

Hough-Transformation

Silas Ueberschaer
Seminar Computer Vision WS 2020/21
Universität Hamburg, Fachbereich Informatik

Gliederung

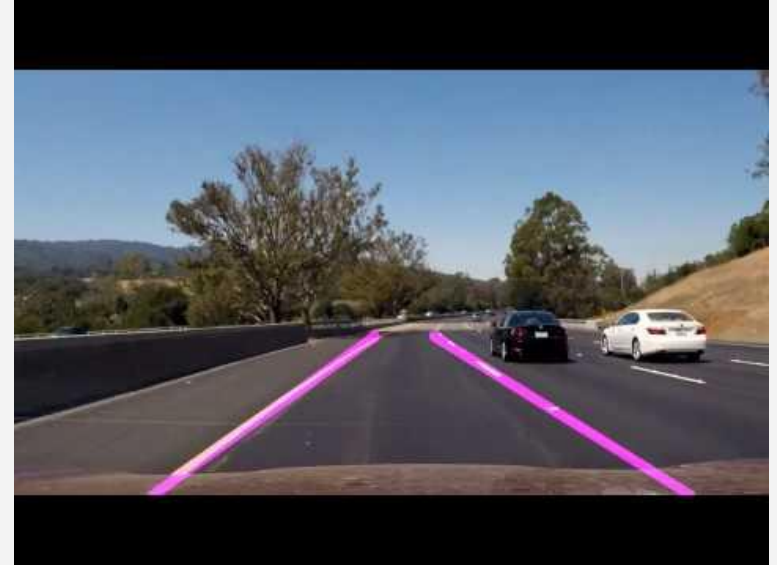
1. **Motivation:** Wozu sollte man Linien in Bildern erkennen?
2. **Einleitung:** Woher kommt die Transformation?
3. **Geradengleichung:** Was sind Vorteile und warum funktioniert sie nicht?
4. **Hough-Transformation:** Wie funktioniert die Transformation?
5. **PPHT:** Was kann sie besser und wo liegen die Unterschiede?
6. **Zusammenfassung**

Gliederung

1. **Motivation:** Wozu sollte man Linien in Bildern erkennen?
2. **Einleitung:** Woher kommt die Transformation?
3. **Geradengleichung:** Was sind Vorteile und Warum funktioniert sie nicht?
4. **Hough-Transformation:** Wie funktioniert die Transformation?
5. **PPHT:** Was kann sie besser und wo liegen die Unterschiede
6. **Zusammenfassung**

Motivation

- Was ist eine Linie in einem Bild?
- Information vom Bild minimieren
- Weiterverarbeitung der Linien
 - Autonomes fahren, Landebahn erkennung,...



[Quelle: youtube.com]

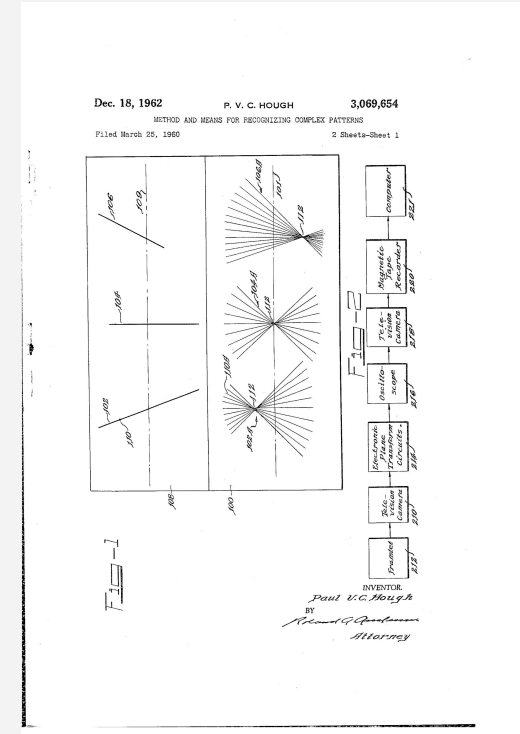
Gliederung

1. **Motivation:** Wozu sollte man Linien in Bildern erkennen?
2. **Einleitung:** Woher kommt die Transformation?
3. **Geradengleichung:** Was sind Vorteile und Warum funktioniert sie nicht?
4. **Hough-Transformation:** Wie funktioniert die Transformation?
5. **PPHT:** Was kann sie besser und wo liegen die Unterschiede
6. **Zusammenfassung**

Erfindung

- Paul V. C. Hough
- Grundstein der Transformation
- Problem Vertikale Kanten
- Richard O. Duda und Peter E. Hart

$$\rho = x_i \cdot \cos(\theta) + y_i \cdot \sin(\theta)$$



Dec. 18, 1962

P. V. C. HOUGH

3,069,654

METHOD AND MEANS FOR RECOGNIZING COMPLEX PATTERNS
Filed March 25, 1960

2 Sheets-Sheet 1

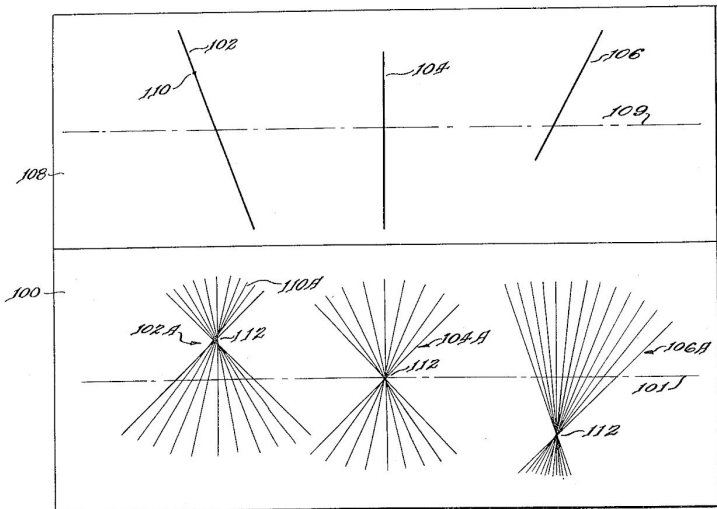


Fig. 1

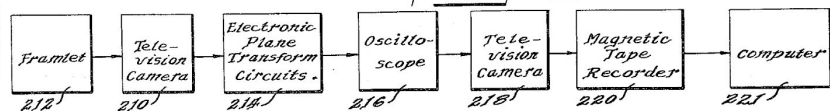


Fig. 2

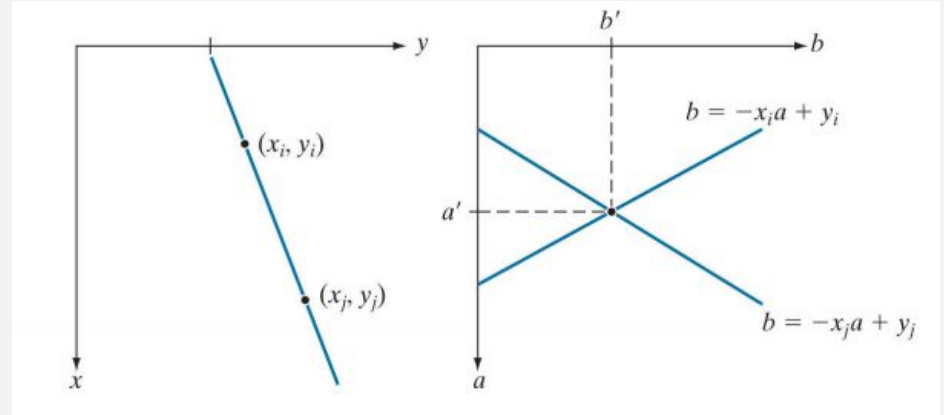
INVENTOR
Paul V.C. Hough
BY
Attorney

Gliederung

1. **Motivation:** Wozu sollte man Linien in Bildern erkennen?
2. **Einleitung:** Woher kommt die Transformation?
3. **Geradengleichung:** Was sind Vorteile und Warum funktioniert sie nicht?
4. **Hough-Transformation:** Wie funktioniert die Transformation?
5. **PPHT:** Was kann sie besser und wo liegen die Unterschiede
6. **Zusammenfassung**

Geradengleichung

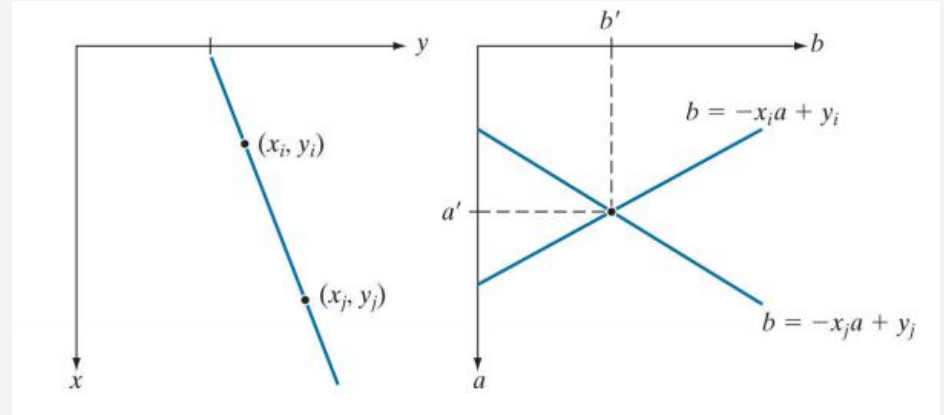
- $y=a*x+b$ ($y=m*x+b$)
- Schnittstelle gibt m und b
- Gleichung aufstellen
- Problem: Vertikale Kanten



[Gonzalez/Woods, 2018]

Geradengleichung

- $y = a \cdot x + b$ ($y = m \cdot x + b$)
- Schnittstelle gibt m und b
- Gleichung aufstellen
- Problem: Vertikale Kanten
 - $-x_j \cdot a + y_j = -x_i \cdot a + y_i$ (Gleichstellen)
 - $a = (y_i - y_j) / (x_i - x_j)$ (nach a umstellen)



[Gonzalez/Woods, 2018]

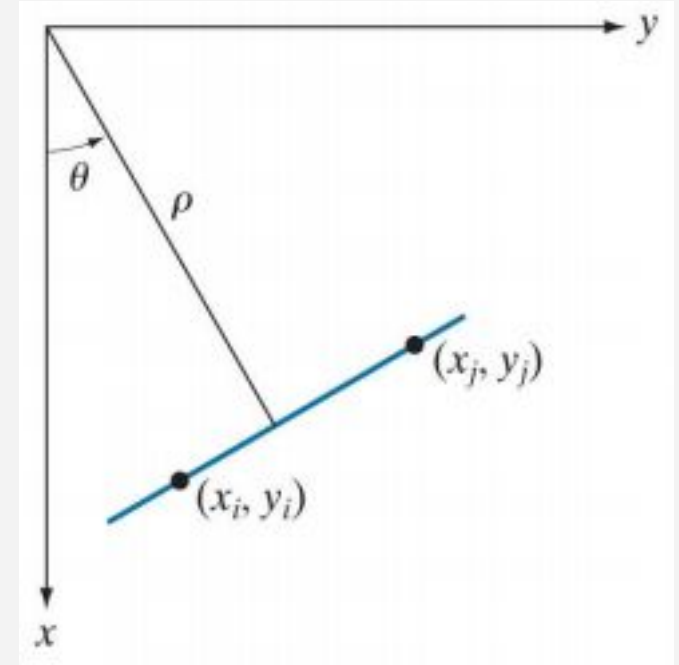
Gliederung

1. **Motivation:** Wozu sollte man Linien in Bildern erkennen?
2. **Einleitung:** Woher kommt die Transformation?
3. **Geradengleichung:** Was sind Vorteile und Warum funktioniert sie nicht?
4. **Hough-Transformation:** Wie funktioniert die Transformation?
5. **PPHT:** Was kann sie besser und wo liegen die Unterschiede
6. **Zusammenfassung**

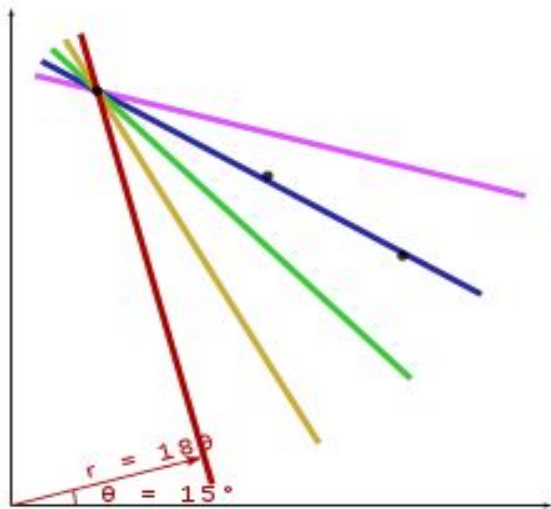
Hough-Transformation

- ρ = Abstand vom Ursprung
- θ = Winkel zwischen X-Achse und ρ

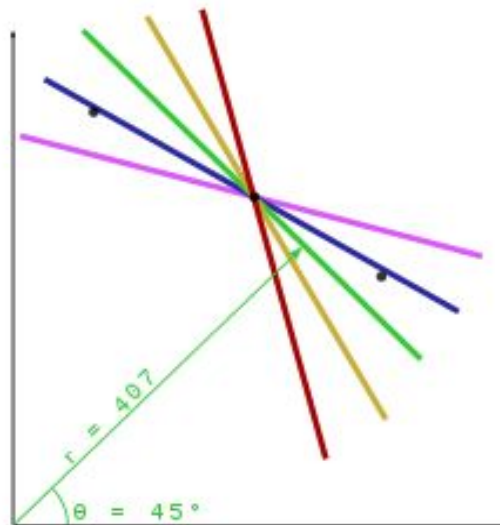
$$\rho = x_i \cdot \cos(\theta) + y_i \cdot \sin(\theta)$$



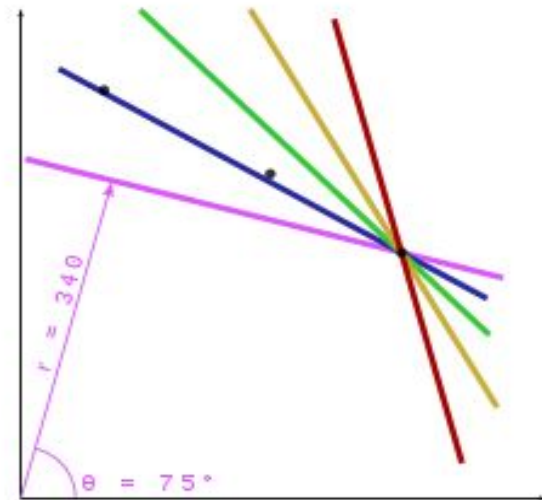
[Gonzalez/Woods, 2018]



θ	r
15	189.0
30	282.0
45	355.7
60	407.3
75	429.4



θ	r
15	318.5
30	376.8
45	407.3
60	409.8
75	385.3

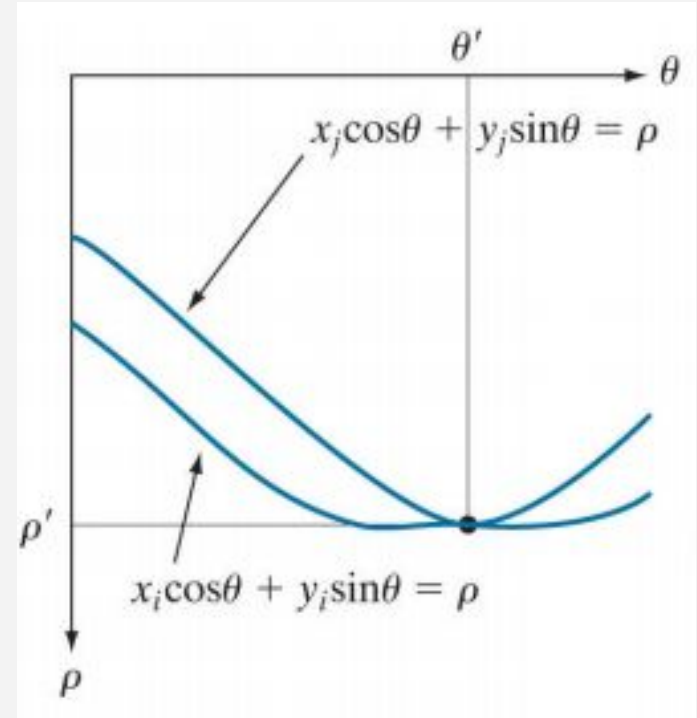


θ	r
15	419.0
30	443.6
45	438.4
60	402.9
75	340.1

[Wikipedia, 2020]

$\rho\theta$ -Koordinatensystem

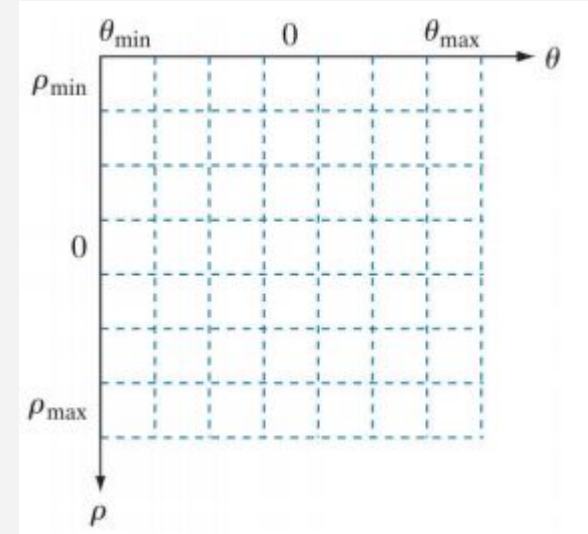
- Eintrag für jedes ρ und θ
- Ergebnis ist Sinuskurve
- Schnittpunkt
- Problem: viele Punkte



[Gonzalez/Woods, 2018]

Accumulator

- Koordinatensystem mit Werten
- Einteilung in kleine Kästchen
- Wert erhöhen
- Höhere Werte = Mehr Punkte auf Gerade



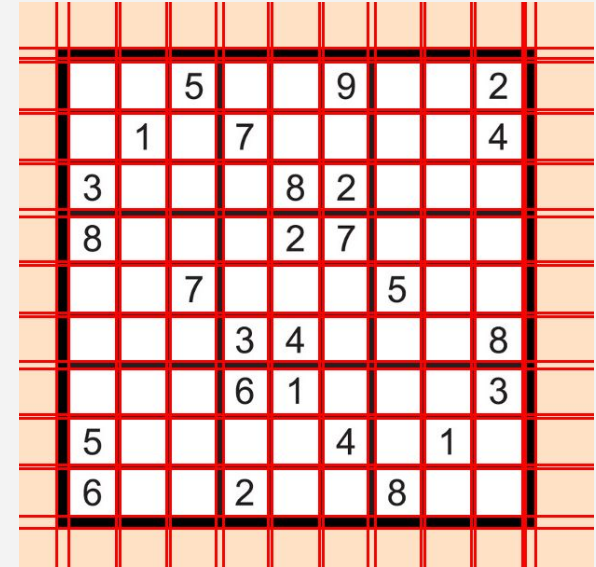
[Gonzalez/Woods, 2018]

Anwendung auf Bild

1. Kantenbild erstellen
2. Sinuskurven berechnen und eintragen
3. Schwellwert überschreiten
4. Linien malen

Problem unendliche Linien

- ρ und θ erzeugen unendliche Linien
- Problem?
- Lösung: Progressive Probabilistic Hough Transformation



[eigene Darstellung]

Gliederung

1. **Motivation:** Wozu sollte man Linien in Bildern erkennen?
2. **Einleitung:** Woher kommt die Transformation?
3. **Geradengleichung:** Was sind Vorteile und Warum funktioniert sie nicht?
4. **Hough-Transformation:** Wie funktioniert die Transformation?
5. **PPHT:** Was kann sie besser und wo liegen die Unterschiede
6. **Zusammenfassung**

PPHT

- August 1998
- Jiri Matas und Charles Galambos
- Max gap, Min line
- Schwellwert

PPHT

- Ablauf:
 - 1. zufällige Auswahl
 - 2. Sinuskurve eintragen
 - 3. aus Kantenbild entfernen
 - 4. Threshold überschritten?
 - nein -> Schritt 1
 - 5. Linie konstruiert
 - 6. Alle Pixel entfernt aus Kantenbild von Linie
 - 7. Für Pixel Accumulator Eintrag gelöscht

PPHT

- Ablauf:
 - 1. zufällige Auswahl
 - **2. Sinuskurve eintragen**
 - 3. aus Kantenbild entfernen
 - 4. Threshold überschritten?
 - nein -> Schritt 1
 - 5. Linie konstruiert
 - 6. Alle Pixel entfernt aus Kantenbild von Linie
 - 7. Für Pixel Accumulator Eintrag gelöscht

PPHT

- Ablauf:
 - 1. zufällige Auswahl
 - 2. Sinuskurve eintragen
 - **3. aus Kantenbild entfernen**
 - 4. Threshold überschritten?
 - nein -> Schritt 1
 - 5. Linie konstruiert
 - 6. Alle Pixel entfernt aus Kantenbild von Linie
 - 7. Für Pixel Accumulator Eintrag gelöscht

PPHT

- Ablauf:
 - 1. zufällige Auswahl
 - 2. Sinuskurve eintragen
 - 3. aus Kantenbild entfernen
 - 4. Schwellwert überschritten?
 - nein -> Schritt 1
 - 5. Linie konstruiert
 - 6. Alle Pixel entfernt aus Kantenbild von Linie
 - 7. Für Pixel Accumulator Eintrag gelöscht

PPHT

- Ablauf:
 - 1. zufällige Auswahl
 - 2. Sinuskurve eintragen
 - 3. aus Kantenbild entfernen
 - 4. Threshold überschritten?
 - nein -> Schritt 1
 - **5. Linie konstruiert**
 - 6. Alle Pixel entfernt aus Kantenbild von Linie
 - 7. Für Pixel Accumulator Eintrag gelöscht

PPHT

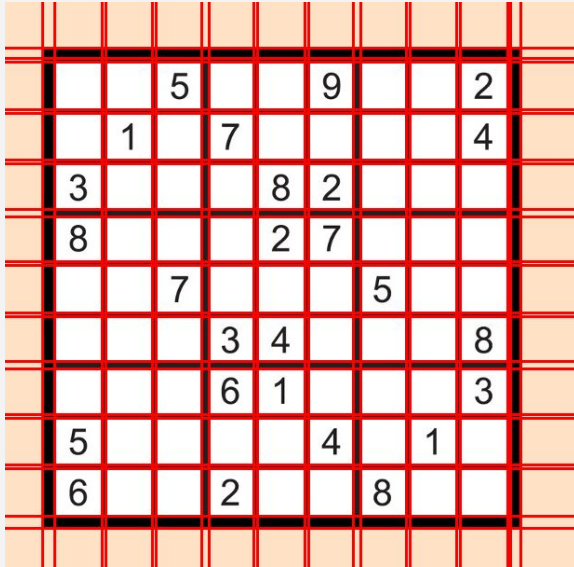
- Ablauf:
 - 1. zufällige Auswahl
 - 2. Sinuskurve eintragen
 - 3. aus Kantenbild entfernen
 - 4. Threshold überschritten?
 - nein -> Schritt 1
 - 5. Linie konstruiert
 - **6. Alle Pixel entfernt aus Kantenbild von Linie**
 - 7. Für Pixel Accumulator Eintrag gelöscht

PPHT

- Ablauf:
 - 1. zufällige Auswahl
 - 2. Sinuskurve eintragen
 - 3. aus Kantenbild entfernen
 - 4. Threshold überschritten?
 - nein -> Schritt 1
 - 5. Linie konstruiert
 - 6. Alle Pixel entfernt aus Kantenbild von Linie
 - 7. Für Pixel Accumulator Eintrag gelöscht

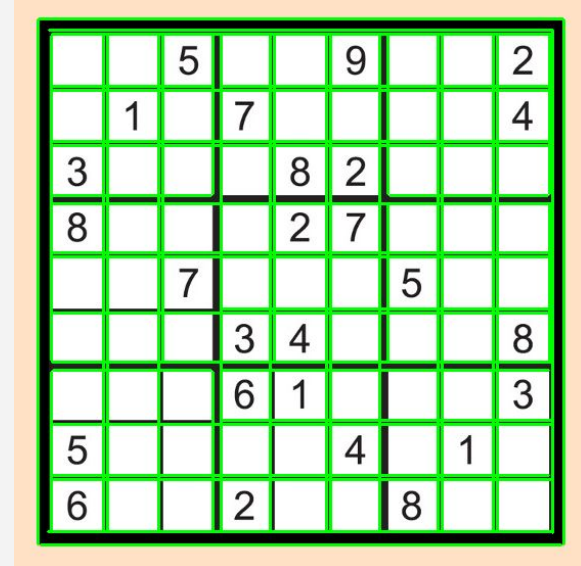
Vergleich

Hough-Transformation



[eigene Darstellung]

PPHT



[eigene Darstellung]

Gliederung

1. **Motivation:** Wozu sollte man Linien in Bildern erkennen?
2. **Einleitung:** Woher kommt die Transformation?
3. **Geradengleichung:** Was sind Vorteile und Warum funktioniert sie nicht?
4. **Hough-Transformation:** Wie funktioniert die Transformation?
5. **PPHT:** Was kann sie besser und wo liegen die Unterschiede
6. **Zusammenfassung**

Zusammenfassung

- **Motivation:** Weiterverarbeiten
 - Autonomes fahren,
- **Geradengleichung:** funktioniert nicht immer
- **Hough-Transformation:** unendliche Linien, Bruteforce
- **PPHT:** endliche Linien, nicht konstant

Vielen Dank für die Aufmerksamkeit!

Literatur

- Richard O. Duda und Peter E. Hart. “Use of the Hough transformation to detect lines and curves in pictures”. In: Communications of the ACM 15 (1972). doi: <https://doi.org/10.1145/361237.361242>.
- Paul V C Hough. “Method and means for recognizing complex patterns”. US3069654A. März 1960.
- C. Galambosy J. Matasy und J. Kittlery. Progressive Probabilistic Hough Transform. Aug. 1988. url: <https://cmp.felk.cvut.cz/~matas/papers/matasy-bmvc98.pdf>.
- Rafael C. Gonzalez Richard E. Woods. Digital Image Processing. Rafael C. Gonzalez, 2018. isbn: 9353062985.
- Azriel Rosenfeld. Picture Processing by Computer. Academic Press, 1969. isbn: 0125973500.

Quellen

Bilder:

https://en.wikipedia.org/wiki/Hough_transform (Englisch)

Videos:

<https://www.youtube.com/watch?v=J2QgPdEhMZE>