

Bildmerkmalssuche

Seminar Computational Photography



EINFÜHRUNG



Einführung

- Bildmerkmalssuche: sehr wichtiges Thema des künstlichen Sehens
- Erkennen von Objekten auf dem Bild oder in einer Bildsequenz anhand von markanten Regionen



Anwendungsbeispiele

Autos: automatisches Erkennen von Verkehrszeichen



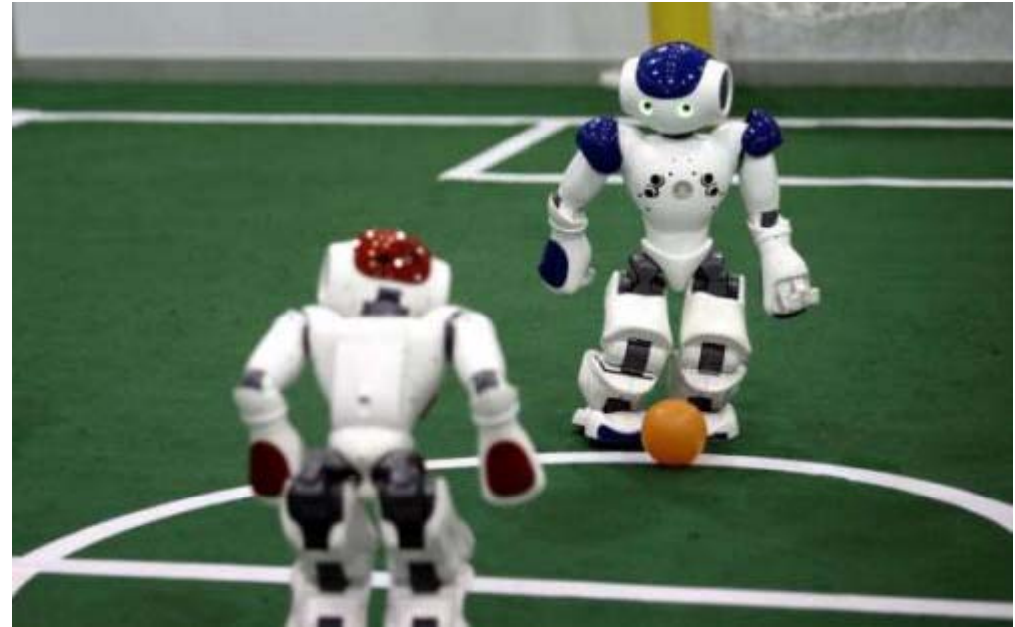
Anwendungsbeispiele

Nao Roboter:
Gesichtserkennung.



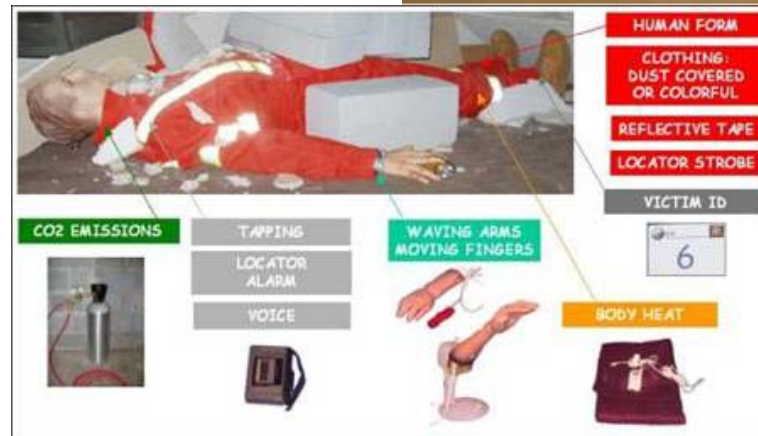
Anwendungsbeispiele

- **RoboCup Soccer League:**
Roboter spielen Fußball
- Roboter müssen in der Lage sein:
 - sich völlig autonom auf dem Spielfeld zu orientieren
 - den Ball und die anderen Roboter zu finden



Anwendungsbeispiele

- **RoboCup Rescue League:** Rettungsroboter
- Roboter müssen autonom agieren:
 - Durchqueren einen Parcours aus Hindernissen
 - Finden simulierter Opfer und Einzeichnen in der von dem Roboter erstellten Karte



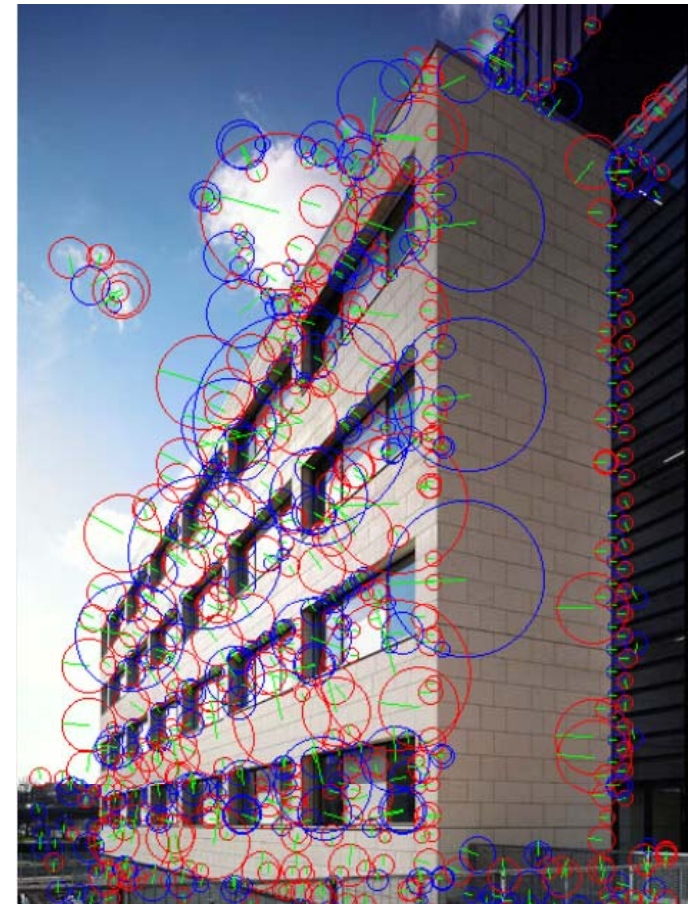
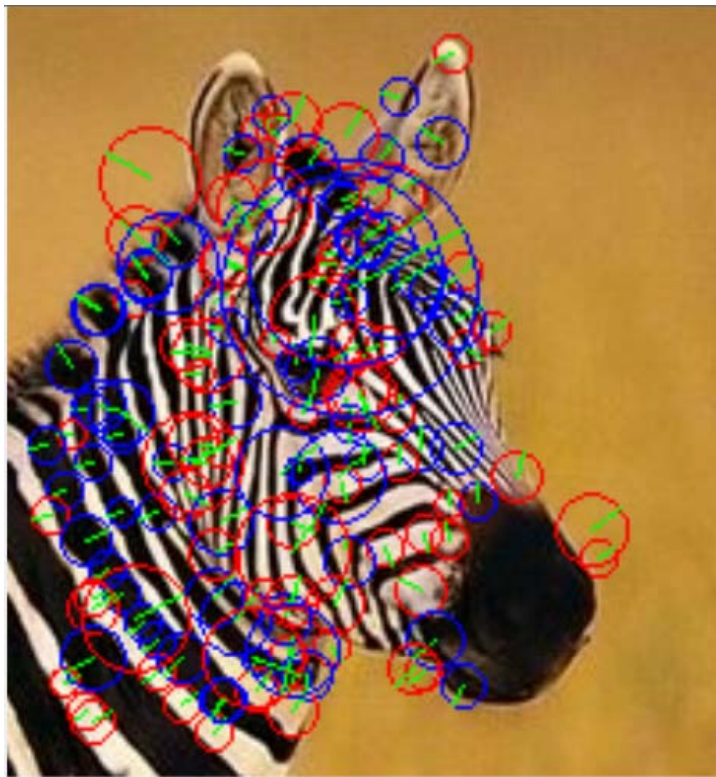
Was sind Bildmerkmale?

- Bildmerkmale (lokale) bezeichnen Bildmuster, die sich von ihrer dichten Umgebung unterscheiden und üblicherweise mit einer Änderung von einer oder mehreren Bildeigenschaften assoziiert sind.
- Beispiele:
 - weiße Flecken auf dunklem Hintergrund (Kontraständerung)
 - Kanten, Ecken (gut strukturierte Objekte)
 - Regionen, deren Farbe sich von anderen Objekten sehr gut unterscheiden lässt



Motivation

- Bildmerkmalssuche: Problemstellung:
 - Finden repräsentativer Bildmerkmale



Motivation

- Bildmerkmalssuche: Problemstellung:
 - Finden repräsentativer Bildmerkmale
 - Zuordnen dieser mit denen eines anderen Bilds



Motivation

- Bildmerkmalssuche: Problemstellung:
 - Finden repräsentativer Bildmerkmale
 - Zuordnen dieser mit denen eines anderen Bilds
- Bildmerkmalssuche: Lösung:
 - Markante Regionen in Bilddaten finden – **Detektionsschritt**
 - Eigenschaften dieser Regionen in kompakter Form beschreiben – **Deskriptor bilden**
 - Anhand dieser Eigenschaften Korrespondenzen zwischen verschiedenen Bildern des selben Objektes bestimmen – **Matching**



Algorithmen: Eigenschaften

- Graustufenbilder oder Farbbilder
- Invarianz gegen:
 - Affine Transformation (Verschiebung, Drehung)
 - Perspektivische Transformation
 - Koordinatentransformation (Translation, Skalierung, Rotation)
 - Photometrische Transformation
- Lokale oder globale Merkmale
- Laufzeit
- Matching zur Laufzeit oder während der Lernphase



SURF

ALGORITHMEN



SURF: Allgemeines

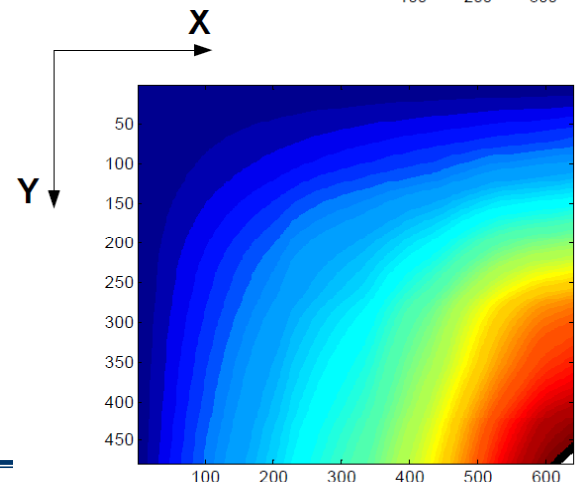
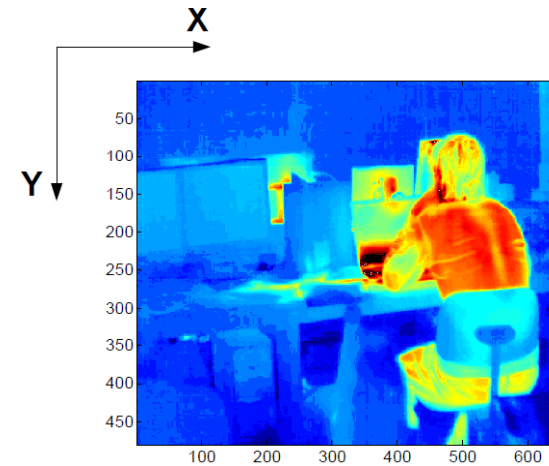
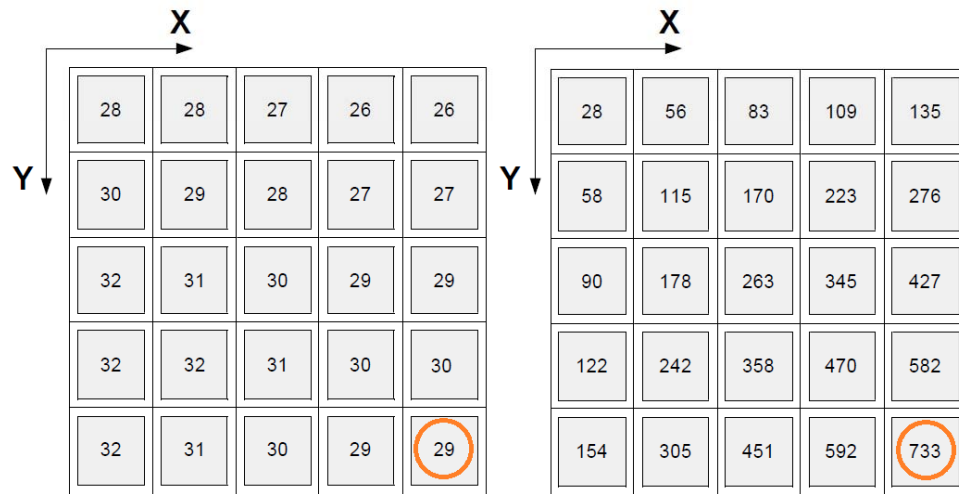
- SURF – *S*peeded *U*p *R*obust *F*eatures
- Nachfolger von SIFT
- Key-Point basierter Algorithmus (lokale Bildmerkmale)
- Verwendung von Integral Images
- Detektion von Key-Points mit Hilfe eines Hesse-Matrix basierten Feature-Detektors
- Skalen- und rotationsinvariante Detektion
- Die Key-Points werden durch einen Eigenschaftsvektor mit 64 Einträgen charakterisiert



Integral Image Darstellung

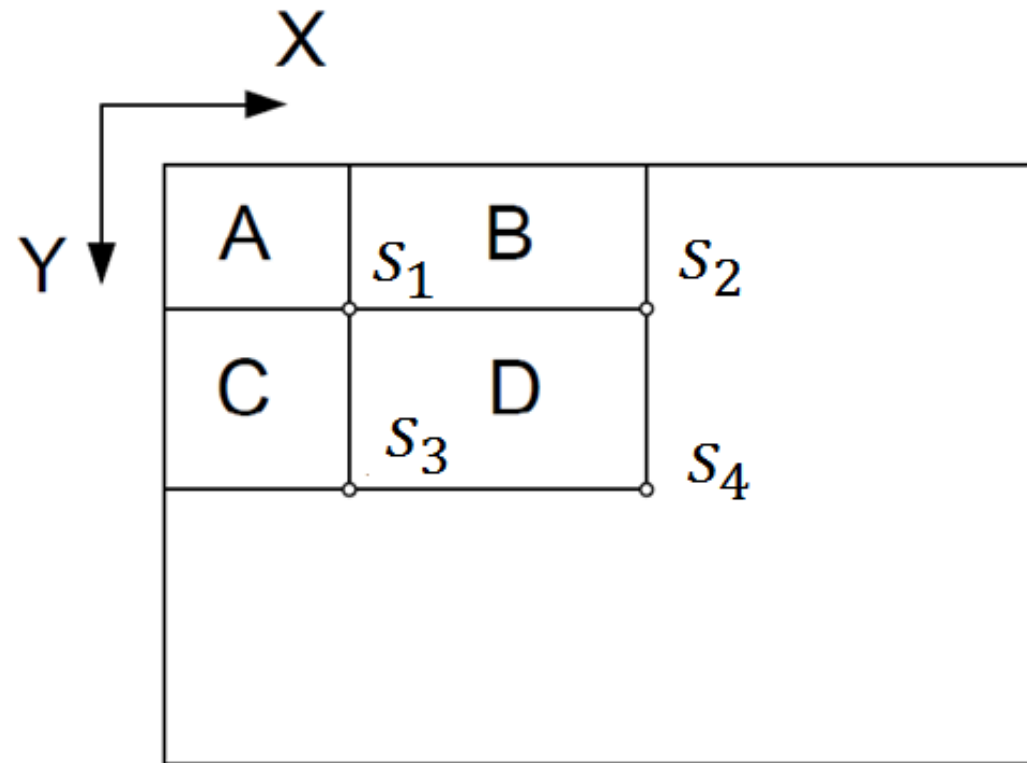
- Es wird die Summe der Intensitätswerte der Pixel eines Bildes die links bzw. oberhalb des Pixels an Position (x, y) liegen.

$$II(x, y) = \sum_{t \leq x, j \leq y} I(x, y)$$



Integral Image Darstellung

- Wesentlicher Vorteil: Jede Flächensumme kann nur mit drei Additionen und vier Speicherzugriffen gefunden werden.



$$D = s_4 - s_3 - s_2 + s_1$$

Feature-Detektion: Fast-Hessian Detektor

- Die Hessematrix enthält die zweiten partiellen Ableitungen der Bildintensitäten in x- und y-Richtung

$$H = \begin{bmatrix} I_{xx} & I_{xy} \\ I_{xy} & I_{yy} \end{bmatrix}$$

- Kern-Charakteristiken:

$$\text{spur}H = I_{xx} + I_{yy} \quad \text{det}H = I_{xx}I_{yy} - I_{xy}^2$$

- Die Maxima von den beiden Werten entsprechen Blob-ähnlichen Strukturen im Bild.
- Die partiellen Ableitungen können durch Faltung des Bildes mit geeigneten Filter-Masken berechnet werden.



Feature-Detektion: Fast-Hessian Detektor

- Zweite Ableitungen von den Intensitätswerten sind gegen Bildrauschen sehr empfindlich → Bilder mit Gauß-Filter glätten
- Gleichzeitig Invarianz gegen Skalierung
- Erweiterte Hessematrix:

$$H(\sigma) = \begin{bmatrix} L_{xx}(\sigma) & L_{xy}(\sigma) \\ L_{xy}(\sigma) & L_{yy}(\sigma) \end{bmatrix}$$

- $L(\sigma)$ ist die Faltung des Bildes (Bildintensität) mit dem Glättungskern $G(\sigma)$

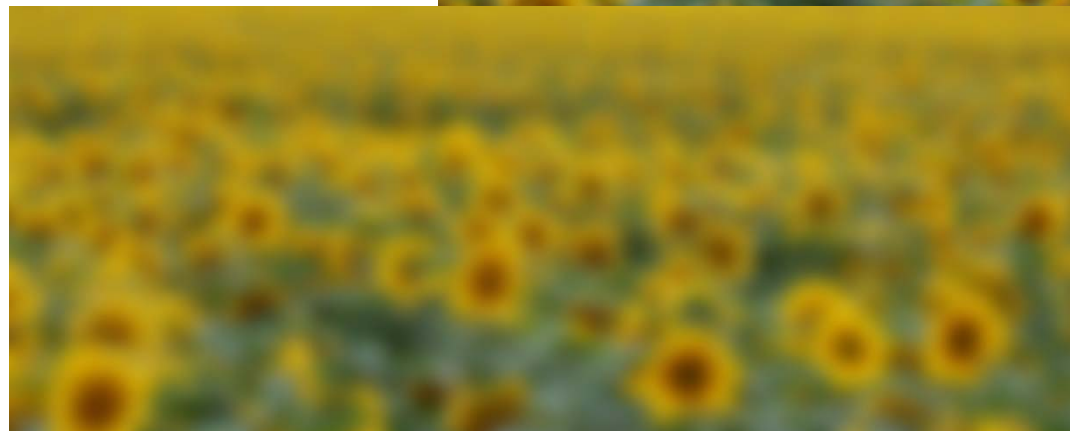
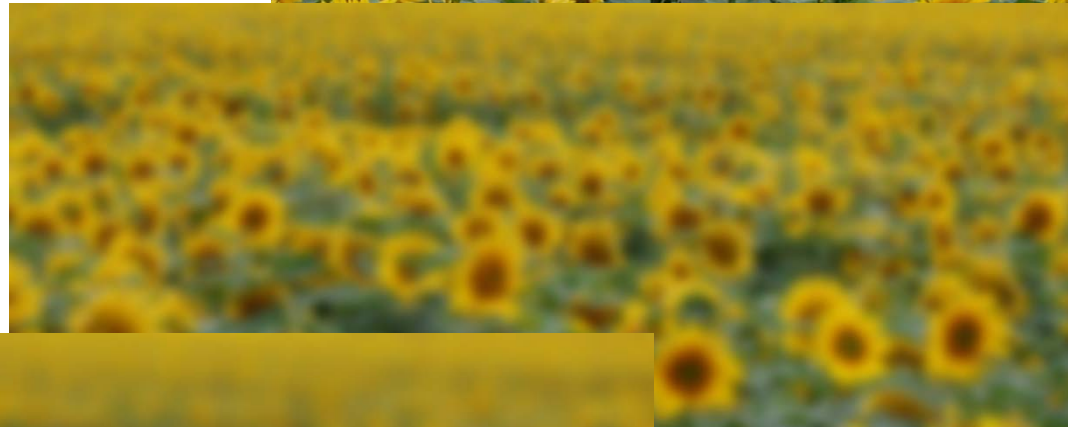


Feature-Detektion: Fast-Hessian Detektor

Originalbild, $\sigma = 0$



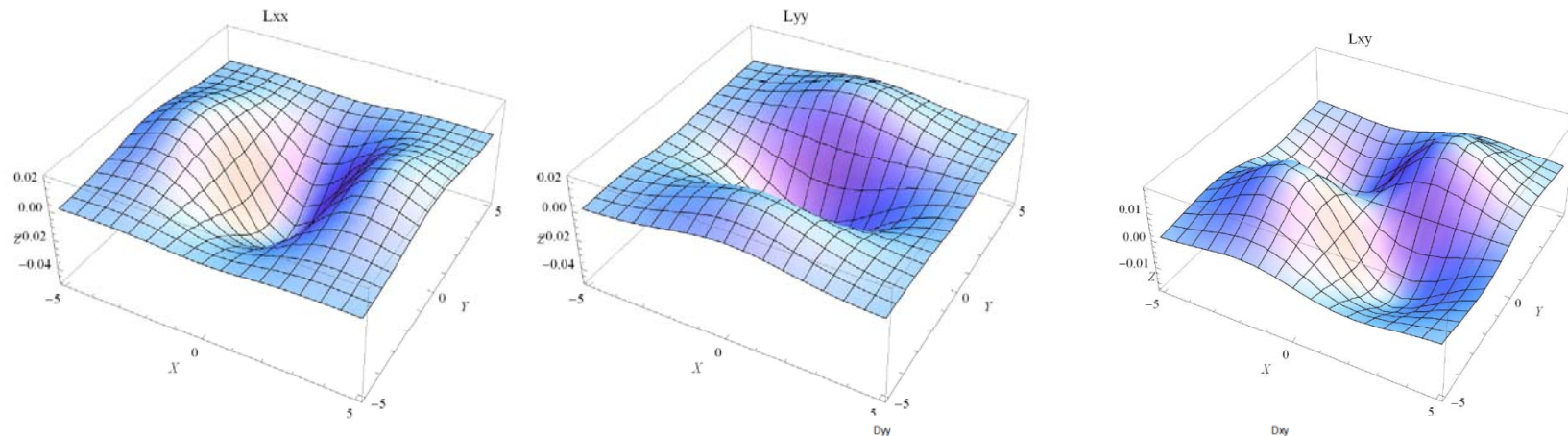
Gauß-Filter, $\sigma = 1,2$



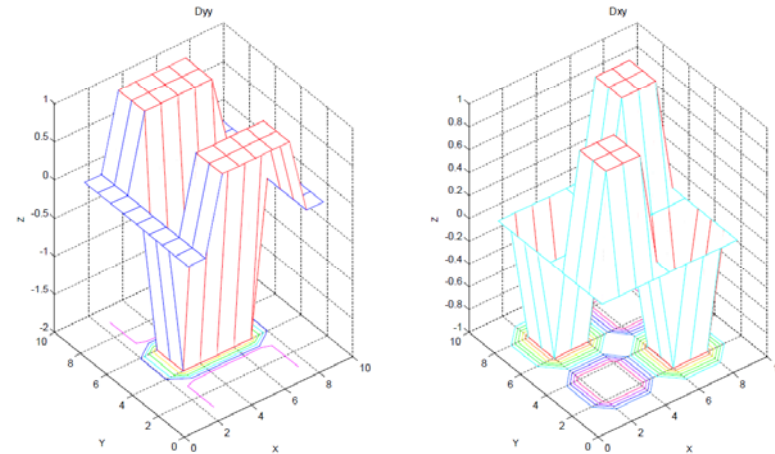
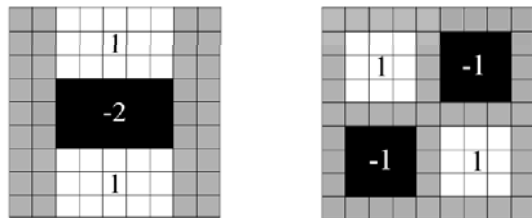
Gauß-Filter, $\sigma = 2,0$

Feature-Detektion: Fast-Hessian Detektor

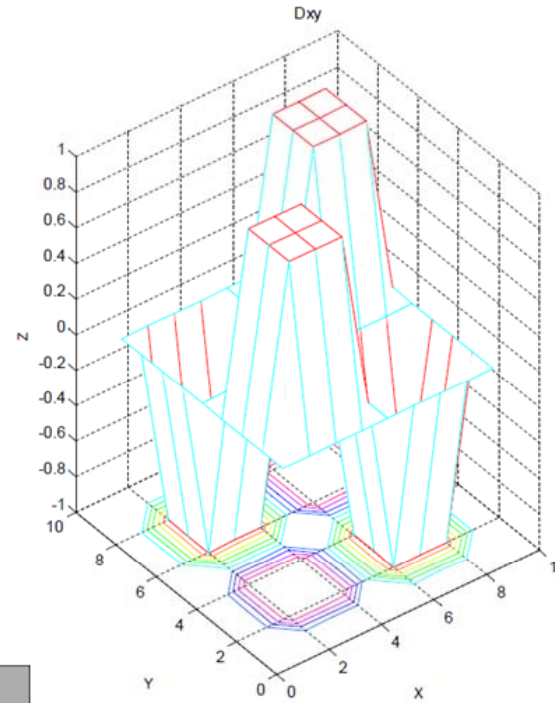
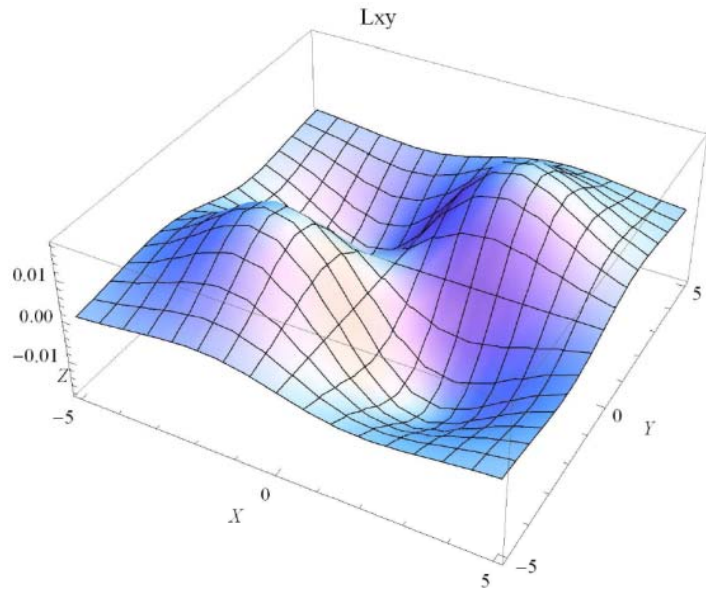
- Bei SURF werden die zweiten Ableitungen von $L(\sigma)$ durch Faltung des Bildes mit Filter-Masken gewonnen.



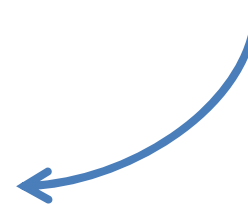
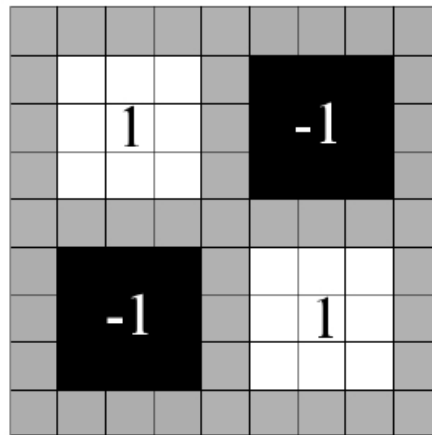
Als Filter aber in dieser Form nicht anwendbar → werden approximiert und diskretisiert.



Feature-Detektion: Fast-Hessian Detektor

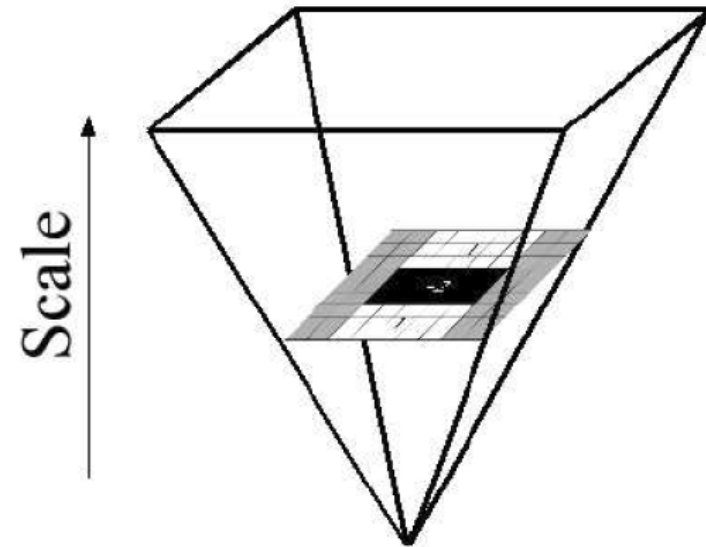


SURF Box-Filter



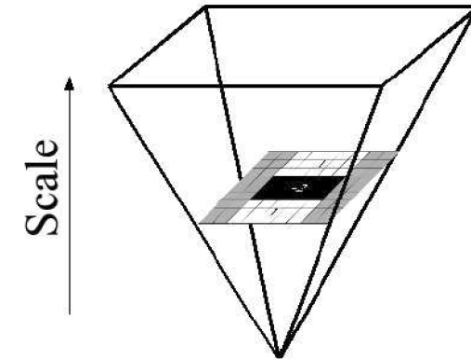
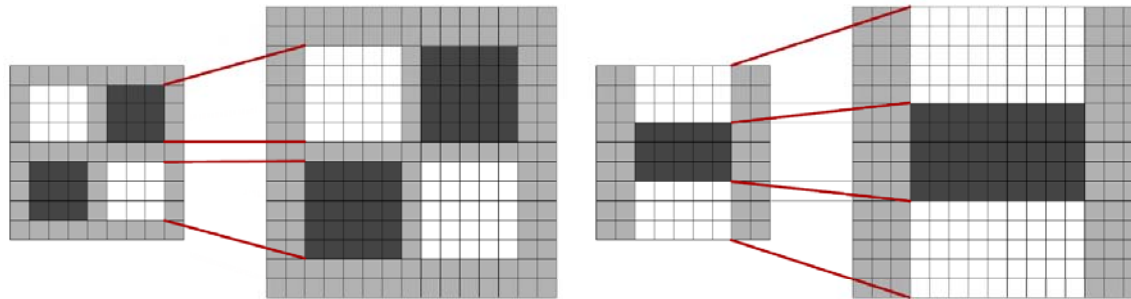
Feature-Detektion: SURF-Filterpyramide

- Ziel: Invarianz gegen Skalierung.
- Es wird aber nicht das Bild wiederholt mit Gauß-Filter geglättet und dann verkleinert.
- Filtermatrizen werden vergrößert und auf das original Bild angewandt (Integral-Image + Box-Filter)
- Vorteil: sehr effizient und keine Aliasing-Effekte



Feature-Detektion: SURF-Filterpyramide

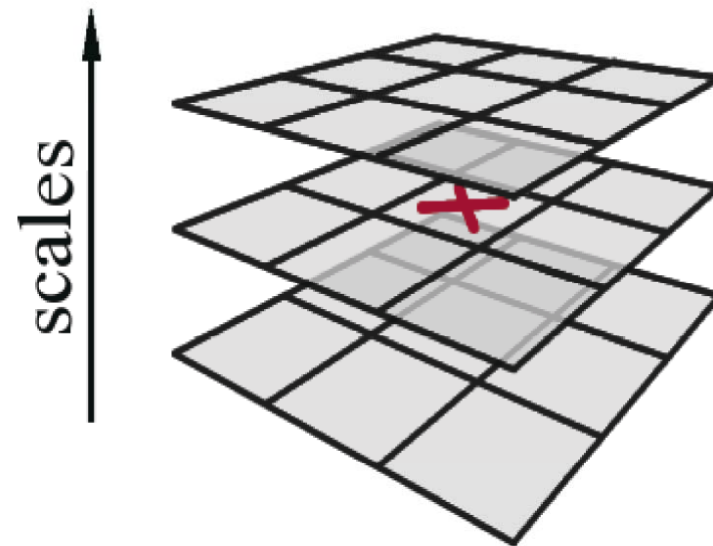
- Minimale Filtergröße: 9*9 Pixel ($\sigma = 1.2$)
- Skalen-Schritt 6 Pixel ($\Delta\sigma = 0.8$)



- Skalenraum wird in Oktaven unterteilt.
- Jede Oktave besteht aus 4 Bildern mit unterschiedlichen Skalen.
- Erste Oktave: 9*9, 15*15, 21*21, 27*27
- Zweite Oktave: 15*15, 27*27, 39*39, 51*51

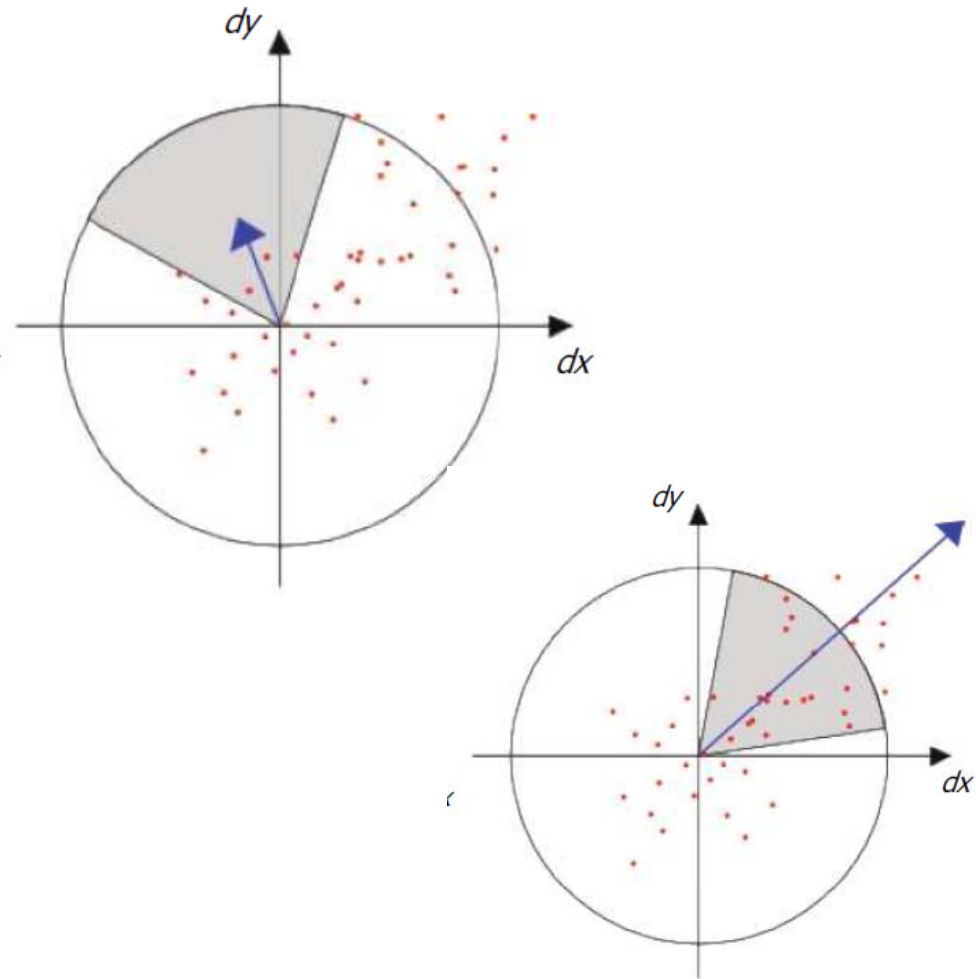
Feature-Detektor: Lokalisierung der Schlüsselpunkte

- Die Determinante der Hessematrix für jeden Punkt auf dem Bild wird bestimmt.
- Der mögliche Schlüsselpunkt wird mit 26 Nachbarn verglichen.
- Positionen der Punkte, die über alle Skalen markant sind, werden gespeichert.
- So wird die Invarianz gegen Skalierung gewährleistet.



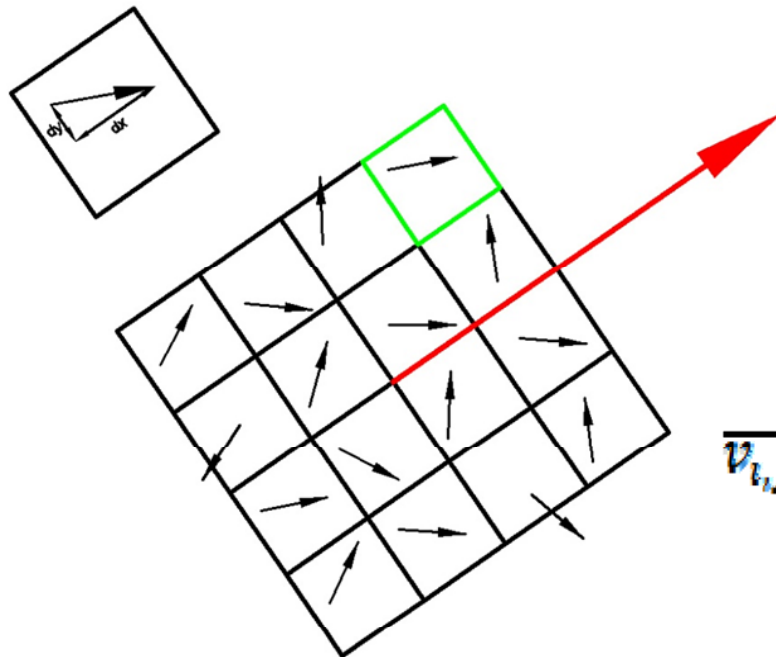
Interest Point Descriptor: Orientierung

- Ziel: Rotationsinvarianz
- Es werden die Gradienten in einer kreisförmigen Region ($r = 6\sigma$) um den Schlüsselpunkt berechnet.
- Dann werden sie in x- und y-Richtung approximiert, abgetastet und gewichtet.
- Es wird ein $\pi/3$ großes Fenster durch die Region verschoben.
- In jedem Fenster wird die Summe der Punkte gebildet.
- Der längste Vektor wird als Orientierung genommen.



Bestimmung des Descriptors

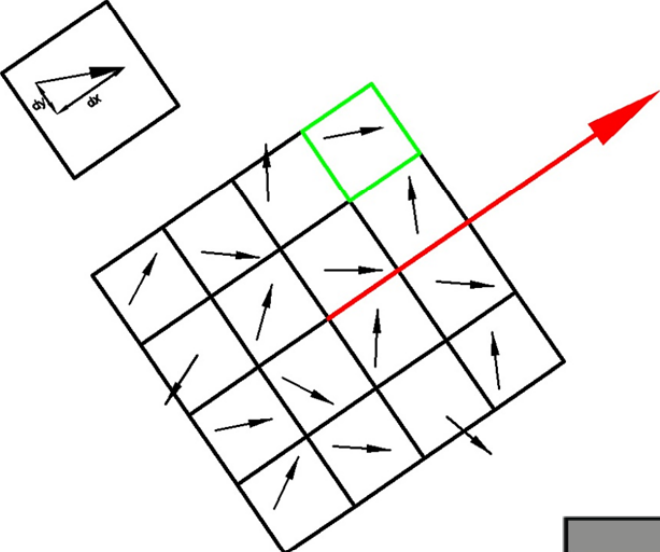
- Über den Schlüsselpunkt wird ein Quadrat (Seite = 20σ) gelegt ausgerichtet nach der Orientierung des Schlüsselpunkts.
- Das Quadrat wird in 16 Regionen unterteilt.
- Für jede Region wird der Gradientenverlauf in x- und y-Richtung für 25 Punkte berechnet.



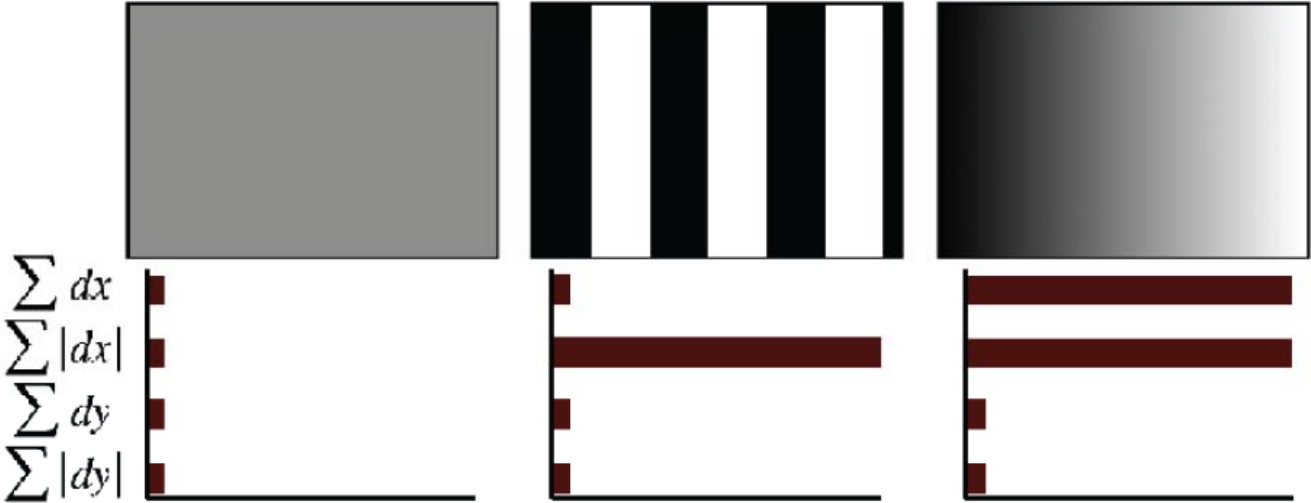
Dann wird für jede Region ein Descriptor gebildet:

$$\vec{v}_{i,j} = \sum_{f=1}^{25} (d_x(t,f), d_y(t,f), |d_x(t,f)|, |d_y(t,f)|)$$

Bestimmung des Descriptors

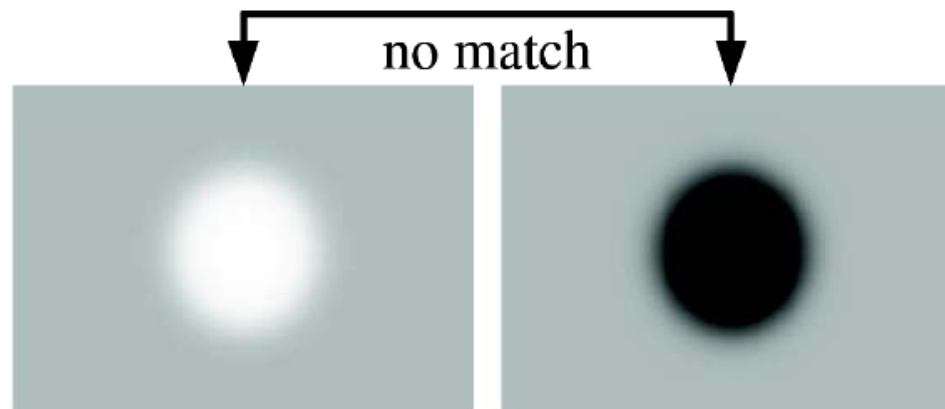


$$\vec{v}_{i,j} = \sum_{f=1}^{25} (d_x(t,f), d_y(t,f), |d_x(t,f)|, |d_y(t,f)|)$$



SURF: Matching

- Spur der Hessematrix
 - Ist bereits vorhanden: wurde beim Detektieren schon berechnet
 - weist auf den Kontrast der Umgebung im Vergleich zum Schlüsselpunkt hin
 - Punkte, bei denen keine Übereinstimmung gefunden wird, werden ausgeschlossen.



SURF: Matching

- Nearest Neighbour Ratio:
 - Zu jedem Schlüsselpunkt mit dem Deskriptor \mathbf{v} (Originalbild) werden zwei Schlüsselpunkte mit den Deskriptoren \mathbf{v}_1 und \mathbf{v}_2 (Objektbild) gesucht, die den kleinsten euklidischen Abstand zu \mathbf{v} haben.
 - Der Quotient aus diesen beiden Abständen wird als *Nearest Neighbour Ratio* bezeichnet.
 - Ist er kleiner als ein bestimmter Schwellenwert, wird dem Punkt mit Deskriptor \mathbf{v} dem Punkt mit dem Deskriptor \mathbf{v}_1 zugeordnet.
 - Ziel: sicherzustellen, dass \mathbf{v}_1 eindeutiger Matching-Punkt ist.

$$\frac{|\mathbf{v} - \mathbf{v}_1|}{|\mathbf{v} - \mathbf{v}_2|} \geq \textit{threshold}$$



SURF: Beispielbilder



1180
Schüsselpunkte
gefunden



SURF: Beispielbilder



73 Schüsselpunkte
gematcht

SURF: Zusammenfassung

- Deutliche Effizienzsteigerung im Vergleich zu SIFT:
 - Integral Images, Box-Filter, Filterpyramide, nur 64 Einträge im Descriptor
- Invarianz gegen Skalierung:
 - Filterpyramide, markante Regionen werden über allen Skalen gesucht
- Invarianz gegen Rotation:
 - Gleitendes Fenster, jedem Schlüsselpunkt wird die Orientierung zugewiesen.
- Vorteile:
 - Robust, relativ schnell
- Nachteile:
 - Closed Source, immer noch SEHR rechenintensiv



Bildmerkmalssuche: Andere Algorithmen

- SIFT
- Punktdetektor von Harris
- Randomized Trees
- Neuronale Netze
- ...



Quellen:

- Herbert Bay, Andreas Ess, Tinne Tuytelaars, Luc van Gool: *Speeded-Up Robust Features (SURF)*
- *Christopher Evans: Notes on the OpenSURF Library*
- C. Harris, M. Stephens: *A combined corner and edge detector*
- Vincent Lepetit, Pascal Fua: *Keypoint Recognition using Randomized Trees*
- David G. Lowe: *Distinctive Image Features from Scale-Invariant Keypoints.*

