

Übung Computergrafik 2

Abgabetermin:

Die Lösung zu diesem Übungsblatt ist bis zum **27.06.2012** abzugeben.

Form der Abgabe:

- Die Übungen können in Gruppen von 2–3 Studierenden bearbeitet werden.
- Die Abgaben bestehen aus den Python-Quelltexten samt aller Bilder, die für die Bearbeitung der Übung verwendet wurden. Idealerweise wird das gesamte PyDev-Projektverzeichnis für die jeweilige Übung über Eclipse als .zip Archiv exportiert. Wichtig: Zur Vermeidung von Namenskonflikten in Eclipse, euren Nachnamen bitte als Präfix vor die Namen der PyDev-Projekte setzen, z.B.: „müller-übung-x“.
- Teilaufgaben, deren Python-Dateien wegen Syntaxfehlern nicht ausführbar sind, werden nicht weiter korrigiert.
- Textaufgaben sollten in Form einer .doc, .odt oder (bevorzugt) als PDF-Datei abgegeben werden, und sollten (falls erforderlich) die benötigten Ausgabebilder der Aufgaben enthalten.
- Alle Übungsabgaben erfolgen über UniWorx¹.

Inhalt: In diesem Übungsblatt behandeln wir die Hough-Transformation.

¹<https://uniworx.ifi.lmu.de>

Aufgabe 1 Hough-Transformation (***)

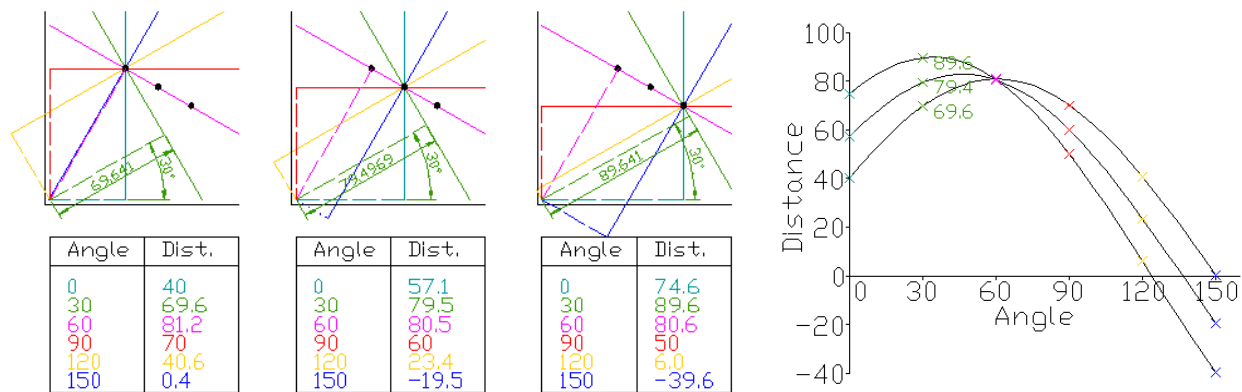


Abbildung 1: Links: (α, d) -Darstellung von Geraden durch einen Punkt. Rechts: Die Akkumulatoren werden im Schnittpunkt der $d(\alpha)$ -Kurven erhöht

In der ihrer Basisvariante ist die Hough-Transformation ein Verfahren zur Geradenfindung in Binärbildern. Dabei betrachtet man für jeden Vordergrundpunkt die möglichen Geraden, die durch diesen Punkt verlaufen, und stellt sie im Hough-Raum (Raum der Parameter (α, d) , Abbildung 1) dar:

$$d(\alpha) = x_n \cos(\alpha) + y_n \sin(\alpha)$$

Die Hough-Transformation eines Kantenbildes beruht nun auf der Diskretisierung des (α, d) -Raums durch Zerlegung in Akkumulatoren. Im einfachsten Fall wird die Gradientenorientierung nicht berücksichtigt. Es wird für jeden Punkt x_n, y_n eine Kurve im (α, d) -Raum diskretisiert. Dabei werden die Akkumulatoren inkrementiert, durch die die Kurve verläuft. Es entsteht ein 2D-Histogramm im Parameterraum. Schließlich sind die Parameter von Linien im Ortsraum durch (α, d) -Kombinationen gegeben, deren Werte nach Ausführung der Transformation am höchsten sind. In anderen Varianten der Hough-Transformation wird die Gradientenorientierung berücksichtigt für jedes Vordergrundpixel nur ein Punkt im (α, d) -Raum inkrementiert. Statt einer einfachen Inkrementierung kann der Akkumulatorwert auch um die Gradientenlänge erhöht werden. Dann ist keine Unterscheidung zwischen Vorder- und Hintergrundpixeln notwendig.

Hough-Transformation für Kreise

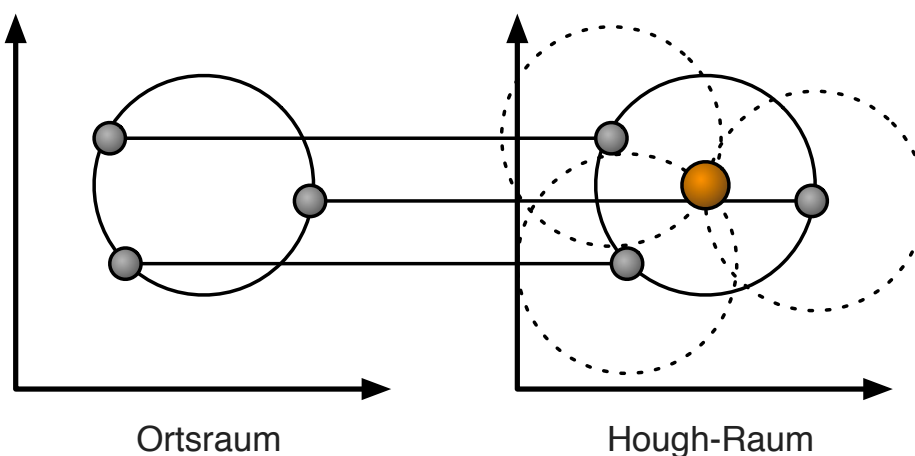


Abbildung 2: Jeder Punkt im Ortsraum (links) generiert einen Kreis im Hough-Raum (rechts), nämlich die Mittelpunkte der Kreise, auf denen der Punkt liegen kann. Die Kreise im Hough-Raum schneiden sich im Mittelpunkt (x_c, y_c) , was auch der Mittelpunkt des Kreises im Ortsraum ist.

Die Hough-Transformation für Kreise betrachtet bei bekanntem Radius eine diskrete Anzahl von Punkten in der Umgebung eines (Mittel-)Punktes (x_c, y_c) . Die Kreisgleichung mit dem Radius r für diesen Punkt lautet dann:

$$(x - x_c)^2 + (y - y_c)^2 - r^2 = 0$$

Für Implementationszwecke ist es sinnvoll, die Kreisgleichung in folgende winkelabhängige Form zu überführen:

$$\begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} x_c + r \cos(\alpha) \\ y_c + r \sin(\alpha) \end{pmatrix}$$

Jeder Punkt einer kreisförmigen Kante mit dem Radius r erzeugt also einen Schnitt mit dem Mittelpunkt (x_c, y_c) , und erhöht somit dessen Akkumulator (Abbildung 2).

- a) Das Berechnen der Rotation von Textzeilen ist z.B. bei der automatischen Texterkennung (OCR) ein wichtiger Vorverarbeitungsschritt. Verwenden Sie die Bilddatei **paper.png** aus den Bildmaterialien zu diesem Übungsblatt. Der Text in der Bilddatei ist rotiert. Ihre Aufgabe ist es, mittels der Hough-Transformation den Rotationswinkel der Textzeilen zur Waagerechten zu finden und alle Textzeilen zu unterstreichen.

- Es ist hilfreich vorher mittels eines Gaußfilters `sci.gaussian_filter` (`import scipy.ndimage as sci`, siehe Hinweis unten) den Text zu Blöcken verschwimmen lassen.
- Die Addition der Gradientenlänge statt einfacher Inkrementierung im Hough-Raum sowie die Berücksichtigung der Gradientenorientierung sind empfehlenswert.
- Da die Textzeilen Geraden mit verschiedenem Abstand zum Koordinatenursprung darstellen, ist eine Aufsummierung der Spalten im Hough-Raum empfehlenswert. Dies setzt allerdings voraus, dass basierend auf der Gradientenorientierung nur ein Eintrag pro Gradient im Hough-Raum gemacht wurde.

- b) In diesem Aufgabenteil entwickeln Sie einen optischen Erkenner für Münzen. Verwenden Sie die Bilddatei **coins.png**. Die Münzen in dem Bild habe zwei verschiedene Radien (r_1 und r_2). Berechnen Sie für das Bild die Hough-Transformation für Kreise der Radien r_1 und r_2 . Markieren Sie in einem Ausgabebild alle Mittelpunkte der Münzen mit einem „+“.

- Die Konturen der Münzen lassen sich gut mit dem Betrag der Ableitungsfunktion (*Magnitude of Gradient*) $M(x_i, y_j) = \sqrt{G(x_i)^2 + G(y_j)^2}$ auf dem durch ein Gaußfilter geglätteten Bild finden.
- Verwenden Sie für die zwei verschiedenen Radii zwei Durchläufe der Hough-Transformation für Kreise mit je einem Radius.
- Verwenden Sie einen Schwellenwert für die Gradientenlänge, die Gradientenlänge zur Aktualisierung des Akkumulators, sowie die Gradientenorientierung.
- Einblenden eines roten Kreuzes an einem Punkt im Bild

```
pp.subplot(1,2,1); pp.imshow(f); pp.plot(cx, cy, 'r+', ms=20)
```

Hinweis: Sie dürfen für diese Aufgabe Funktionen des *multi-dimensional image processing*² Moduls von SciPy³ (`scipy.ndimage.*`) verwenden. Insbesondere können Sie für die Erzeugung der Gradienten den vorhandenen Sobel-Operator verwenden: `scipy.ndimage.filters.sobel(input, axis)`, wobei mit `axis` die Koordinatenachse (Richtung des Sobel-Operators) gewählt werden kann.

²<http://docs.scipy.org/doc/scipy/reference/ndimage.html>

³<http://www.scipy.org/SciPy>

Aufgabe 2 Generalisierte Hough-Transformation (**)

Verwenden Sie das in der Vorlesung beschriebene Verfahren der generalisierten Hough-Transformation um das Vorkommen des Modells (in **generalhoughmodel.png**) im Testbild (**generalhoughtest.png**) zu finden. Es genügt, als Resultat das Hough-Bild darzustellen.

Gehen Sie wie folgt vor:

- R-Tabelle für das Modell erstellen. Dazu Sobel-Operatoren anwenden und einen geeigneten Schwellenwert für die Gradientenlänge des Modellbilds festlegen. Anhand der Vordergrundpixel Mittelpunkt des Modells als Referenzpunkt berechnen. Für jedes Vordergrundpixel dessen Gradientenorientierung und den Differenzvektor zum Referenzpunkt berechnen. Gradientenorientierung quantisieren und in der entsprechenden Zeile der R-Tabelle eintragen, d.h. in der zugehörigen Liste anhängen.
- Suchen nach dem Modell im Testbild: Dazu Sobel-Operatoren anwenden und einen geeigneten Schwellenwert für die Gradientenlänge des Modellbilds festlegen. Für jedes Vordergrundpixel die Gradientenorientierung quantisieren und die entsprechende Zeile der R-Tabelle auslesen. Für alle Einträge in der zur quantisierten Orientierung gehörenden Liste die entsprechenden Punkte im Hough-Bild inkrementieren.
- Das Hough-Resultatbild anzeigen.