

Intelligente Sehsysteme

9 Modellbasierte Segmentierung

Template Matching,
Hough-Transformation

Volker Steinhage

Inhalt

- Modellbasierte Segmentierung
 - Template Matching
 - Hough-Transformation
 - HAT für gerade Liniensegmente
 - CHT für kreisförmige Segmente

Modellbasierte Segmentierung (1)

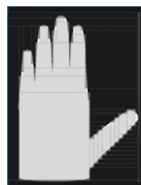
- Modellbasierte Segmentierungsansätze
 - nutzen eine auf Bilddaten anwendbare **Repräsentation eines Modells**, welches eine gesuchte Klasse bzw. gesuchten Klassen von Szenenobjekten beschreibt,
 - können so ggf. Segmentierung und Objekterkennung kombinieren
- Hier werden vorgestellt:
 - Template-Matching
 - Hough-Transformation

Template Matching (1)

Beim Template Matching wird das gesuchte Segment

- durch ein Muster (engl. *Template*) in seiner Form und Orientierung spezifiziert
- nicht aber in seiner Position spezifiziert!

Template



Eingabebild



Bildquelle: <http://cgvr.informatik.uni-bremen.de/publications.shtml> (14.12.2012)

Template Matching (2)

- Ein **Muster ist ein zweidim. Feld $t(p,q)$** mit $p = 0, \dots, P-1$, $q = 0, \dots, Q-1$, mit
 - der gesuchten Struktur

und

 - dem Hintergrund in unmittelbarer Nähe
- **Gesucht sind Positionsparameter x_k und y_k** im Eingabebild $I[x,y]$, so dass der Bildausschnitt $I_{PQ}(x_k, y_k)$ mit $x = x_k, \dots, x_k + P-1$ und $y = y_k, \dots, y_k + Q-1$ mit dem Muster t optimal übereinstimmt

Template



Eingabebild



Bildquelle: <http://cgvr.informatik.uni-bremen.de/publications.shtml> (14.12.2012)

Abstandsmaße (1)

- Zwei häufig verwendete Maße für diese Übereinstimmung sind
 - der durchschnittliche quadratische Abstand (*mean square distance – msd*):

$$msd(x_k, y_k) = \frac{1}{PQ} \sum_{p=0}^{P-1} \sum_{q=0}^{Q-1} (I(x_k + p, y_k + q) - t(p, q))^2$$

- der durchschnittliche Abstand der Beträge (*mean absolute distance – mad*):

$$mad(x_k, y_k) = \frac{1}{PQ} \sum_{p=0}^{P-1} \sum_{q=0}^{Q-1} |I(x_k + p, y_k + q) - t(p, q)|$$

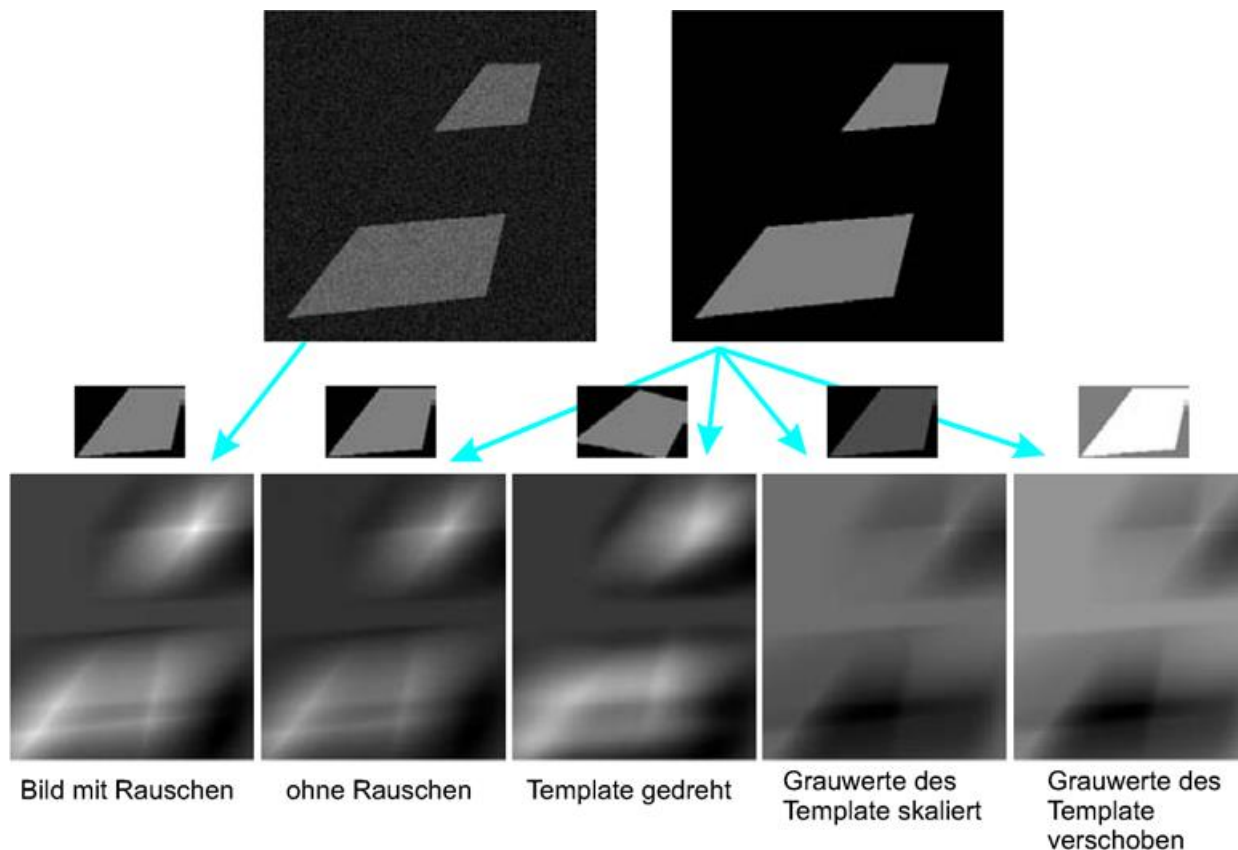
Abstandsmaße (2)

- Selbst mit diesen einfachen Abstandsmaßen ist Template Matching robust gegenüber Rauschen
- Die Ergebnisse werden aber schlechter,
 - wenn sich die durchschnittl. Helligkeiten zw. Bild und Muster unterscheiden
 - wenn sich der Kontrast zwischen zw. Bild und Muster unterscheidet
- Damit muss das Muster sehr genau an das Bild bzw. die Bildklasse angepasst werden, um sinnvoll genutzt werden zu können (s. folg. Bspl.)

Abstandsmaße (3)

- Beispiel: die durchschnittl. Grauwertdifferenz (hier $mad(x_k, y_k)$) zwischen Muster und Bildausschnitt ist als Maß für ein Template Matching besonders anfällig gegenüber **Skalierung oder Verschiebung der Grauwerte**.

Rauschen dagegen beeinflusst wegen der Integration der Werte über die Fläche des Musters das Resultat kaum. **Rotationen** zw. Musters und Bild führt zu einer vom Rotationswinkel abhängigen Verschlechterung des Resultats.



Bildquelle: Klaus
Tönnies: Grund-
lagen der Bildver-
arbeitung, Pearson
Studium, 2005.

Abstandsmaße (4)

- Eine größere Robustheit des Template Matching gegenüber Verschiebung und Skalierung der Intensitätswerte ergibt sich, wenn anstelle der Abstandsmaße zwischen den Intensitätswerten deren Korrelationskoeffizient berechnet wird.
- Der **Korrelationskoeffizient** $cc_{f,g}$ zwischen zwei Verteilungen f und g ist gegeben
 - über die **Kovarianz** $\sigma_{f,g}$
 - über die **Varianzen** σ_f^2 und σ_g^2
 - als $cc_{f,g} = \sigma_{f,g} / (\sigma_f^2 \cdot \sigma_g^2)^{1/2}$
- Genauer:
 - zunächst werden der Bildausschnitt $I_{PQ}(x_k, y_k)$ und das Muster $t(p, q)$ für die beiden Verteilungen f und g gesetzt.

Abstandsmaße (5)

- Zum Korrelationskoeffizient (Forts. 1):
 - Eine Schätzung der Ähnlichkeit zwischen Bildausschnitt $I_{PQ}(x_k, y_k)$ und Muster $t(p, q)$ ergibt sich aus der Kovarianz $\sigma_{I,t}$ zwischen $I_{PQ}(x_k, y_k)$ und $t(p, q)$:

$$\sigma_{I,t}(x_k, y_k) = \frac{1}{PQ} \sum_{p=0}^{P-1} \sum_{q=0}^{Q-1} \left(\left(I(x_k + p, y_k + q) - \overline{I(x_k + p, y_k + q)} \right) \left(t(p, q) - \overline{t(p, q)} \right) \right).$$

- Die überstrichenen Werte stehen für die Mittelwerte im Bildausschnitt $I_{PQ}(x_k, y_k)$ und im Muster $t(p, q)$.

Abstandsmaße (6)

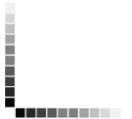
- Zum Korrelationskoeffizient (Forts. 2):

$$\sigma_{I,t}(x_k, y_k) = \frac{1}{PQ} \sum_{p=0}^{P-1} \sum_{q=0}^{Q-1} \left((I(x_k + p, y_k + q) - \overline{I(x_k + p, y_k + q)}) (t(p, q) - \overline{t(p, q)}) \right).$$

- Bei hoher positiver Kovarianz $\sigma_{I,t}$ sind $I_{PQ}(x_k, y_k)$ und $t(p, q)$ bis auf eine additive Komponente und einen Skalierungsfaktor ähnlich.
- Bei hoher negativer Kovarianz $\sigma_{I,t}$ ist $I_{PQ}(x_k, y_k)$ eine bzgl. der Intensitätsverteilung invertierte Variante von $t(p, q)$. Wo $I_{PQ}(x_k, y_k)$ dunkel ist, ist $t(p, q)$ hell und umgekehrt.
- Bei Kovarianz $\sigma_{I,t}$ gleich Null sind $I_{PQ}(x_k, y_k)$ und $t(p, q)$ nicht ähnlich.

Abstandsmaße (7)

- Zum Korrelationskoeffizient (Forts. 3):
 - Die mögliche Kovarianz $\sigma_{I,t}$ zwischen Bildausschnitt $I_{PQ}(x_k, y_k)$ und Muster $t(p, q)$ lässt sich durch ein 2D-Histogramm $H(I_{PQ}, t)$ visualisieren
 - Die beiden Achsen von $H(I_{PQ}, t)$ geben die Intensitätswerte von $I_{PQ}(x_k, y_k)$ und $t(p, q)$ wieder
 - Jeder Eintrag $H(I_1, I_2)$ gibt die Häufigkeit wieder, mit der Pixel existieren, die in $I_{PQ}(x_k, y_k)$ den Wert I_1 und in $t(p, q)$ den Wert I_2 haben
 - Der Betrag der Kovarianz ist hoch, wenn sich alle Einträge mit von Null verschiedenen Werten auf einer Geraden befinden
 - Der Betrag der Kovarianz ist niedrig, wenn die Häufigkeit beliebig in $H(I_{PQ}, t)$ streut



Abstandsmaße (8)

- Zum Korrelationskoeffizient (Forts. 4):
 - Ein perfektes Ähnlichkeitsmaß würde für alle Bildpaare, deren gemeinsame Häufigkeitsverteilung im 2D-Histogramm $H(I_{PQ}, t)$ auf einer Gerade liegen, denselben Wert ergeben
 - Das ist bei der Kovarianz nicht der Fall, da deren Wert noch vom Winkel der Geraden, d.h. vom Verhältnis der Varianzen beider Bildfunktionen abhängt. Für eine **Normalisierung** ist die Kovarianz daher **mit den Varianzen beider Bildfunktionen** (bzw. den Wurzeln aus diesen) zu normieren:

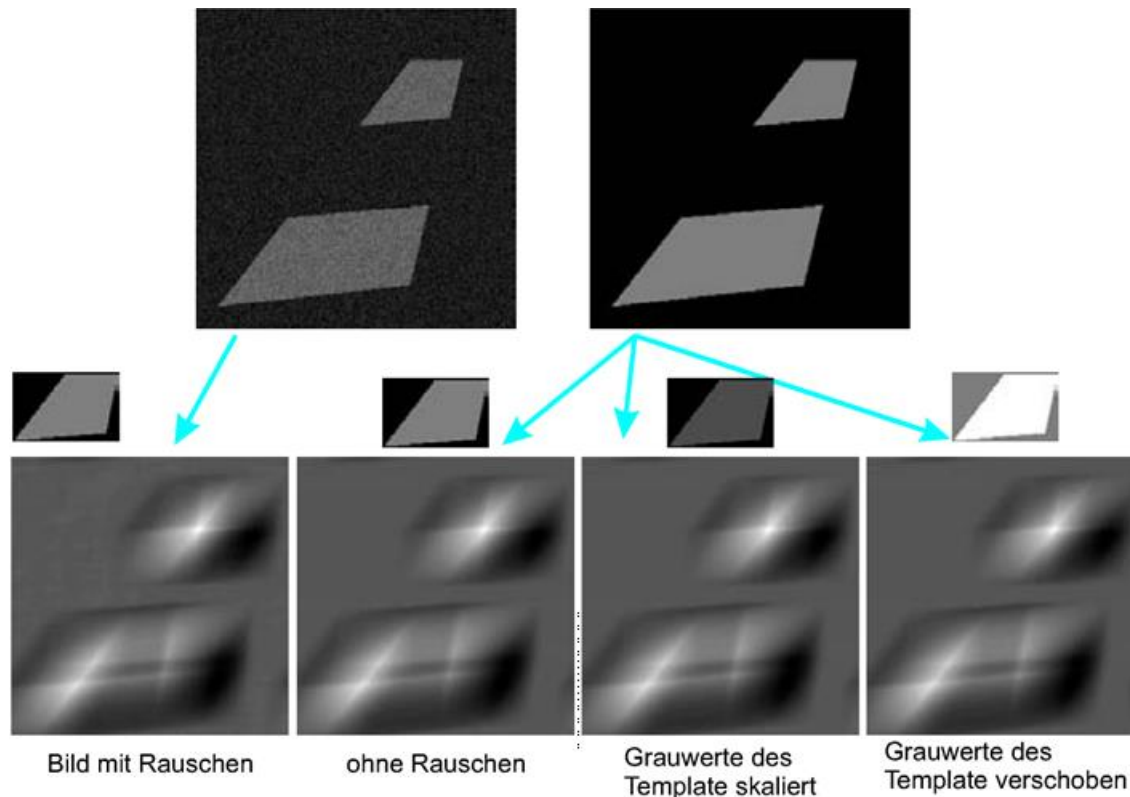
$$cc_{I,t}(x_k, y_k) = \frac{\sigma_{I,t}(x_k, y_k)}{\sqrt{\sigma_I^2(x_k, y_k) \cdot \sigma_t^2(p, q)}}$$
$$= \frac{\sum_{p=0}^{P-1} \sum_{q=0}^{Q-1} \left((I(x_k + p, y_k + q) - \overline{I(x_k + p, y_k + q)}) (t(p, q) - \overline{t(p, q)}) \right)}{\sqrt{\sum_{p=0}^{P-1} \sum_{q=0}^{Q-1} \left((I(x_k + p, y_k + q) - \overline{I(x_k + p, y_k + q)}) \right)^2 \cdot \sum_{p=0}^{P-1} \sum_{q=0}^{Q-1} \left((t(p, q) - \overline{t(p, q)}) \right)^2}}$$

Abstandsmaße (9)

- Zum Korrelationskoeffizient (Forts. 5):
 - $cc_{l,t}(x_k, y_k) = 1$, wenn es Skalare $s > 0$ und d gibt mit $l_{PQ}(x_k, y_k) = s \cdot t(p, q) + d$.
 - $cc_{l,t}(x_k, y_k) = -1$, wenn $s < 0$.
 - $cc_{l,t}(x_k, y_k) = 0$, wenn keine lin. Abhängigkeit besteht zw. $l_{PQ}(x_k, y_k)$ und $t(p, q)$.

Abstandsmaße (10)

- Zum Korrelationskoeffizient (Forts. 6): der Einsatz des Korrelationskoeffizienten macht das Template Matching **unempfindlich gegenüber Skalierung oder Verschiebung der Intensitätswerte zwischen Muster und Bild**. Anwendungsabhängig ist zu entscheiden, ob die negative Korrelation (Bild ist dort hell, wo Muster dunkel ist) akzeptabel ist. Wenn ja, wird der Betrag des Korrelationskoeffizienten zur Findung der Übereinstimmung zw. Muster und Bild genutzt.



Bildquelle: Klaus Tönnies: Grundlagen der Bildverarbeitung, Pearson Studium, 2005.

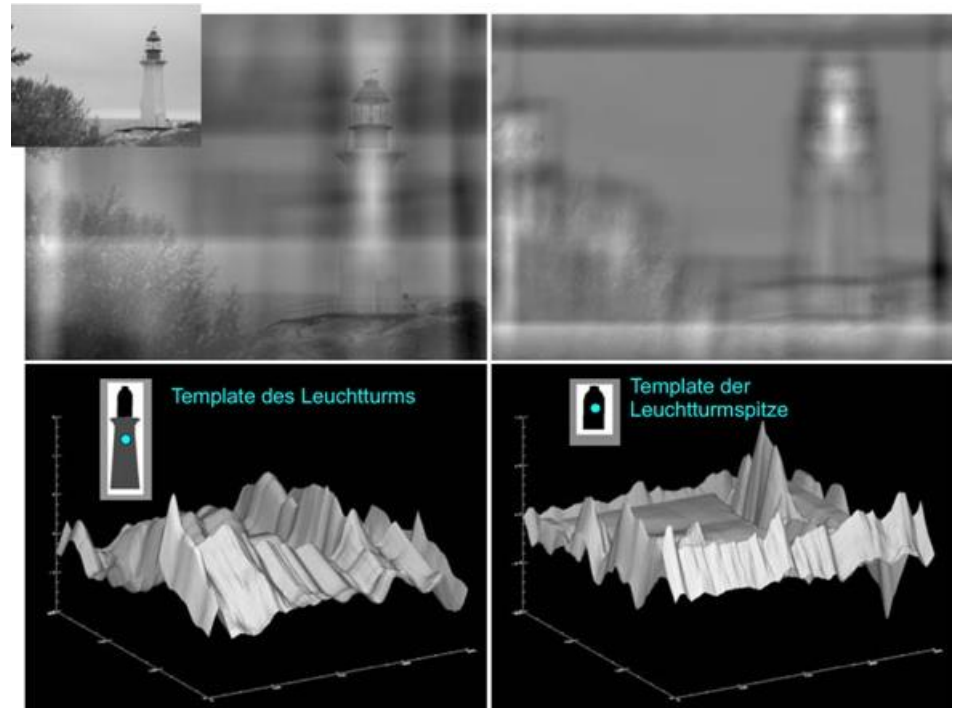
Relation zwischen Intensitätswerten

- **Beispiel:** die Überlagerung eines Template Matching mit dem Originalbild (oben). Das Muster links (der gesamte Leuchtturm) ist detaillierter als das Muster rechts (nur die Spitze des Leuchtturms).

Die Matching-Funktionen unten zeigen rechts ein besseres Resultat.

Der Grund: die Korrelation ist zwar unempfindlich gegenüber Skalierungen der Intensitätswerte.

Die Relation zwischen den Intensitätswerten der Komponenten (hier der Turm, die Spitze und der Hintergrund) muss aber korrekt sein. Das ist für das Muster links nicht der Fall.



Bildquelle: Klaus Tönnies: Grundlagen der Bildverarbeitung, Pearson Studium, 2005.

Größe des Templates

Sowohl bei den Abstandsmaßen msd und mad als auch beim Korrelationskoeffizienten cc erfolgt der Vergleich nur zwischen Bildausschnitten $I_{PQ}(x_k, y_k)$ und dem Template $t(p, q)$.

Die Größe des Musters $t(p, q)$ und der Bildausschnitte $I_{PQ}(x_k, y_k)$ sollten so groß gewählt werden, dass neben dem eigentlichen **Zielobjekt** auch **Hintergrund** erfasst wird, damit die **Form des eigentlichen Zielobjektes** in die Suche einfließen kann.

Wie im Bspl. des Leuchtturms auf der vorherigen Folie, die drei Komponenten zeigt: den Turm und die Spitze des Zielobjektes *und* den benachbarten Hintergrund.



Template-Varianten

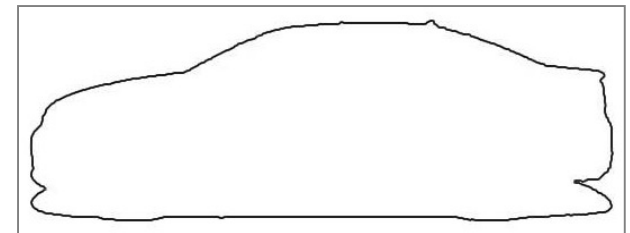
- Templates für Muster mit unbekannten Intensitätsvariationen:

Nicht immer ist die Intensitätsverteilung im Muster festlegbar!

- Bspl: Seitenansichten von Autos mit Fließheck können verschiedenste dunkle und helle Farben und Farbkombinationen zeigen. Ein Muster mit einer Ausprägung kann diese Variationsbreite nicht abdecken
- Allgemein kann dann das Template Matching auch auf **Gradientenbeträgen** ausgeführt werden. Das Muster entspricht dann der **Objekt-silhouette**.



Bildquelle: Toyota Motor Corp.



Die Silhouette könnte durch Glättung noch „generalisiert“ werden, um verschiedene Fließheckautotypen abdecken zu können. Ferner: das Miterfassen des Bodenschattens ist in der bildbasierten Objekterkennung durchaus üblich, da er auch in der Realität auftritt.

- Die erhöhte Flexibilität kostet aber Stabilität: da die Form nur noch über den Objektrand definiert ist, führen Abweichungen leichter zu größeren Variationen im Korrelationskoeffizienten.

Robustheit von Template Matching (1)

- **Formtoleranz:** auch kleine Formabweichungen der Objektinstanz im Bild vom Muster werden toleriert, da jedes Pixel des Musters mit demselben Gewicht in die Berechnung des Qualitätsmaßes einfließt.

Solange die große Mehrheit der Bildpixel dem Muster entspricht, wird die entspr. Position zu einem lokalem Maximum der Ergebnisfunktion führen.

- **Skalierungs- und Rotationsabhängigkeit:** Template Matching ist empfindlich gegenüber Größen- und Rotationsabweichungen zw. Muster und Bild.

Dem kann begegnet werden, indem der Parameterraum um zwei Parameter s und α der Skalierung bzw. der Rotation erweitert wird.

Die Suche wird aufwändiger, da die Bewertung dann in einem vier-dimensionalen Raum in den Parametern x_k , y_k , s und α zu bestimmen ist.

Robustheit von Template Matching (2)

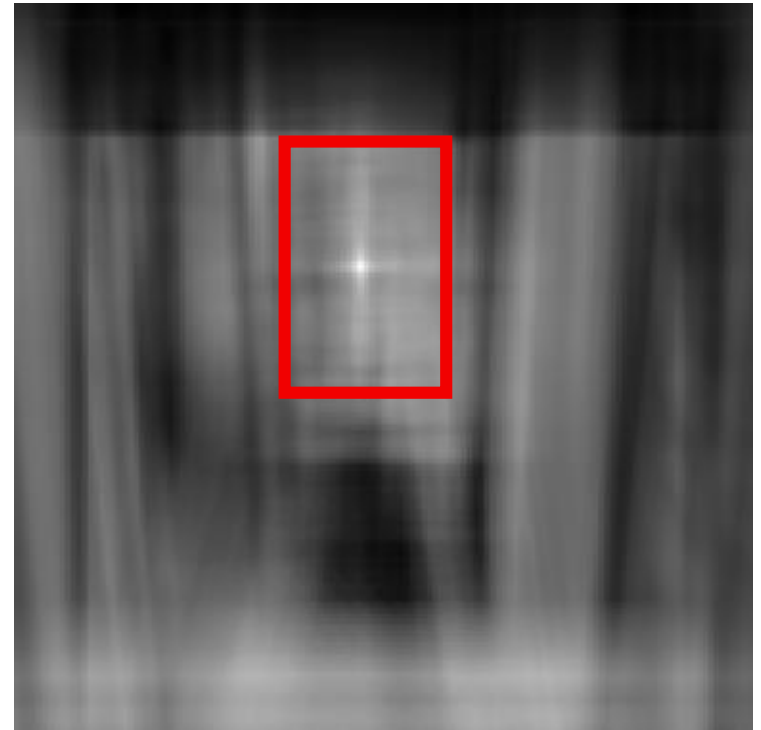


This is a chair

Find the chair in this image



Output of normalized correlation

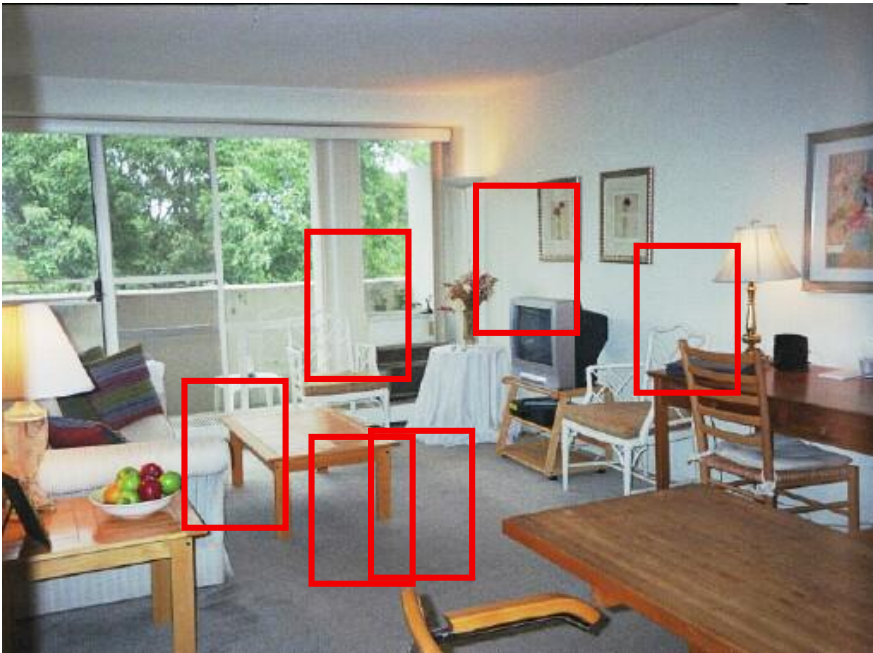


Robustheit von Template Matching (3)

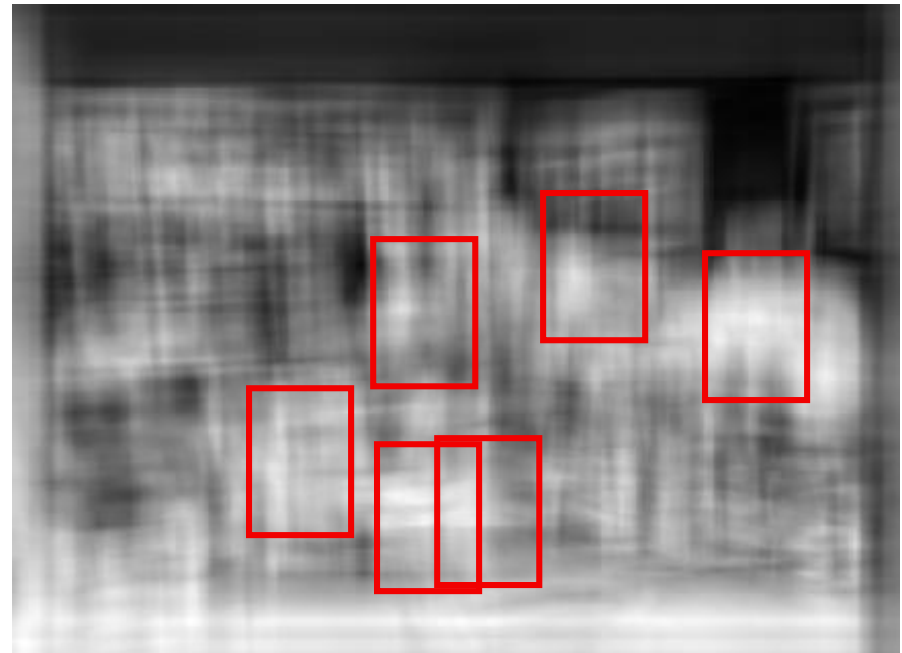


This is a chair

Find the chair in this image



Pretty much garbage
Simple template matching
is not going to make it

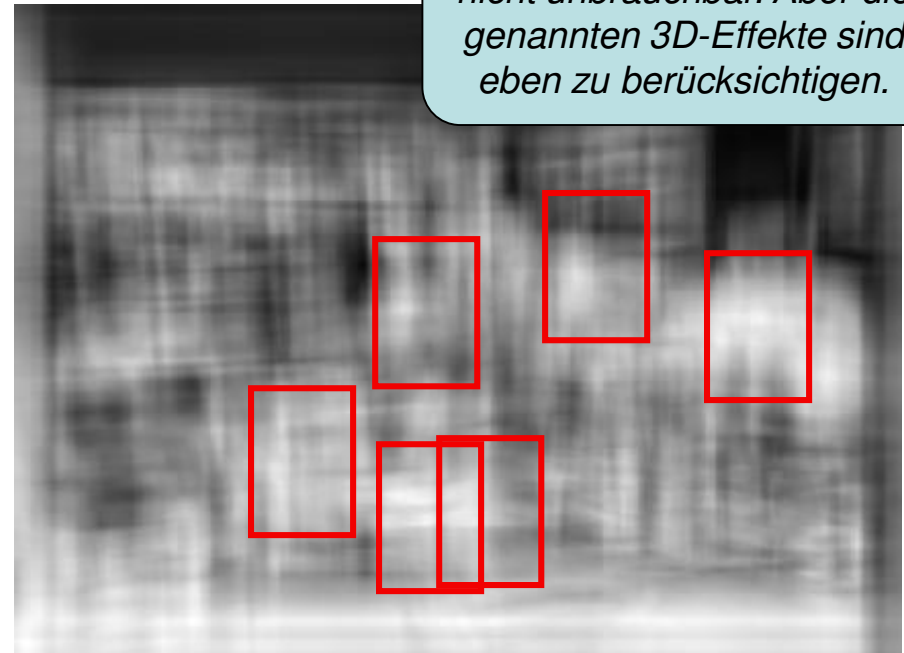
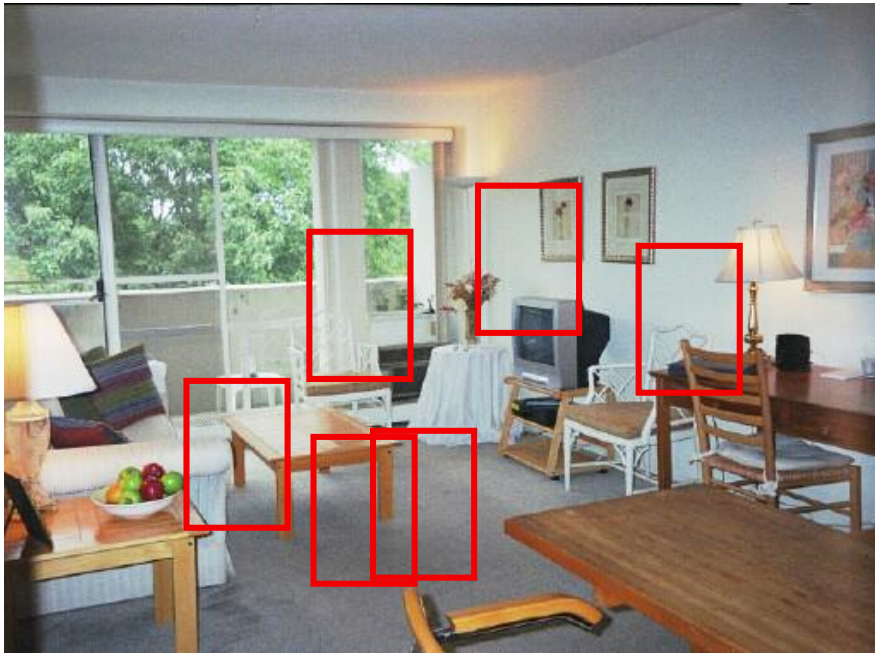


Robustheit von Template Matching (4)



A “popular method is that of template matching, by point to point correlation of a model pattern with the image pattern. These techniques are inadequate for *three-dimensional scene analysis* for many reasons, such as *occlusion*, *changes in viewing angle*, and *articulation of parts*.” Nevatia & Binford, 1977.

Template-Matching ist damit nicht unbrauchbar. Aber die genannten 3D-Effekte sind eben zu berücksichtigen.



Hough-Transformation (1)

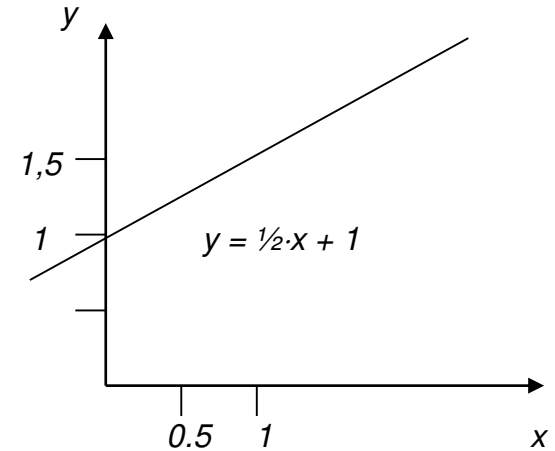
- Vergleich der Hough-Transformation (HT) zu Template Matching:
 - Die HT ist dem Template Matching insofern ähnlich, indem die HT ebenfalls nach vorgegebenen Mustern sucht.
 - Die Muster sind jetzt aber nicht durch Pixelfelder definiert, sondern durch Kantenzüge, die durch eine Parameterbeschreibung definierbar sind.

Hough-Transformation (2)

Bspl. 1: Geraden sind im zweidim. Bildraum durch zwei Parameter beschreibbar. Z.B. durch die Steigung m und den y-Achsenabschnitt b nach

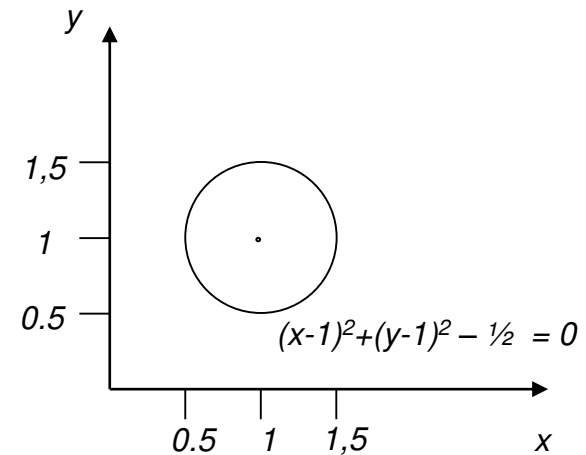
$$y = m \cdot x + b:$$

Nur Punkte (x,y) , die diese Gleichung erfüllen, liegen auf dieser Geraden.



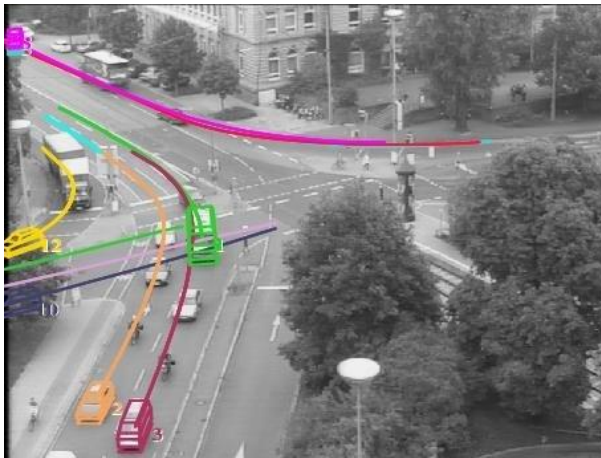
Bspl. 2: Kreise sind im zweidim. Bildraum durch drei Parameter beschrieben: zwei Positionsparameter für den Mittelpunkt (x_0, y_0) und den Radius r . Alle Punkte x, y des Kreises müssen dessen Kreisgleichung erfüllen:

$$(x-x_0)^2 + (y-y_0)^2 - r^2 = 0.$$



Hough-Transformation für Geraden (1)

- Die **Hough-Transformation für Geraden** im Bildraum
 - Zahlreiche Anwendungsszenarien enthalten viele von Menschen geschaffene Objekte und Strukturen (engl. *man-made objects*): Städtebau, Architektur, Straßen und Automobile, Innenraummobiliar, etc.
 - Viele dieser *man-made objects* sind geometrisch durch Polyederformen (ggf. approximativ) beschreibbar. Die Bildprojektionen von *man-made objects* sind entsprechend durch **geradlinige Konturen** begrenzt.
 - Aus diesem Grunde ist die *Extraktion von geraden Konturkanten* relevant.



Bildquelle: Univ. Karlsruhe, Inst. für Algorithmen und Kognitive Systeme, FG Prof. Dr. H.-H. Nagel



Bildquelle: J. Behley, V. Steinhage: Generation of 3D City Models Using Domain-Specific Information Fusion. Proc. 7th Intern. Conf. on Computer Vision Systems, ICVS 2009, 164-173, 2009.

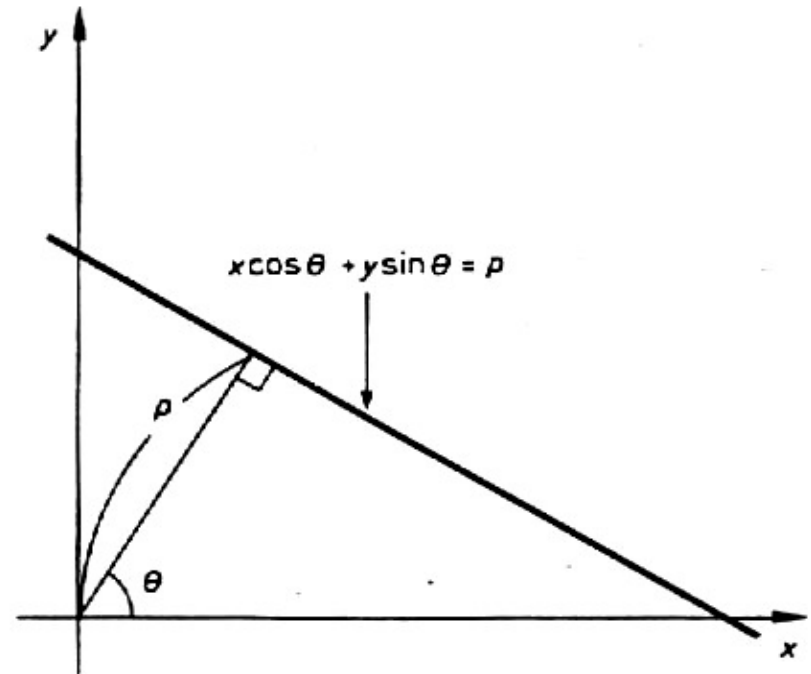
Hough-Transformation für Geraden (2)

Die Hessesche Normalform

- Die HT nutzt statt der Gleichung $y = mx + b$ die sog. Hessesche Normalform zur Beschreibung von Geraden in den beiden Parametern θ und ρ :

$$x \cdot \cos(\theta) + y \cdot \sin(\theta) = \rho.$$

- θ bezeichnet den Winkel zw. der Normalen der Geraden und der positiven x-Achse
- ρ bezeichnet den Abstand der Geraden vom Ursprung.



Hough-Transformation für Geraden (3)

Aufgabe und Quelle der Hough-Transformation für Geraden

- Die Hough-Transformation zur Erkennung von Geraden in Bildern hat genauer die Aufgabe der Erkennung von **kollinearen Punkteanordnungen** in Bildern.
- Wird die HT für Geraden auf den **Gradientenbeträgen** angewendet, so werden geradlinige Konturkanten extrahiert.
- Die Hough-Transformation basiert in der Grundversion i. W. auf einem Patent von P. Hough (1962) und Vorschlägen von R. O. Duda und P. E. Hart (1972).

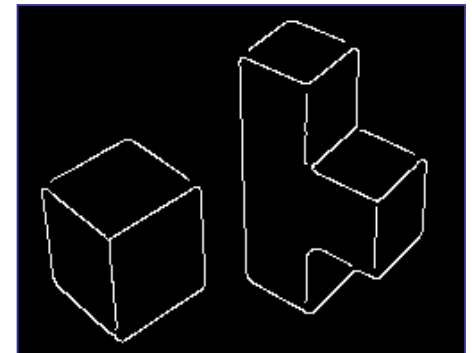
⁽¹⁾ P. V. C. Hough: *Method and Means for Recognizing Complex Patterns*. U.S. Patent 3069654, 1962.

⁽²⁾ R. O. Duda, P. E. Hart: *Use of the Hough Transformation to Detect Lines and Curves in Pictures*. Communications of the ACM, Vol. 15, 11-15, 1972.

Hough-Transformation für Geraden (4)

Ansatz der Hough-Transformation für Geraden

- Die Hough-Transformation arbeitet generell nach einem **Abstimmungsprinzip** (engl. *voting system*):
 - Bei der HT für Geraden votiert jedes Kantenpixel (hoher Gradientenbetrag) für *alle* Geraden, die dieses Pixel enthalten können
 - Das Abstimmen erfolgt über die Parameter θ und ρ der entspr. Geraden-gleichungen in Hess. NF
 - Die Stimmen werden im zweidim. **Parameterraum** (θ, ρ) gesammelt, dem sog. **Hough-Raum**
 - Alle Kantenpixel, die zu einem geradlinigen Kantenzug gehören, erzeugen ein **lokales Stimmenmaximum** in Parameterraum
 - Die lokalen Maxima im Hough-Raum bilden die Hypothesen für die Geraden im Bildraum



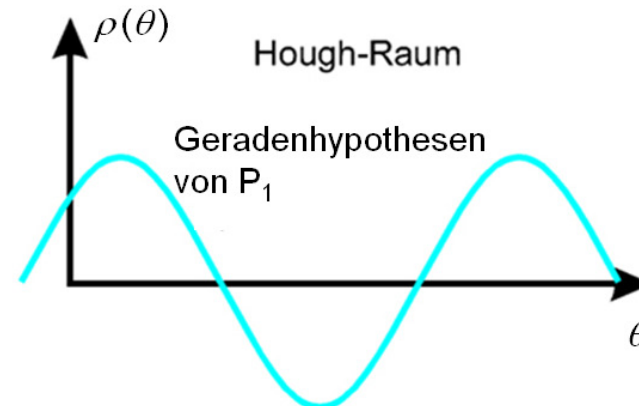
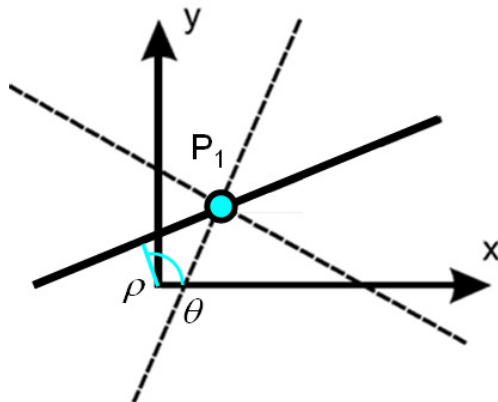
Hough-Transformation für Geraden (5)

Das Abstimmungsprinzip der Hough-Transformation (1)

- Für die **Abstimmung** wird die Hess. NF umgestellt: der Abstand ρ wird als Funktion des Winkels θ geschrieben:

$$\rho(\theta) = x \cos(\theta) + y \sin(\theta).$$

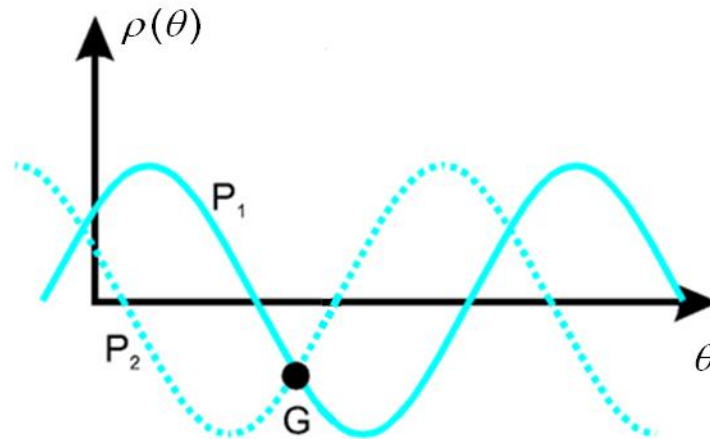
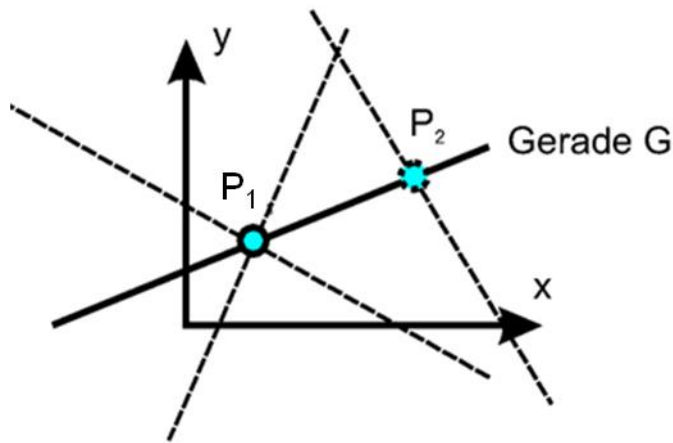
- Ein Kantenpunkt $P_1 = (x_1, y_1)$ votiert für alle Geraden mit Parameterpaaren (ρ, θ) , die sich nach obigem Funktional für x_1, y_1 mit $0^\circ \leq \theta < 180^\circ$ ergeben.
- Wegen der trigonometr. Funktion bilden die Parameterpaare (ρ, θ) aller Geradenhypothesen zu $P = (x_1, y_1)$ eine sinusförmige Kurve im Hough-Raum:



Hough-Transformation für Geraden (6)

Das Abstimmungsprinzip der Hough-Transformation (2)

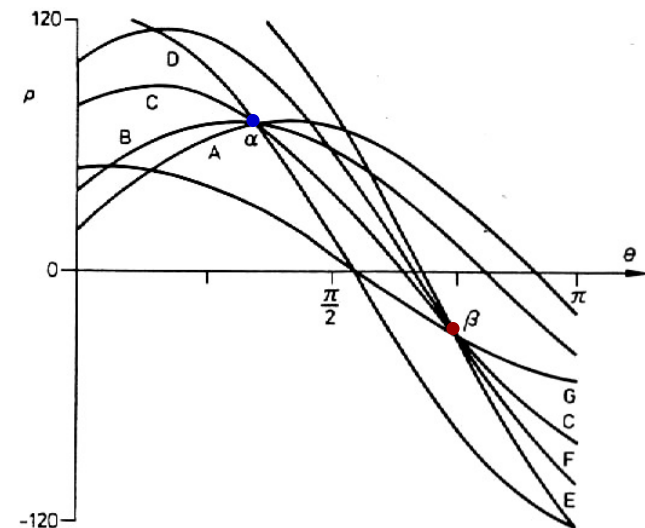
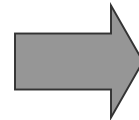
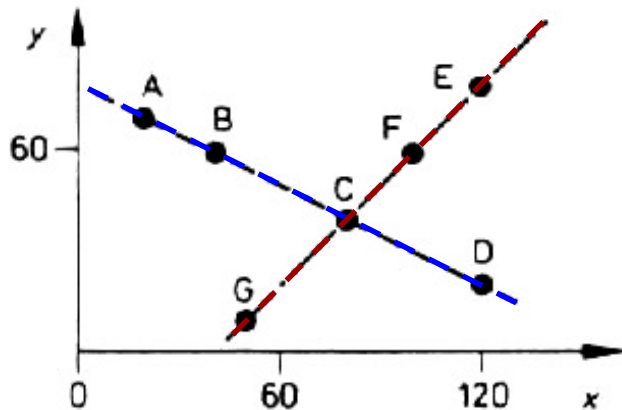
- Ein zweiter Kantenpunkt $P_2 = (x_2, y_2)$ votiert wieder für alle Geraden mit Parameterpaaren (ρ, θ) nach $\rho(\theta) = x_2 \cos(\theta) + y_2 \sin(\theta)$ für $0^\circ \leq \theta < 180^\circ$ und erzeugt eine weitere sinusförmige Kurve im Hough-Raum.
- Dort wo sich beide Kurven schneiden, finden sich die Parameterpaare (ρ, θ) der Geraden, auf der P_1 und P_2 liegen:



Hough-Transformation für Geraden (7)

Das Abstimmungsprinzip der Hough-Transformation (3)

- Zwei Kantenpunkte sind natürlich kein Anzeichen für das Vorliegen eines geradlinigen Kantenzuges. Mit zunehmender Zahl von kollinear angeordneten Kantenpunkten müssen sich jedoch auch mehr Kurven im entspr. Parameterpaar (ρ, θ) schneiden.
- Nach Transformation aller Kantenpunkte in den Hough-Raum sind diese Schnittpunkte als Parameterpaare (ρ, θ) von Geradenhypothesen zu suchen:



Diskrete Hough-Transformation für Geraden (1)

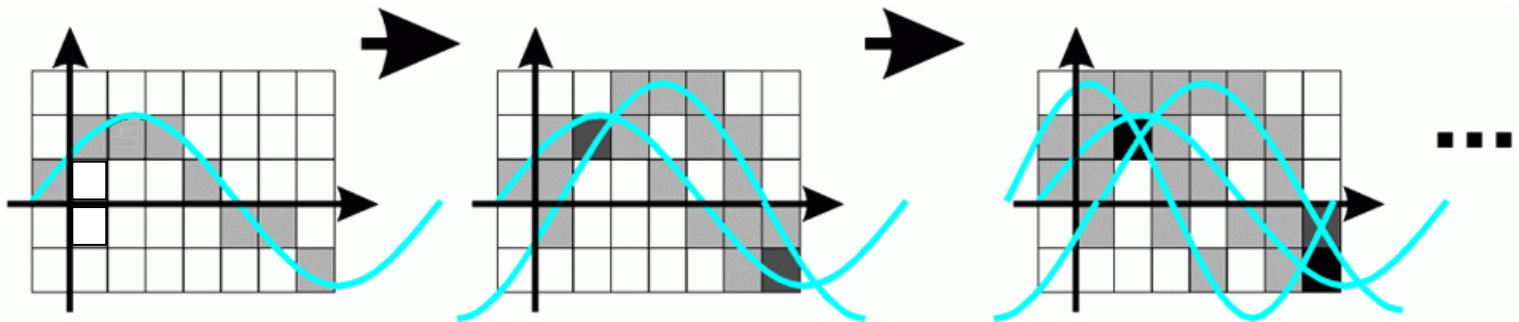
Die diskrete Hough-Transformation (1)

- I.A. liegen die Kantenpunkte in digitalen Bildern als **Kantenpixel** mit diskreten Koordinaten (x,y) vor.
- Zudem kann wegen der unvermeidlichen **Rechen- und Diskretisierungsungenauigkeiten** nicht erwartet werden, dass sich die Kurven in genau einem Punkt schneiden.
- Daher diskretisiert die **diskrete Form der HT** den Hough-Raum in sog. **Zellen**, die diskreten Parameterpaaren (ρ, θ) entsprechen.

Diskrete Hough-Transformation für Geraden (2)

Die diskrete Hough-Transformation (2)

- Diese Zellen des Hough-Raums werden auch **Akkumulatorzellen** genannt (weshalb der Hough-Raum auch als **Akkumulatorfeld** bezeichnet wird):
 - Jede sinusförmige Kurve, die sich aus der HT ergibt, wird mit der Auflösung des Hough-Raums abgetastet;
 - überdeckt eine Kurve eine Zelle, so erhält die Zelle eine Stimme;
 - nach Abschluss der HT werden die akkumulierten Stimmen ausgezählt: lokale Maxima entsprechen den Parametern der gesuchten Geraden.



Diskrete Hough-Transformation für Geraden (3)

Wertebereich und Diskretisierung der Parameter der diskreten HT:

- der Winkelparameter θ :

$$0 \leq \theta < \pi$$

- der Abstandsparameter ρ kann durch die Bilddiagonale beschränkt werden, also die maximalen x- und y-Werte:

$$-\sqrt{x_{\max}^2 + y_{\max}^2} \leq \rho \leq +\sqrt{x_{\max}^2 + y_{\max}^2}$$

- Die Diskretisierung des Winkelparameters θ kann z.B. in Schritten von 1° erfolgen
- Die Diskretisierung des Abstandsparameter ρ kann z.B. durch die Pixelgröße erfolgen

Diskrete Hough-Transformation für Geraden (4)

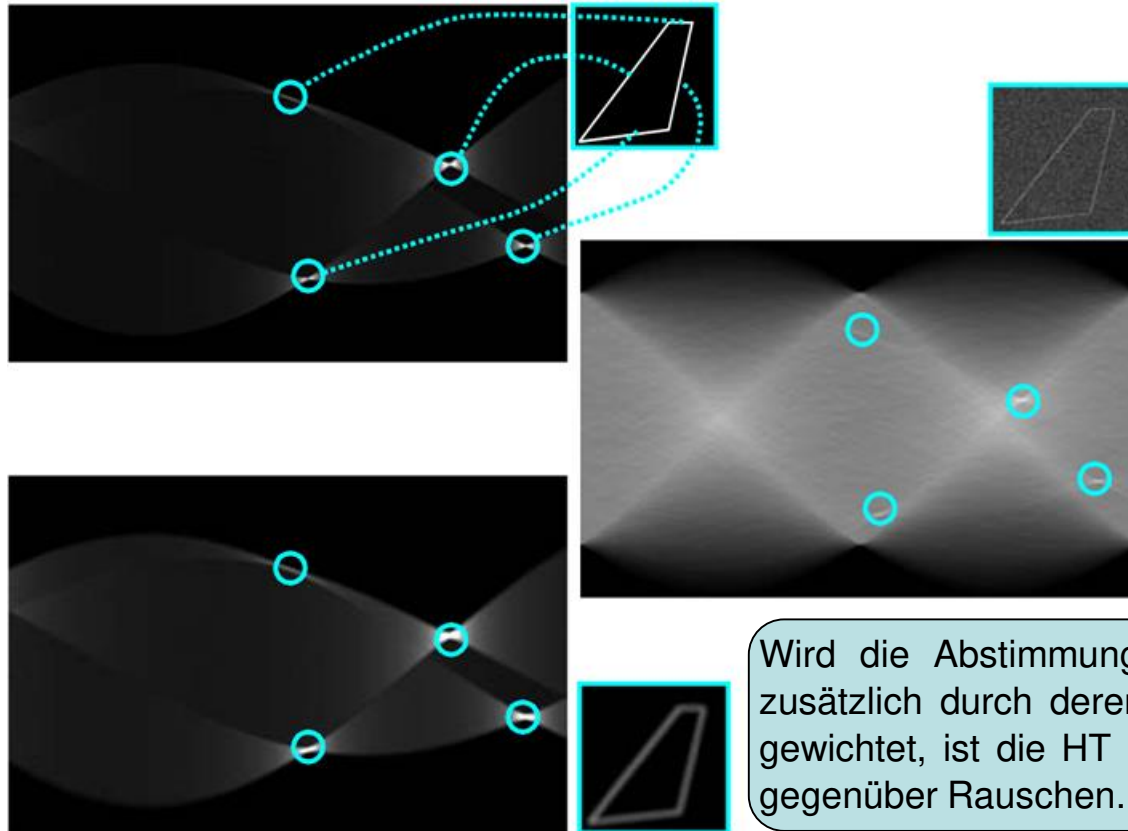
Pseudocode der diskreten Hough-Transformation für Geraden:

```
function discrete_HT (i:image);  
     $\rho\_max = ( (image\_width)^2 + (image\_height)^2 )^{1/2}$ ;  
     $\rho\_min = -1 * \rho\_max$ ;  
    HoughSpace [0... $\pi$ ][ $\rho\_min \dots \rho\_max$  ] = 0; // all cells set to 0  
    for all contour pixels p do  
        for ( $\theta = 0$  ;  $\theta < \pi$  ,  $\theta ++$ ) do // incrementing by 1°  
             $\rho = p.x \cdot \cos(\theta) + p.y \cdot \sin(\theta)$ ;  
            if ( $\rho\_min \leq \rho \leq \rho\_max$ ) then HoughSpace[ $\theta, \rho$ ] $++$  ;  
        endfor;  
    endfor;  
    return;  
end;
```

Diskrete Hough-Transformation für Geraden (5)

Beispiel für diskrete Hough-Transformation:

- Die HT für ein einfaches, von vier Linien begrenztes Objekt erzeugt vier klar erkennbare lokale Maxima im Hough-Raum.



Unscharfe Kanten führen zu einer Verbreiterung der lokalen Maxima.

Wird die Abstimmung der Kantenpunkte zusätzlich durch deren Gradientenbeträge gewichtet, ist die HT relativ unempfindlich gegenüber Rauschen.

Diskrete Hough-Transformation für Geraden (6)

Zur Extraktion der Maxima bei der diskreten HT:

- Jeder Kantenpixel kann bei Diskretisierung in 1° -Schritten 180 verschiedene Stimmen für Geradenhypothesen abgeben:
 - nur eine dieser Stimmen ist korrekt!
 - 179 Stimmen sind Fehlstimmen!
- Damit wird im Hough-Raum quasi ein „Meer von Fehlstimmen“ erzeugt, das man als „Abstimmungsrauschen“ bezeichnen könnte.
- Die lokalen Maxima der korrekten Geradenhypothesen müssen stark genug sein, um sich gegenüber diesem Abstimmungsrauschen durchzusetzen.
- Ein Anliegen beim Entwurf der HT kann also sein, die Zahl der Fehlstimmen sinnvoll zu minimieren!

Diskrete Hough-Transformation für Geraden (7)

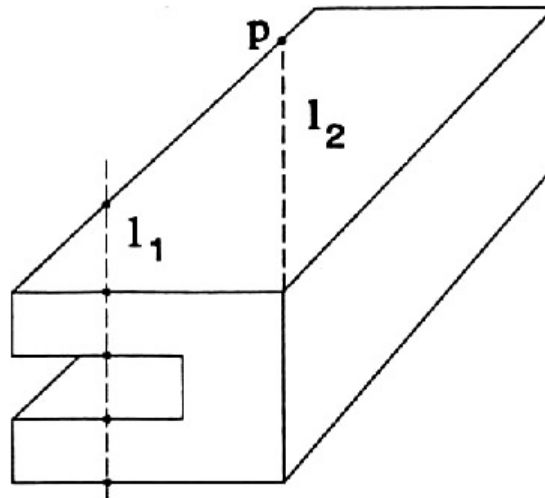
Vermeidung von Fehlstimmen bei der diskreten HT (1):

- Ein erster Ansatz dazu wurde beiläufig im letzten Beispiel erwähnt:
 - Die Stimmen der Kantenpixel werden mit den Gradientenbeträgen derselben gewichtet
 - Unter der Annahme eines günstigen Signal-Rausch-Verhältnisses sollte das Rauschen Gradienten mit geringeren Beträgen erzeugen als bei Kantenpixeln
 - Damit sollten also nur noch Stimmen von Kantenpixeln und nicht mehr durch Bildrauschen berücksichtigt werden
 - Zusätzlich werden die Gradientenbeträge als ein Maß der „Verlässlichkeit“ der Stimmen gewertet

Diskrete Hough-Transformation für Geraden (8)

Vermeidung von Fehlstimmen bei der diskreten HT (2):

- Trotz der Berücksichtigung der Gradientenbeträgen wird jedes Kantenpixel immer 179 Fehlstimmen abgeben.
- Diese erzeugen nicht nur ein Abstimmungsrauschen, sondern können sogar zu Fehlhypothesen führen:

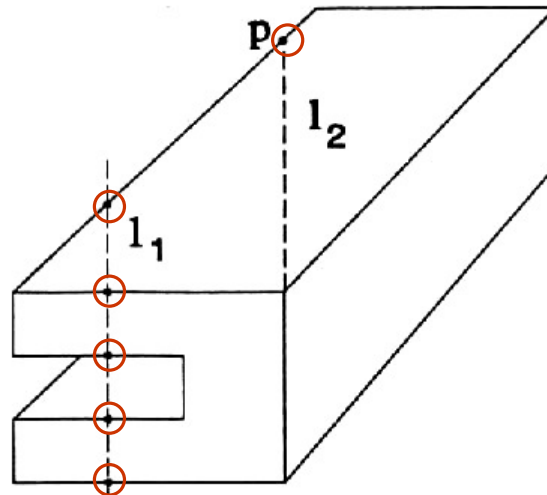


Nach W.-N. Lie, Y-C. Chen: *Line Drawing Extraction Of Polyhedrons Using Polarized Hough Transform*. In Proc. 14th Annual Conf. of Industrial Electronics Society 1988 (IECON '88), 1988.

Diskrete Hough-Transformation für Geraden (9)

Vermeidung von Fehlstimmen bei der diskreten HT (3):

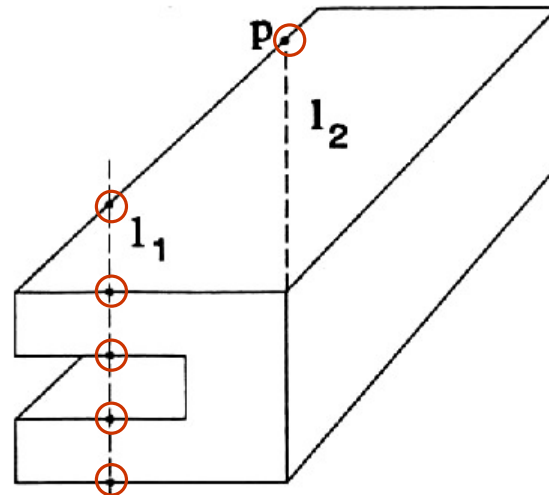
- Die Fehlstimmen können durch lokale Inspektion vermieden werden.
- Am einfachsten durch Bezug auf die Gradientenorientierungen der votierenden Kantenpixel.
- Deren Gradientenorientierungen sollten orthogonal zur Kantenrichtung sein. Dies gilt offenbar nicht für die rot markierten Kantenpunkte im Bspl.



Diskrete Hough-Transformation für Geraden (10)

Vermeidung von Fehlstimmen bei der diskreten HT (4):

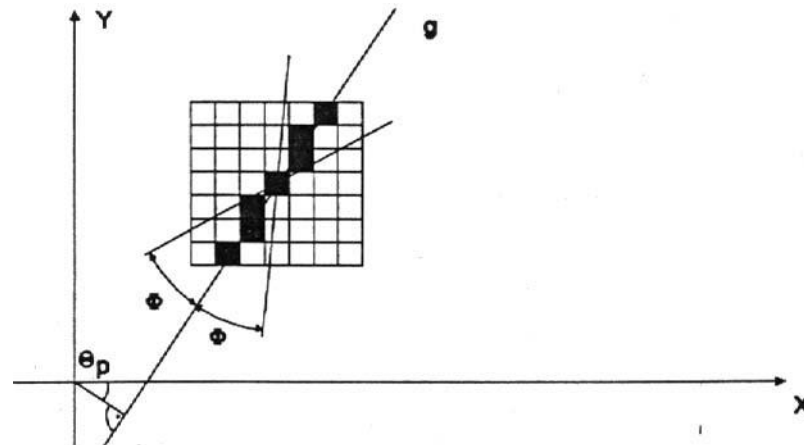
- Implementiert wird also eine Einschränkung der Abstimmung eines Kantenpixels auf einen **Polarisationswinkel** θ_p und einem **Toleranzwinkel** ϕ , also auf den Bereich $\theta_p \pm \phi$,
 - wobei θ_p in Gradientenrichtung des Kantenpixels gewählt wird
 - und ϕ heuristisch durch Abschätzung der Orientierungsunschärfe zu wählen ist (z.B. 5° oder 10°).



Diskrete Hough-Transformation für Geraden (11)

Vermeidung von Fehlstimmen bei der diskreten HT (5):

- Der **Polarisationswinkel** θ_p und der **Toleranzwinkel** ϕ können auch genauer durch eine etwas aufwändiger lokale Inspektion einer Nachbarumgebung bestimmt werden.
- Dazu wird eine quadratische Nachbarschaftsumgebung von z.B. 5×5 oder 7×7 Pixeln definiert, die im votierenden Kantenpixel zentriert ist. In dieser Umgebung wird θ_p durch lokale Geradenorientierung interpoliert und ϕ als maximale Abweichung von dieser geschätzt:



Diskrete Hough-Transformation für Geraden (12)

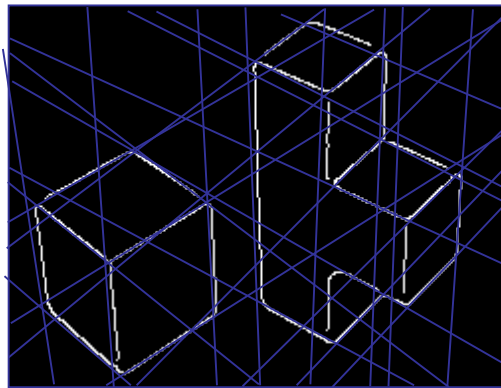
Lücken in Kantenzügen und die HT:

- Im Ggs. zu Edge Linking und der interaktiven Kantenverfolgung ist die HT robuster gegenüber Lücken in Kantenzügen:
 - solange genug Kantenpixel für eine Gerade votieren, wird diese trotz Lücken erkannt.
- Die gilt übrigens analog für Template Matching: solange genug Bildpixel mit dem Muster übereinstimmen, wird dieses trotz partiellen Signalausfall oder Rauschen erkannt.
- Allgemeiner: Edge Linking und die interaktive Kantenverfolgung sind eher lokale Verfahren, die nur lokal begrenzte Information zum Ausgleich bzw. zur Überbrückung einsetzen können.
- HT und Template Matching sind eher globale Verfahren, die das gesamte Bild bzw. einen größeren Bildausschnitt zusammenhängend nach einem Zielkriterium auswerten.

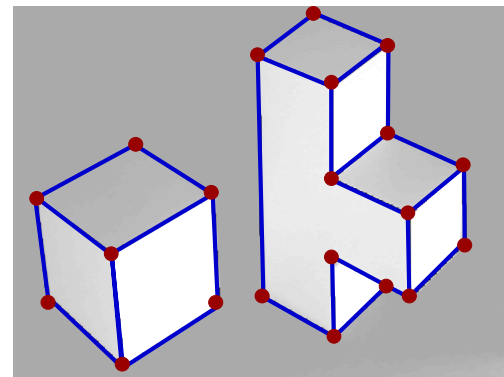
Diskrete Hough-Transformation für Geraden (13)

Begrenzte Kantenzüge und die HT (1):

- Die HT für Geraden erkennt die Gleichungen von Geraden *unendlicher* Länge
- Die gesuchten Kanten in einem Bild haben aber *endliche* Längen
- Dazu müssen die lokalen Maxima aus dem Hough-Raum in das originale Bild zurückprojiziert werden
- Eine endliche Kante im Bild ergibt sich aus Kantenpixeln (mit hohen Gradientenbeträgen und orthogonalen Gradientenrichtungen), die eine zurückprojizierte Gerade „unterstützen“



Geradenhypothesen

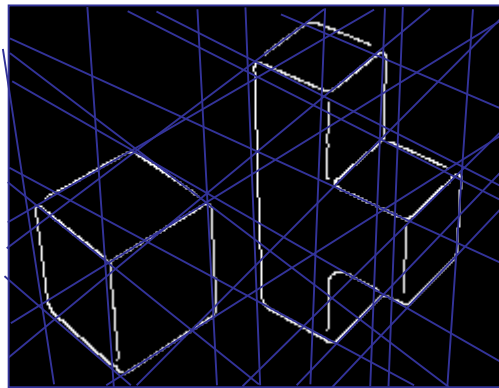


Validierte Linienabschnitte mit
End- bzw. Verbindungspunkten

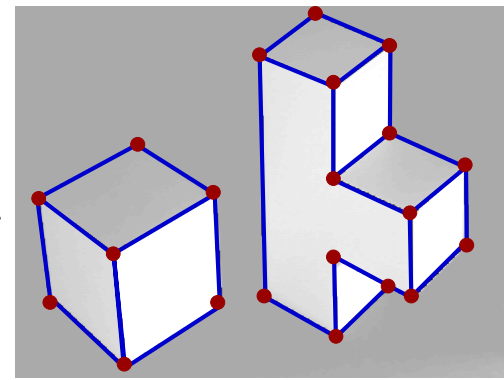
Diskrete Hough-Transformation für Geraden (14)

Begrenzte Kantenzüge und die HT (2):

- Bei der Ermittlung der endlichen Kante im Bild aus den Kantenpixel und den zurückprojizierten Gerade ist i.A. wieder eine Lückentoleranz einzubauen.
- Die Ermittlung der End- und Verbindungspunkte von endlichen Geraden-segmenten kann durch Betrachtung der Gradientenorientierungen bzw. durch Schnittbildung von Kantenhypothesen ermittelt werden.



Geradenhypothesen



Validierte Linienabschnitte mit
End- bzw. Verbindungspunkten

Hough-Transformation für Kreise (1)

Die HT für Kreise (*Circular Hough Transform* – CHT):

- Die HT ist auch für andere parametrisierbare Formen einsetzbar
- **Kreise** sind im zweidim. Bildraum durch drei Parameter beschrieben:
 - zwei Positionsparameter für den Mittelpunkt (x_0, y_0)
 - der Radius r
- Alle Punkte x, y des Kreises müssen dessen Kreisgleichung erfüllen:

$$(x-x_0)^2 + (y-y_0)^2 - r^2 = 0.$$

Hough-Transformation für Kreise (2)

Die HT für Kreise (*Circular Hough Transform* – CHT):

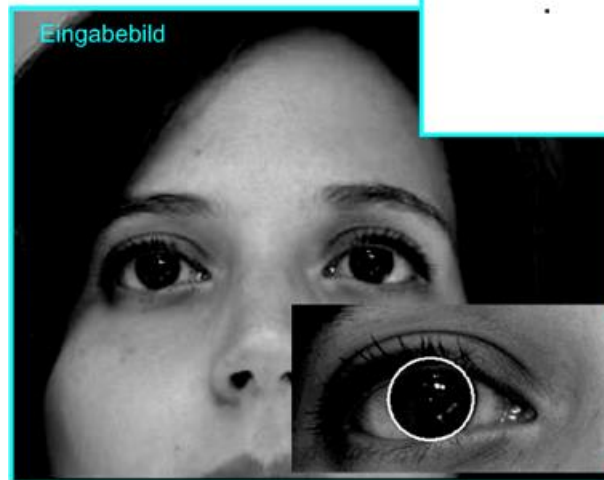
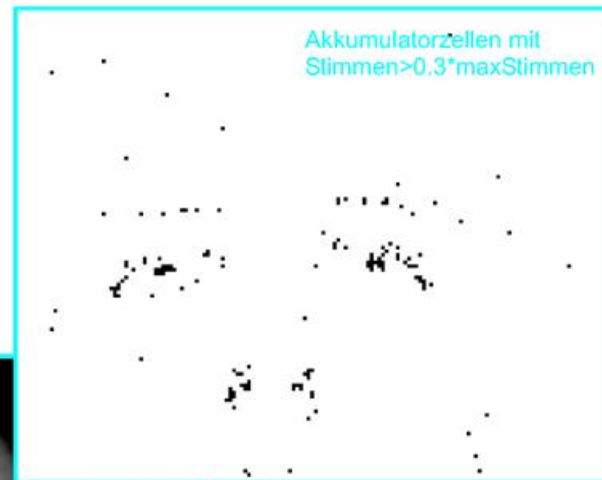
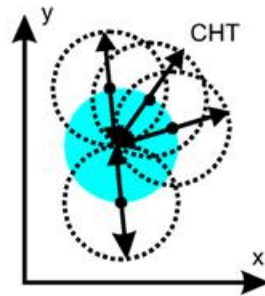
- Die HT für Kreise erfolgt völlig analog zur HT zur Geradendetektion.
- Der Radiusparameter r kann ggf. durch Vorgaben der Anwendung auf ein Intervall $[r_{\min}, r_{\max}]$ beschränkt werden.
- Die CHT kann beschleunigt werden, indem beim Votieren nur Kantenpixel berücksichtigt werden, deren Gradienten zum Kreiszentrum zeigen bzw. weg von diesem. Also werden nur Stimmen in Gradientenrichtung gezählt.

Hough-Transformation für Kreise (3)

Beispiel für die HT für Kreise:

- Die HT für Kreise wird besonders einfach und anschaulich, wenn der Radius r bekannt ist und dann nur noch die Kreiszentren zu suchen sind. Der Hough-Raum entspricht dann dem Bildraum.

Bei vorgegebenen Radius r gibt jeder Kantenpunkt Stimmen für alle Orte im Abstand r ab.



Effizienz der Hough-Transformation

Effizienz der HT:

- Die originale Hough-Transformation ist aufwändig und daher z. B. nicht zur Analyse von Videosequenzen in Echtzeit geeignet.
- Es gibt aber viele Weiterentwicklungen der Hough-Transformation, die häufig als „Fast Hough Transform“ (FHT) benannt werden.
- Viele Ansätze der FHT zerlegen den Hough-Raum hierarchisch (von Ansatz her ähnlich wie Quadrees oder Octrees) in Hyperkuben. Die Bearbeitung auf einem derart hierarchisch untergliederten Hough-Raum kann dann einer Grob-nach-fein-Strategie (engl. *coarse to fine*) folgen.

Zusammenfassung (1)

- Beim **Template Matching** wird das gesuchte Segment durch ein Muster (engl. Template) in seiner Form und Orientierung spezifiziert, nicht aber in seiner Position spezifiziert!
 - Der Einsatz des **Korrelationskoeffizienten** macht das Template Matching unempfindlich gegenüber Skalierung oder Verschiebung der Intensitätswerte zwischen Muster und Bild.
 - Durch **zusätzliche Parameter s und α** der Skalierung bzw. der Rotation ist Template Matching explizit erweiterbar. Die Suche erfolgt dann in einem 4D-Raum in den Parametern x_k , y_k , s und α und wird entsprechend aufwändiger.
 - Um stärkeren Intensitätsvariationen im Muster zu begegnen, kann Template Matching auch auf **Gradientenbeträgen** ausgeführt werden. Das Muster entspricht dann der **Objektsilhouette**.

Zusammenfassung (2)

- Die **Hough-Transformation** geht von fest vorgegebenen und parametrisierbaren Formklassen aus, deren Instanzen im Bild wiederzufinden sind.
 - Vorgegeben ist also die Formklasse. Gesucht sind die Parameter von Position, Größe (bzw. Skalierung) und Orientierung der Instanzen der Formklasse im Bild.
 - Jeder Bildpunkt, der Teil der entspr. Formklasseninstanz sein kann, votiert in einen diskretisierten Parameterraum, dem sog. Hough-Raum, für die Parameter aller möglichen Formklasseninstanzen, zu denen er gehören kann.
 - Die Parameter der im Bild vorzufindenden Klasseninstanzen ergeben sich dann als lokale Maxima im Hough-Raum.
 - Bei der HT für unendliche Geraden sind die endlichen Kanten im Bild über die Rückprojektion ins Bild zu bestimmen.