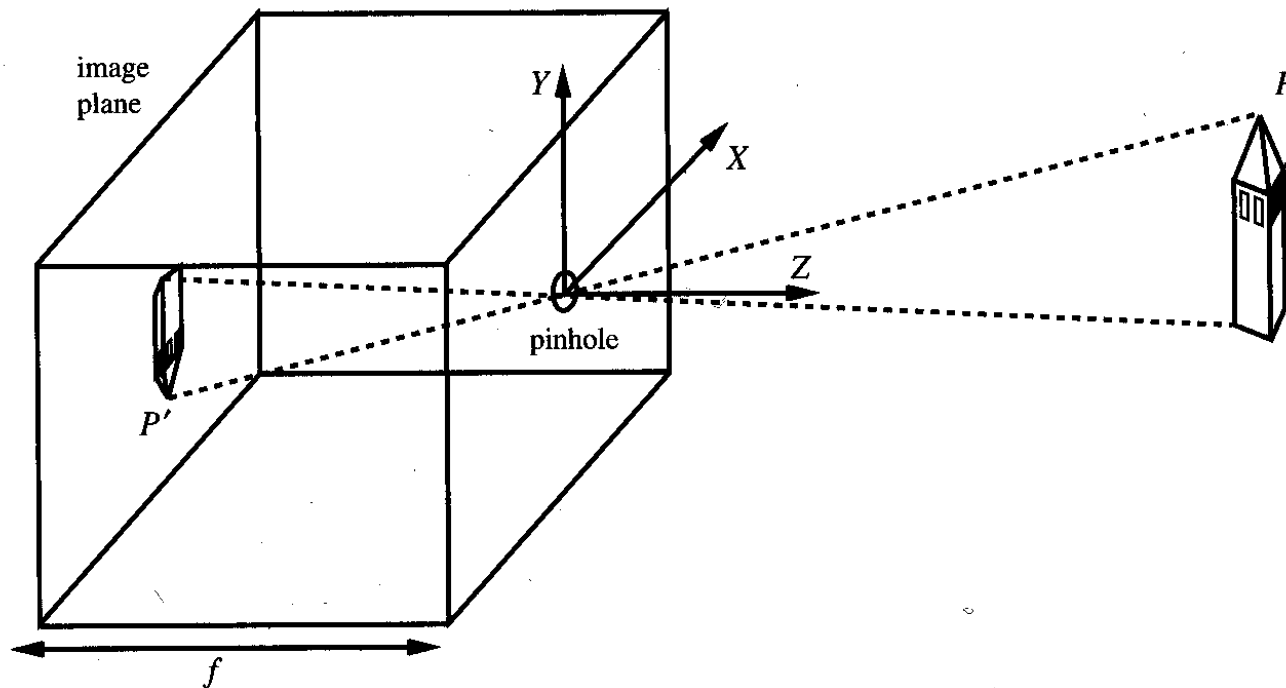
Bayer-Pattern

- Sehr hochwertige Kameras, wie z.B. zum Filmen verwendet, besitzen drei Chips pro Pixel
- Die meisten Farbkameras haben einen Chip pro Pixel, der gegenüber der Farbe Rot, Grün oder Blau empfindlich ist
- Bei „Ein-Chip-Kameras“ wird meist das *Bayer-Pattern* verwendet
 - Um nach RGB24 zu konvertieren, muss interpoliert werden
 - Die Empfindlichkeit einer Ein-Chip-Kamera ist um Faktor 3 niedriger als die einer reinen Graustufen-Kamera!



Bilderzeugung: Lochkamera 1

- Einfachstes Modell auf Basis linearer Strahlenoptik :
Lochkameramodell



Interner Parameter: Brennweite f („Fokalabstand“)

Bildgenerierung: Lochkamera 2

Projektion eines Szenenpunktes $P = (X, Y, Z)$ auf einen Bildpunkt $p = (u, v, w)$:

$$\frac{-u}{f} = \frac{X}{Z}, \frac{-v}{f} = \frac{Y}{Z}, w = -f \Rightarrow X = -\frac{uZ}{f}, Y = -\frac{vZ}{f}$$

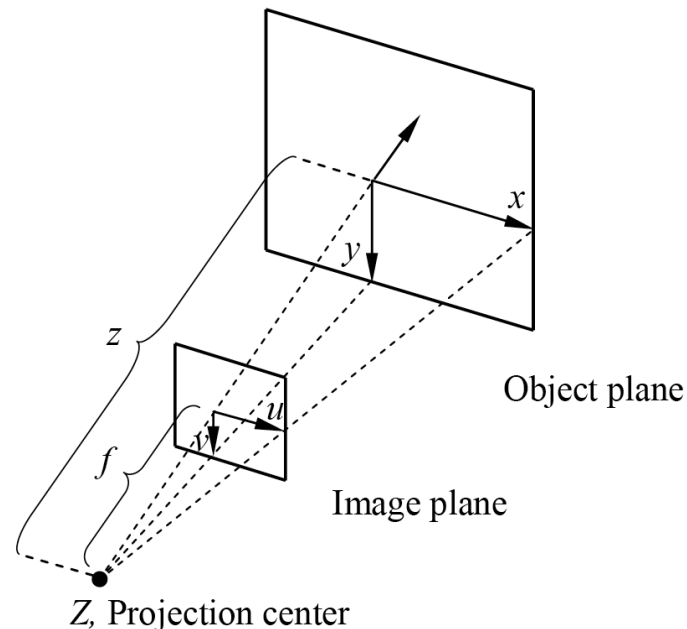
$$p = \begin{pmatrix} u \\ v \\ w \end{pmatrix} = \begin{pmatrix} u \\ v \\ -f \end{pmatrix} = -\frac{f}{Z} \begin{pmatrix} X \\ Y \\ Z \end{pmatrix} = -\frac{f}{Z} P$$

Rückprojektion

Perspektivprojektion

Bei der Projektion geht die Z -Komponente verloren!

- Oft verwendete Variante:
Lochkameramodell *in Positivlage*:
 - Projektionszentrum C liegt hinter der Bildebene
 - Dadurch: keine Spiegelung (Minuszeichen entfallen)



Bildverarbeitung

Bildverarbeitung: Grundoperationen

- Homogene Punktoperatoren
- Histogrammauswertung
- Filterung
- Geometrische Operationen



Unterdrückung von Störungen
„Verschönern“ von Bildern
Verformen von Bildern

Homogene Punktoperatoren

- Anwendung: $\text{Img}'(u,v) = f(\text{Img}(u,v))$
- Unabhängig von der Position bzw. den Nachbarn des Pixels
- Implementierung der Funktion f oftmals als Look-Up-Table (Hardware)

Affine Punktoperatoren 1

Affine Punktoperatoren: $f : [0..q] \rightarrow [0..q]$
 $x \mapsto ax + b$

Parameter a und b legen die Funktion fest.

Anwendungen:

- Kontrasterhöhung / -verminderung
- Helligkeitserhöhung / -verminderung
- Invertierung
- und Kombinationen

- Algorithmus für die Anwendung eines affinen Punktoperators

Algorithmus 8 AffinePunktoperation(I, a, b) $\rightarrow I'$

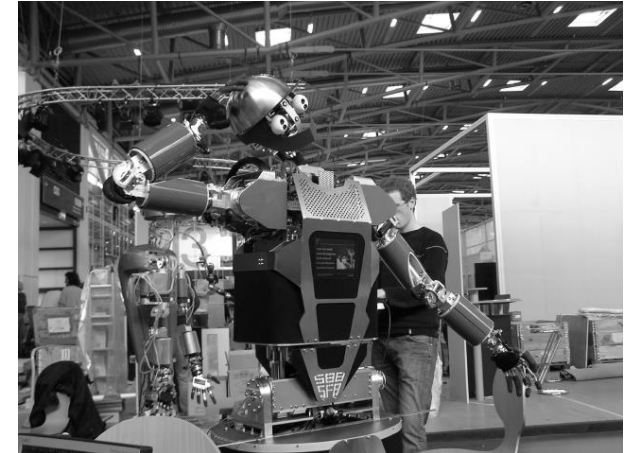
for all pixels (u, v) in I **do**
 $g := \text{round}(a \cdot I(u, v) + b)$
 $I'(u, v) := \begin{cases} 0 & \text{falls } g < 0 \\ q & \text{falls } g > q \\ g & \text{sonst} \end{cases}$
end for

Affine Punktoperatoren 3

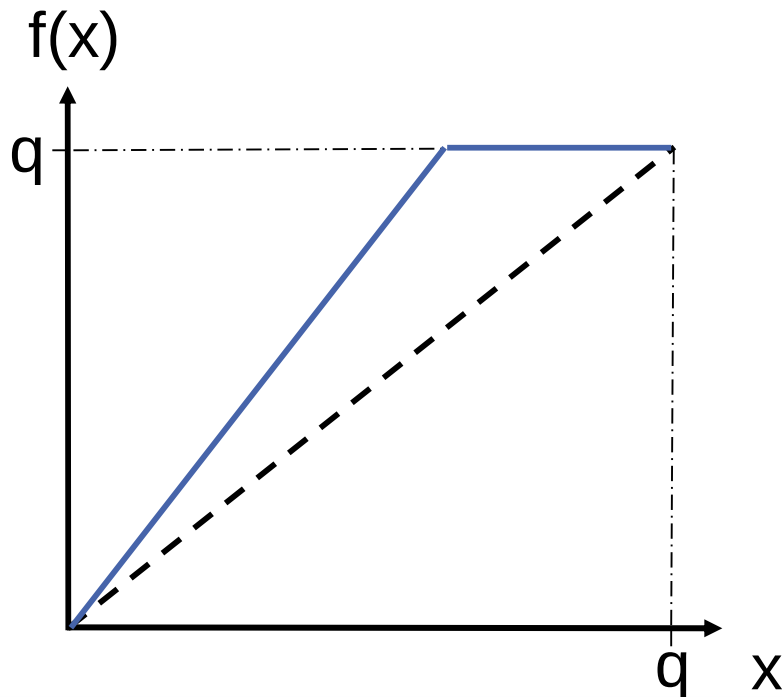
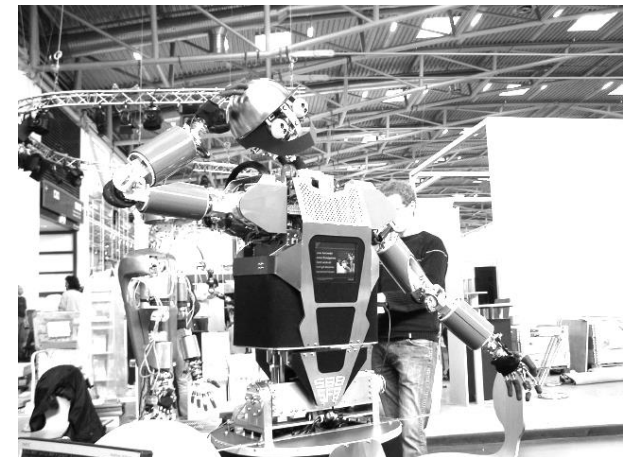
Kontrasterhöhung:
 $b = 0; a > 1$

Beispiel: $f(x) = 2x$

Original



Ergebnis

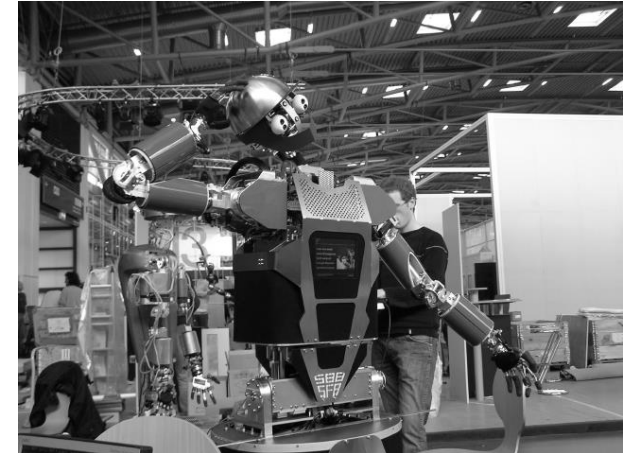


Affine Punktoperatoren 4

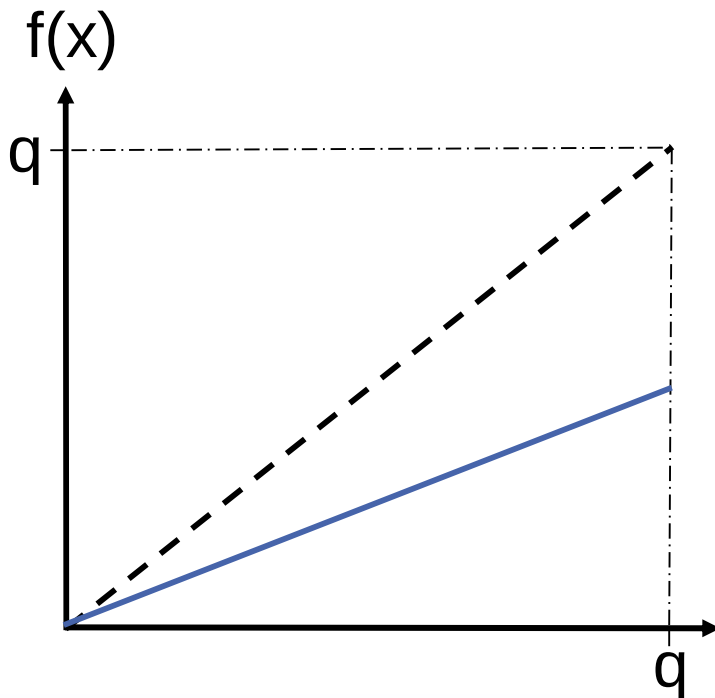
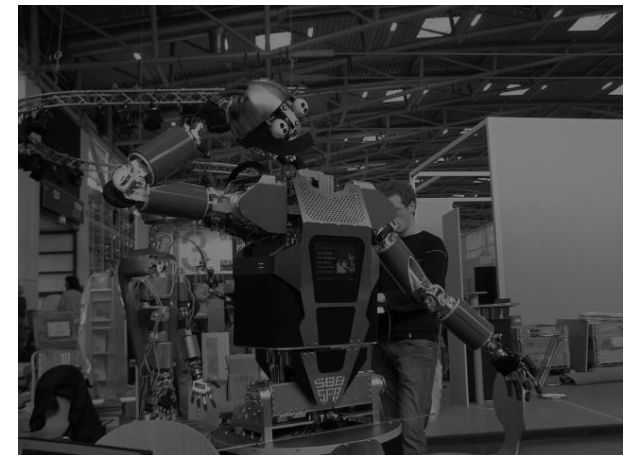
Kontrastverminderung:
 $b = 0; a < 1$

Beispiel: $f(x) = 0.5x$

Original



Ergebnis



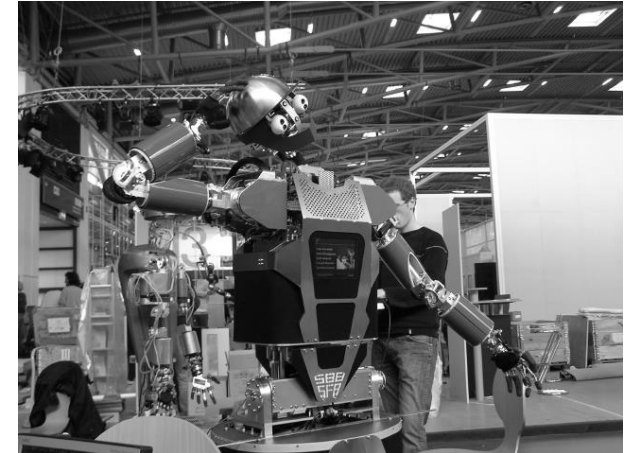
Affine Punktoperatoren 5

Helligkeitserhöhung:

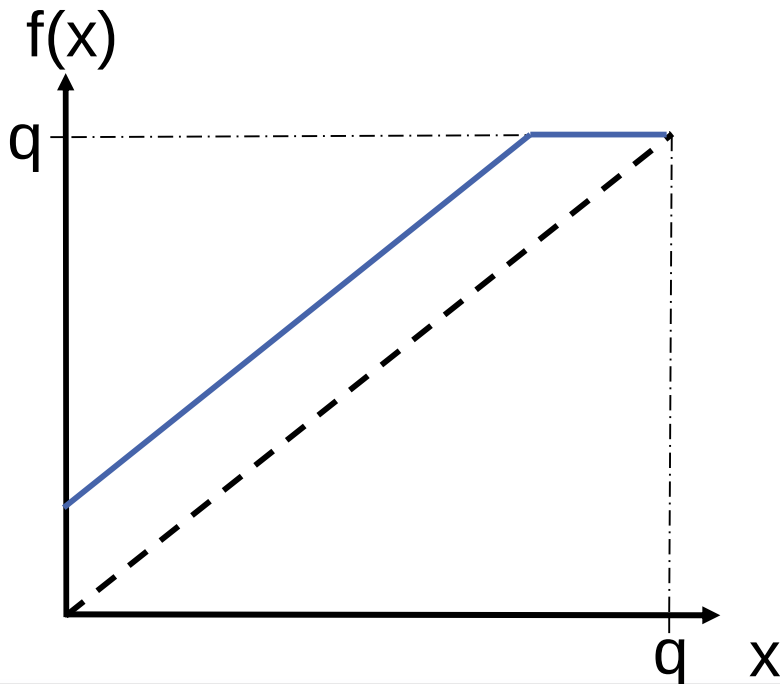
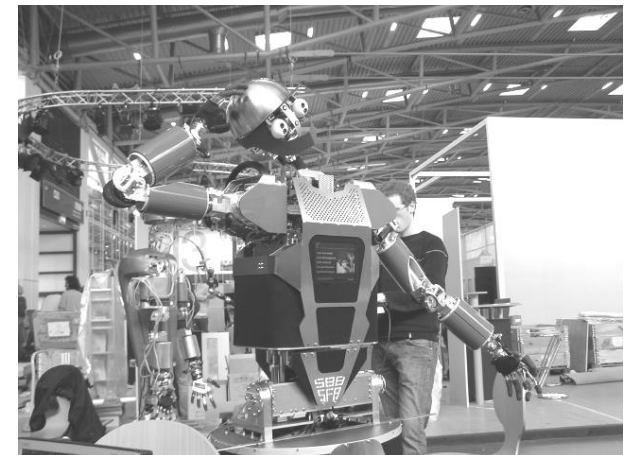
$$b > 0; a = 1$$

Beispiel: $f(x) = x + 50$

Original



Ergebnis

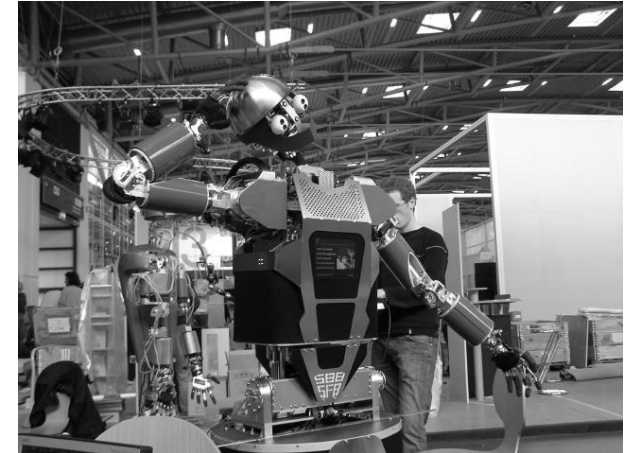


Affine Punktoperatoren 6

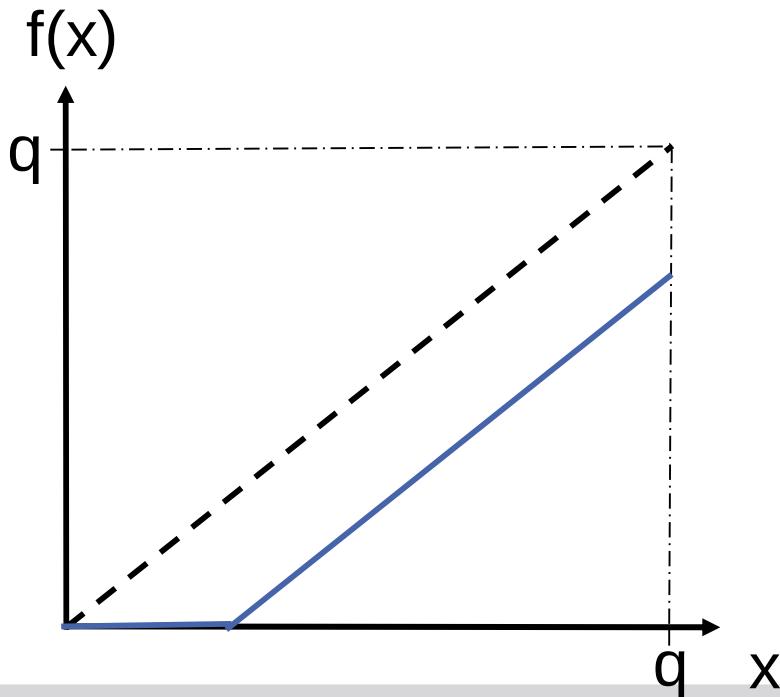
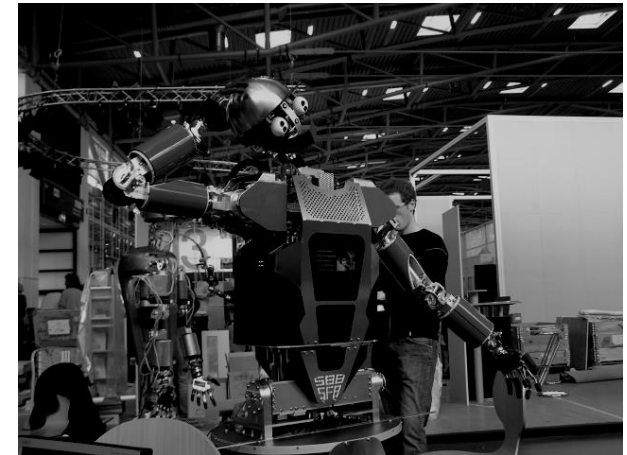
Helligkeitsverminderung:
 $b < 0; a = 1$

Beispiel: $f(x) = x - 50$

Original



Ergebnis

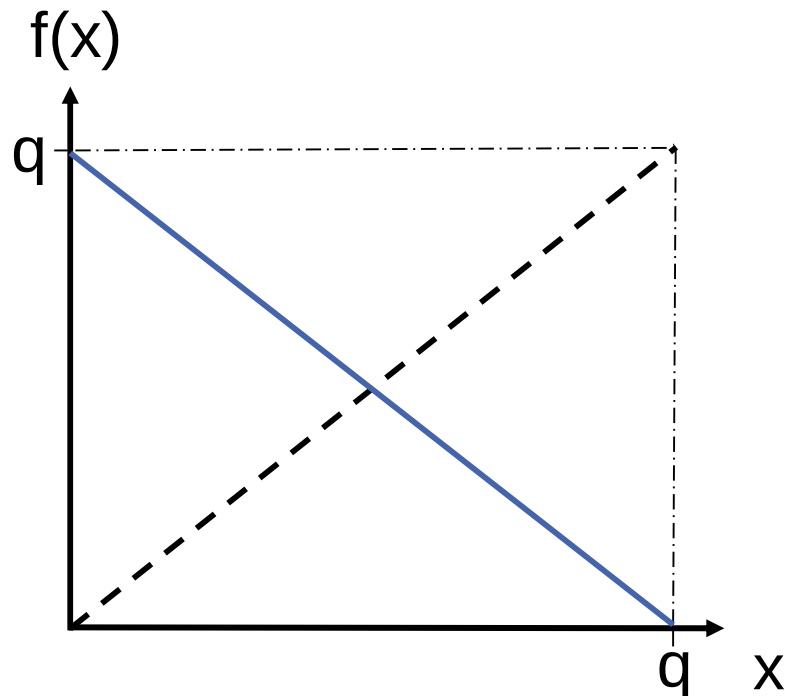


Affine Punktoperatoren 7

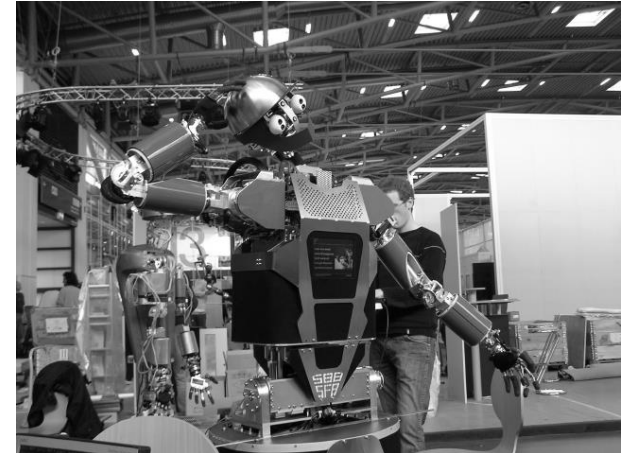
Invertierung:

$$b = q; a = -1$$

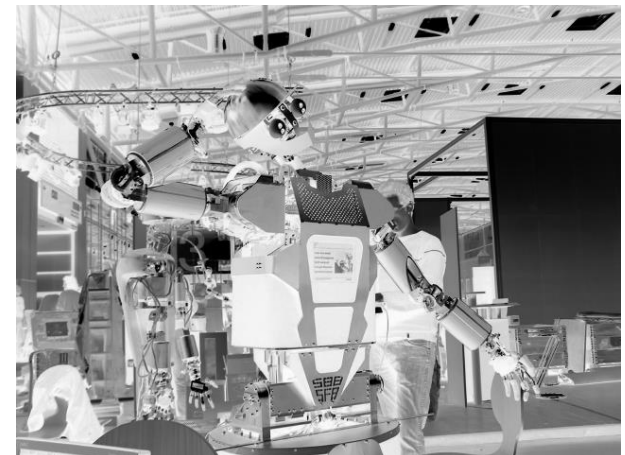
Beispiel: $f(x) = -x + q$



Original



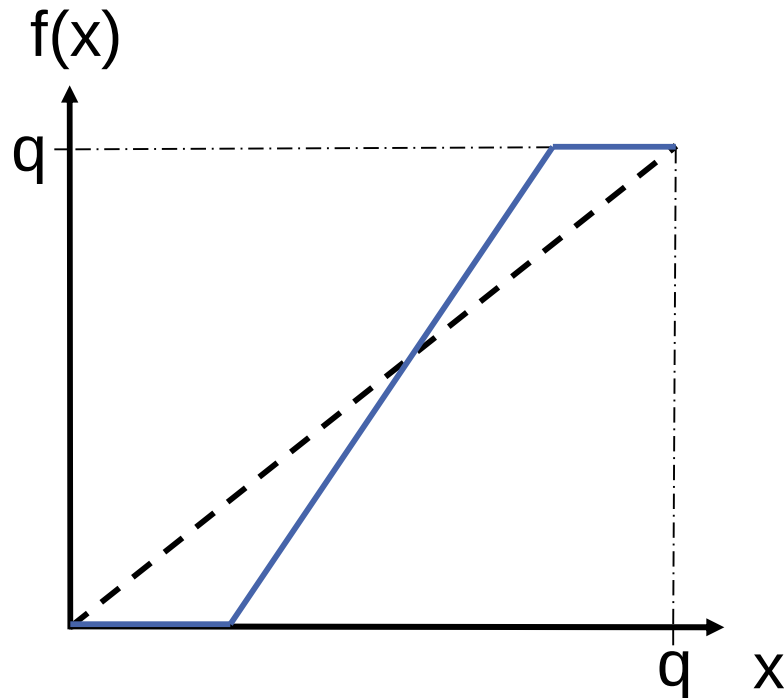
Ergebnis



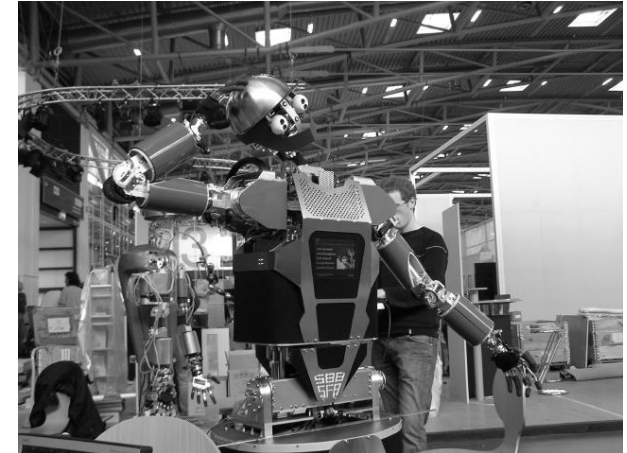
Affine Punktoperatoren 8

Kombinationen:

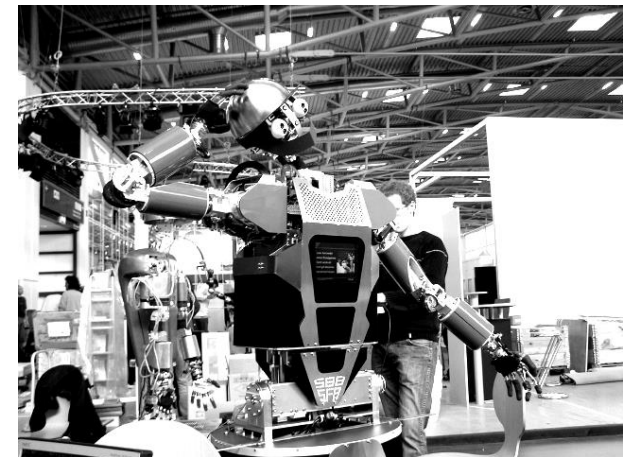
Beispiel: $f(x) = 2x - 50$



Original

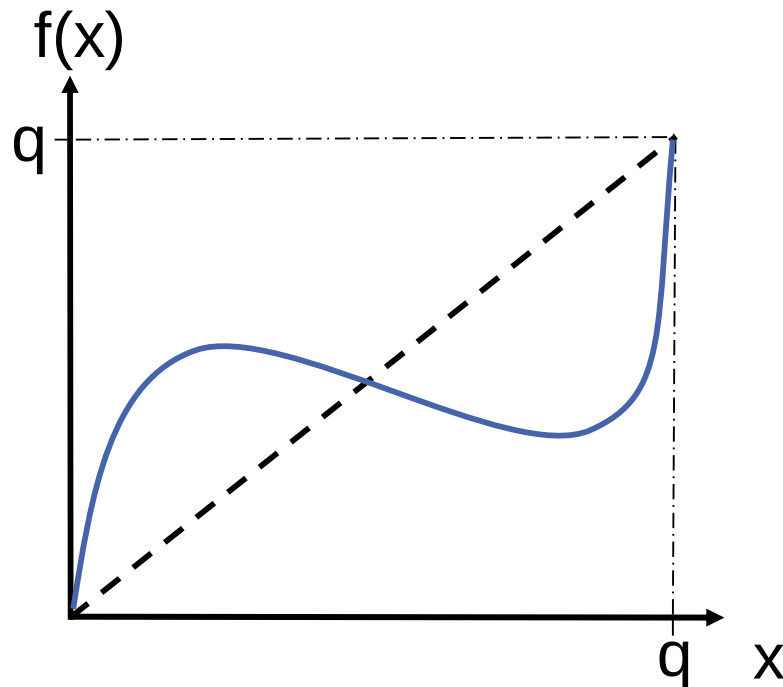


Ergebnis

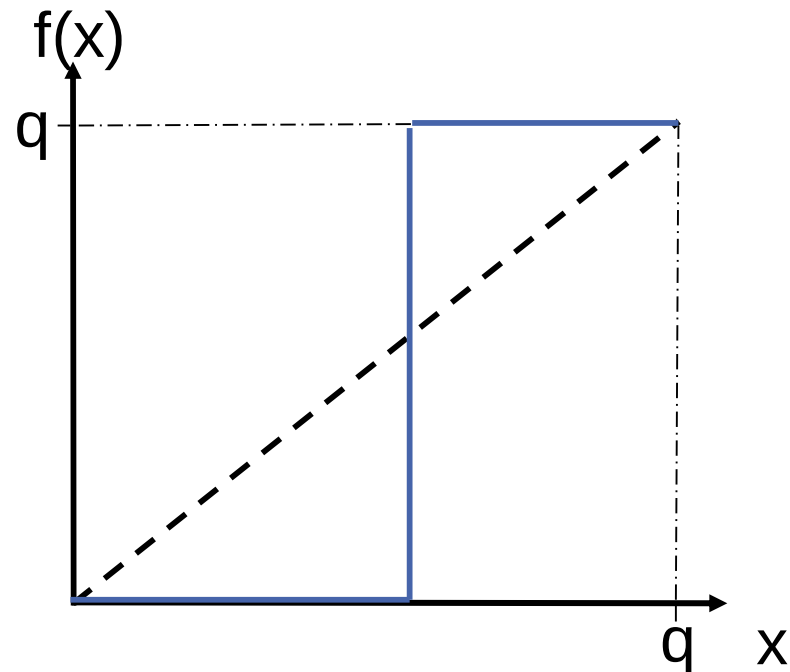


Nicht-affine Punktoperatoren

Beliebige Abbildungsfunktion $f : [0..q] \rightarrow [0..q]$



Sensor-Ausgleich



Binarisierung

Nicht-affine Punktoperatoren

Anwendung:

- Ausgleich von Sensor-Nichtlinearitäten
- Gewichtung
- Binarisierung

- Spreizung
- Histogrammdehnung
- Histogrammausgleich

- Die Spreizung berechnet in einem Bild zunächst die minimale Intensität min und die maximale Intensität max
- Anschließend wird das Intervall $[min, max]$ linear auf $[0, 255]$ abgebildet und es werden entsprechend die Intensitäten im Bild angepasst
- Die Spreizung ist eine affine Punktoperation
- Nachteil:
 - In der Praxis nicht robust
 - Nur unter optimalen Bedingungen anwendbar

Abbildungsvorschrift: $I'(u, v) = q \cdot \frac{I(u, v) - \min}{\max - \min}$

Als affine Punktoperation: $I'(u, v) = \frac{q}{\max - \min} \cdot I(u, v) - \frac{q \cdot \min}{\max - \min}$

Algorithmus 11 BerechneMinMax(I) $\rightarrow \min, \max$

```
min := q, max := 0
for all pixels (u, v) in I do
  if  $I(u, v) < \min$  then
    min :=  $I(u, v)$ 
  end if
  if  $I(u, v) > \max$  then
    max :=  $I(u, v)$ 
  end if
end for
```

Algorithmus 12 Spreize(I) $\rightarrow I'$

$\min, \max \leftarrow \text{BerechneMinMax}(I)$ {Algorithmus 11}

```
if  $\min = \max$  then
  STOP {Bild I ist homogen}
end if
```

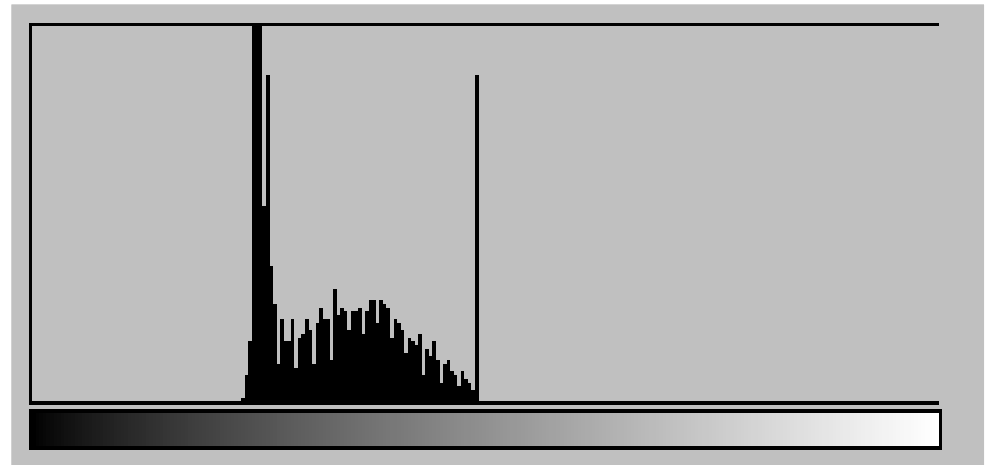
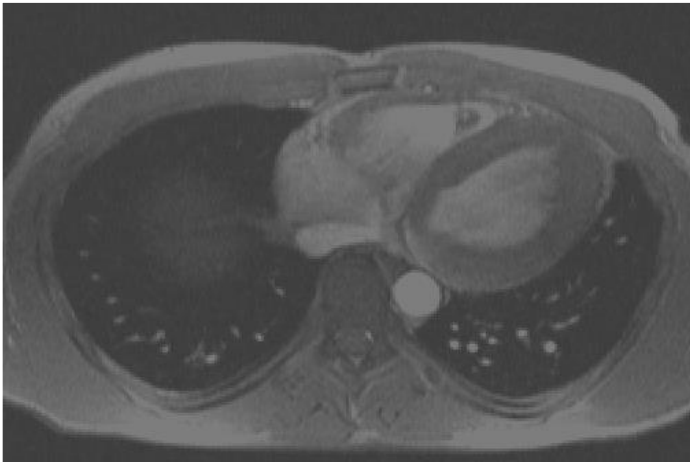
$$a := \frac{q}{\max - \min}$$
$$b := -\frac{q \cdot \min}{\max - \min}$$

$I' \leftarrow \text{AffinePunktoperation}(I, a, b)$ {Algorithmus 8}

Grauwerthistogramm

- Histogrammfunktion: gibt die Häufigkeit eines Merkmals an
Normalerweise: Grauwert

$$H(x) = \#(u, v) : I(u, v) = x, x \in [0..q-1]$$



Berechnung des Grauwertistogramms

Algorithmus 9 BerechneGrauwertHistogramm(I) $\rightarrow H$

for $x := 0$ to q do

$H(x) := 0$

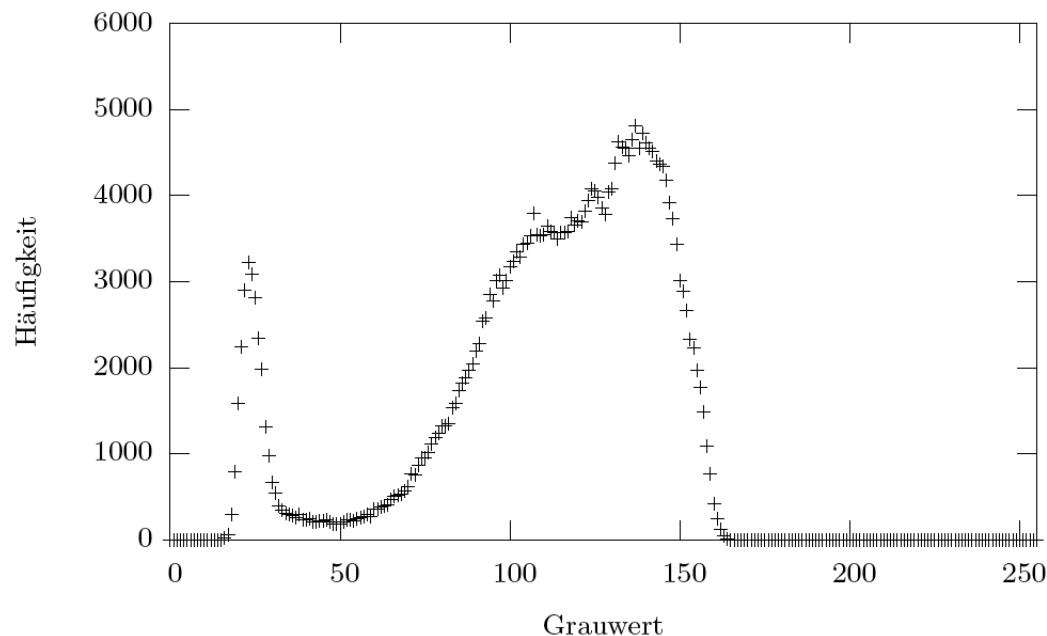
end for

for all pixels (u, v) in I do

$H(I(u, v)) := H(I(u, v)) + 1$

end for

Bei einem 8 Bit Graustufenbild
ist $q = 2^8 - 1 = 255$.



- Die Histogrammdehnung ist eine Verbesserung der Spreizung
- Anstatt der minimalen bzw. maximalen Intensität werden Quantile verwendet, um tatsächliche Maxima im Histogramm zu erkennen
- Die Histogrammdehnung ist eine affine Punktoperation
- Definition der Quantilsfunktion $H_q(p)$ für die Anwendung auf Histogrammen ($p \in [0, 1]$):

$$H_a(x) := \sum_{k=0}^x H(k)$$

$$H_q(p) := \inf \{x \in \{0, \dots, q\} : H_a(x) \geq p \cdot H_a(q)\}$$

Algorithmus 13 BerechneAkkumuliertesHistogramm(H) $\rightarrow H_a$

```
 $H_a(0) := H(0)$   
for  $x := 1$  to  $q$  do  
   $H_a(x) := H_a(x - 1) + H(x)$   
end for
```

Algorithmus 14 BerechneQuantil(H_a, p) $\rightarrow v$

```
for  $x := 0$  to  $q$  do  
  if  $H_a(x) \geq p \cdot H_a(q)$  then  
     $v := x$   
    RETURN  
  end if  
end for
```

Algorithmus 15 HistogrammDehnung($I, p_{min} = 0, 1, p_{max} = 0, 9$) $\rightarrow I'$

```
 $H \leftarrow$  BerechneGrauwertHistogramm( $I$ ) {Algorithmus 9}  
 $H_a \leftarrow$  BerechneAkkumuliertesHistogramm( $H$ ) {Algorithmus 13}  
 $min \leftarrow$  BerechneQuantil( $H_a, p_{min}$ ) {Algorithmus 14}  
 $max \leftarrow$  BerechneQuantil( $H_a, p_{max}$ ) {Algorithmus 14}
```

```
if  $min = max$  then  
  STOP {Bild  $I$  ist homogen}  
end if
```

```
 $a := \frac{q}{max-min}, b := -\frac{q \cdot min}{max-min}$   
 $I' \leftarrow$  AffinePunktoperation( $I, a, b$ ) {Algorithmus 8}
```

- Der Histogrammausgleich erhöht den Kontrast in Bereichen des Histogramms mit vielen Treffern
- Dadurch werden stark vertretene Grauwertbereiche für das Auge sichtbarer gemacht
- Der Histogrammausgleich ist eine homogene Punktoperation, jedoch keine affine Punktoperation
- Nachteil:
Obwohl der Histogrammausgleich in der Praxis in der Regel den Kontrast erhöht, kann er auch zu einer Verminderung des Kontrasts führen (siehe Übung)

Algorithmus 16 HistogrammAusgleich(I) $\rightarrow I'$

$H \leftarrow \text{BerechneGrauwertHistogramm}(I)$ {Algorithmus 9}

$H_a \leftarrow \text{BerechneAkkumuliertesHistogramm}(H)$ {Algorithmus 13}

for $x := 0$ **to** q **do**

$H_n(x) := \text{round}(\frac{q \cdot H_a(x)}{H_a(q)})$

end for

for all pixels (u, v) in I **do**

$I'(u, v) := H_n(I(u, v))$

end for

Beispiele zur Kontrastanpassung

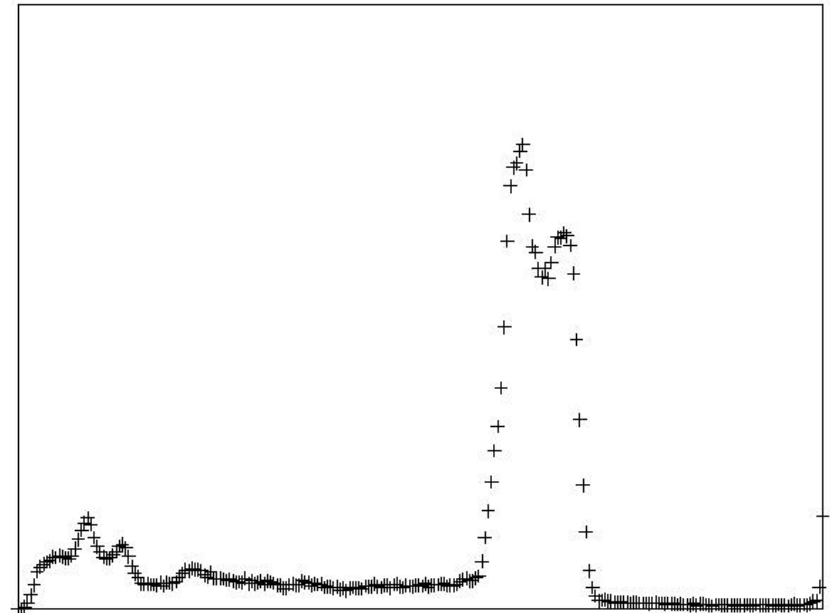
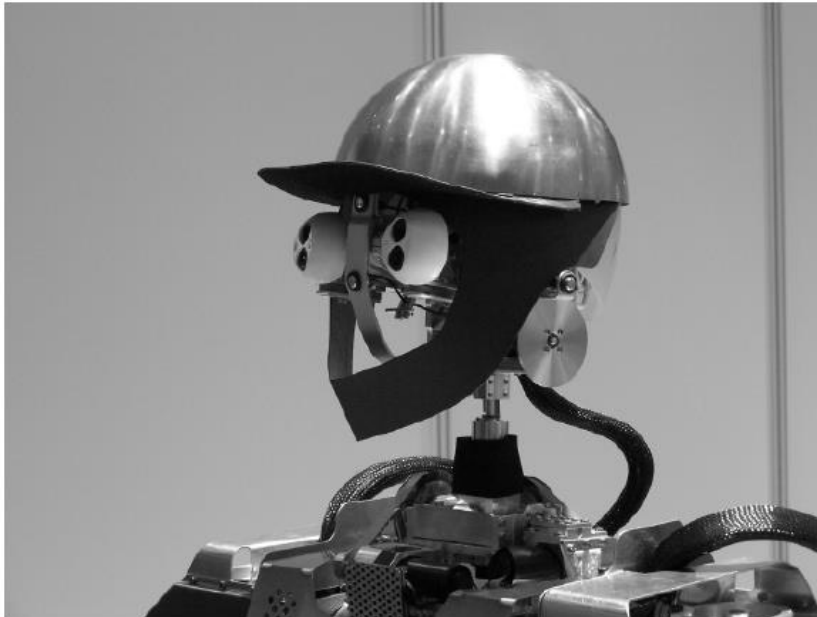
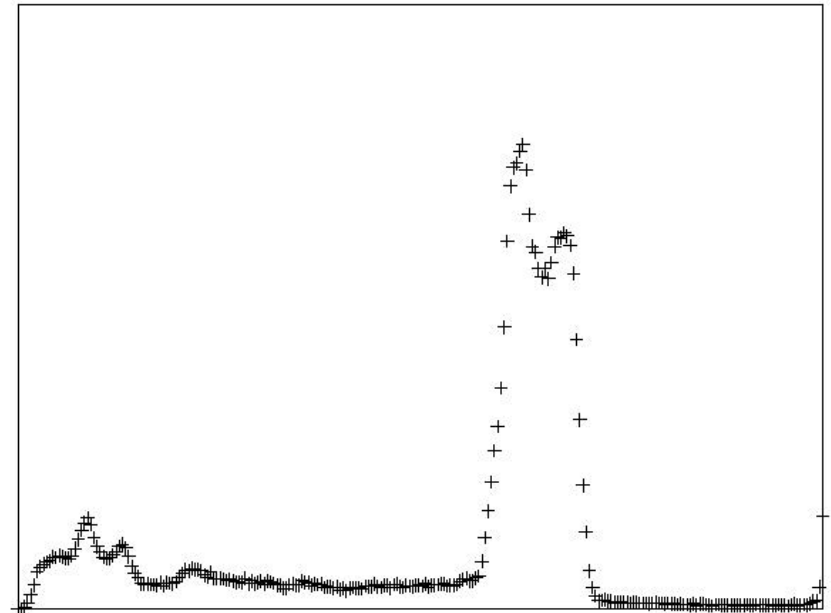
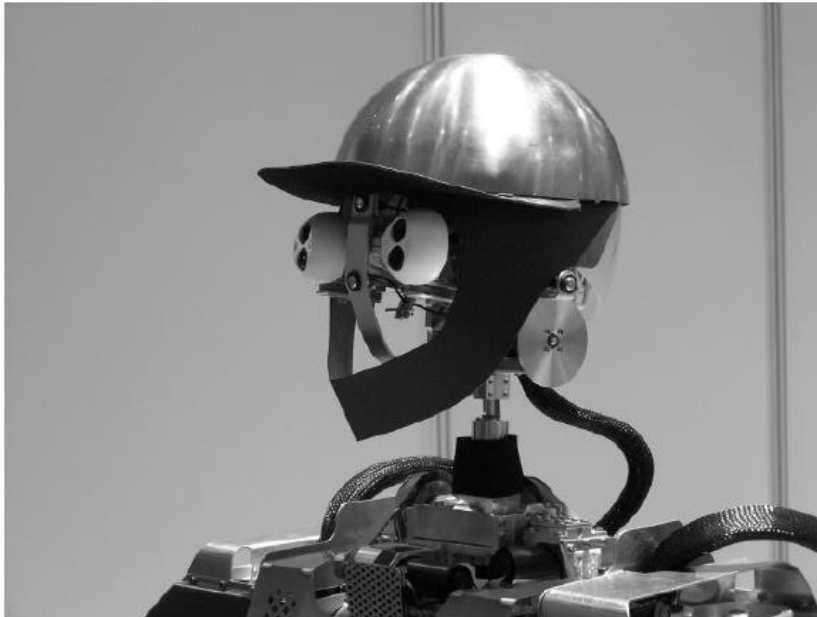


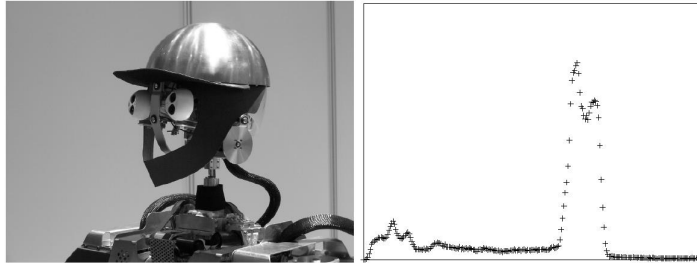
Bild des Kopfes des humanoiden Roboters ARMAR III.
Links: Eingabebild. Rechts: Grauwerthistogramm.

Beispiel Spreizung

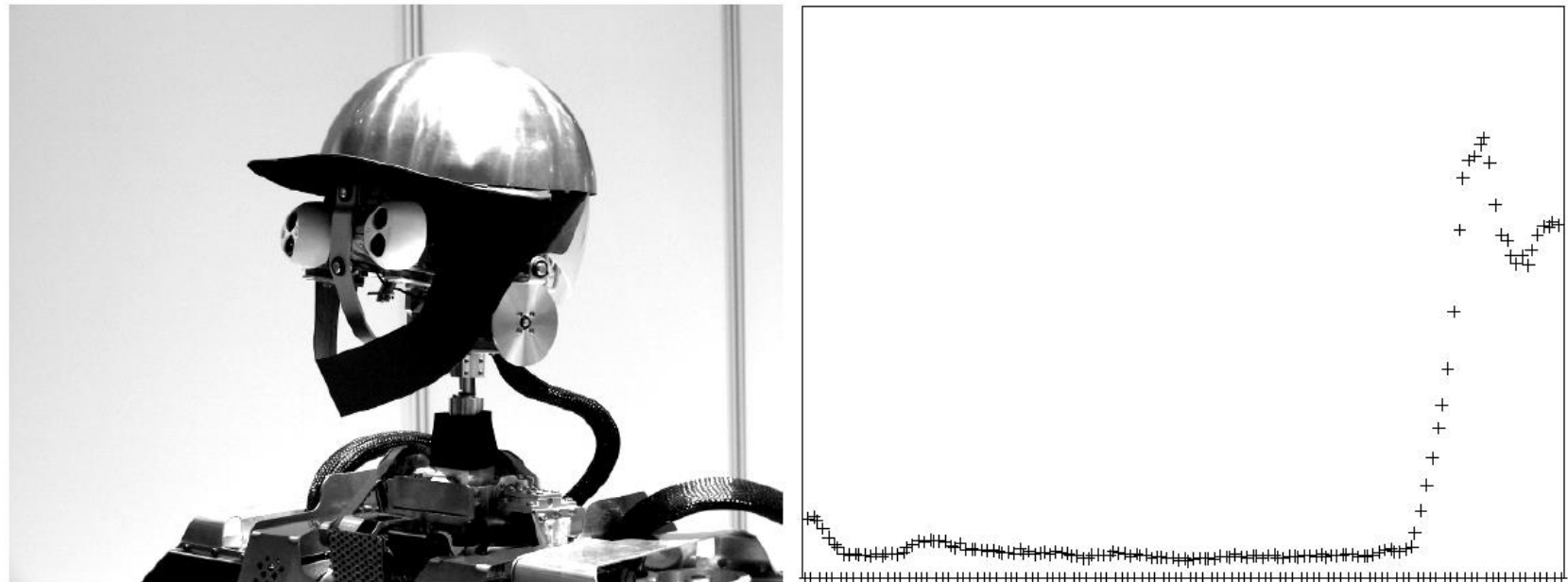


Ergebnis der Spreizung (identisch mit Eingabebild, da sowohl Intensität 0 als auch Intensität 255 vorhanden).
Links: Ergebnisbild. Rechts: Grauwerthistogramm.

Beispiel Histogrammdehnung

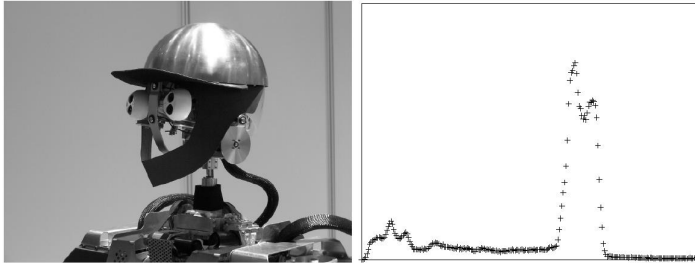


Eingabebild mit
Grauwerthistogramm.

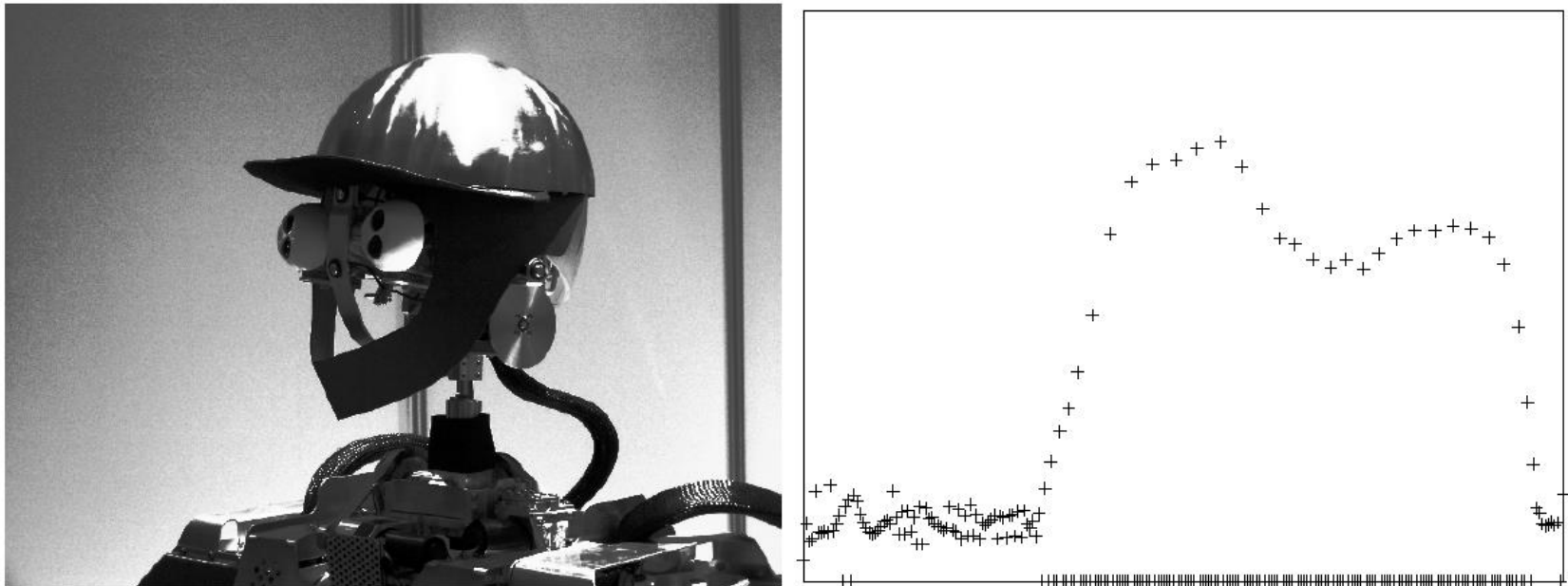


Ergebnis einer Histogrammdehnung mit $p_{\min} = 0,1$ und $p_{\max} = 0,9$.
Links: Ergebnisbild. Rechts: Grauwerthistogramm.

Beispiel Histogrammausgleich



Eingabebild mit
Grauwerthistogramm.



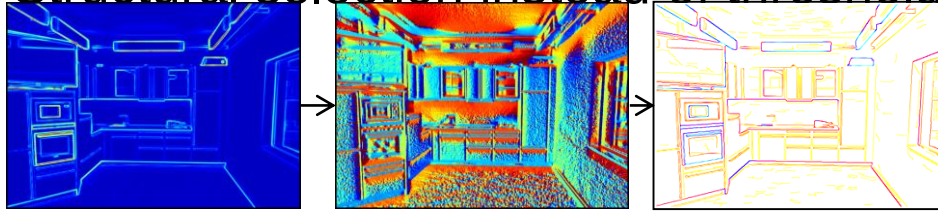
Ergebnis des Histogrammausgleichs.
Links: Ergebnisbild. Rechts: Grauwerthistogramm.

Visual Feature Extraction

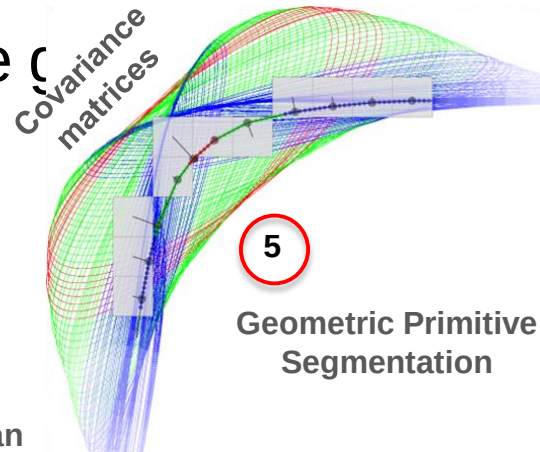
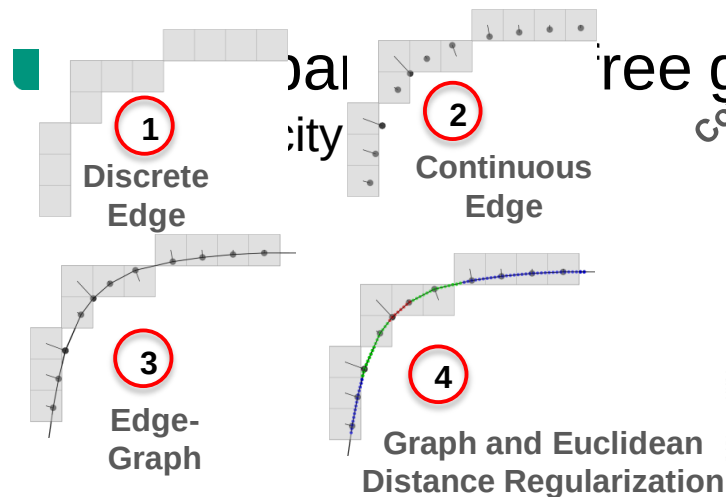
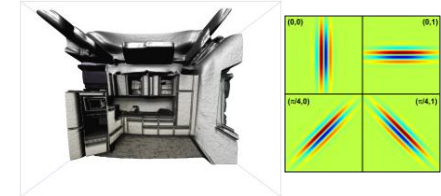
Edge extraction [Grigorescu et al. 2006]

Gabor kernels

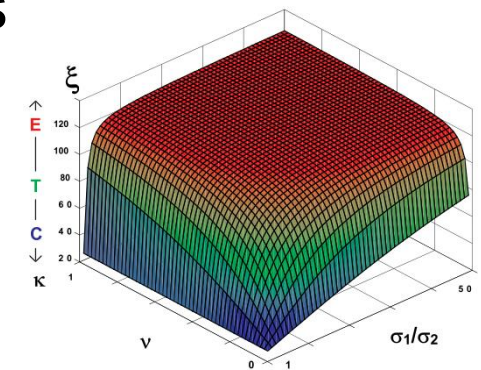
Structural selection instead of thresholds



$$\Gamma(\mathbf{x}) = \sum_{p=0}^n \sum_{q=0}^1 U_{\langle p\pi/2n \rangle} \left\{ \Phi(\mathbf{x}) \otimes \Psi^{(p\pi/2n, q)} \right\}$$



Geometric Primitive Segmentation



Eccentricity Segmentation Model

Nächste Vorlesung:

Mittwoch, 3. Mai 2016

Bildverarbeitung 2: Filtertheorie