




Computergraphik I

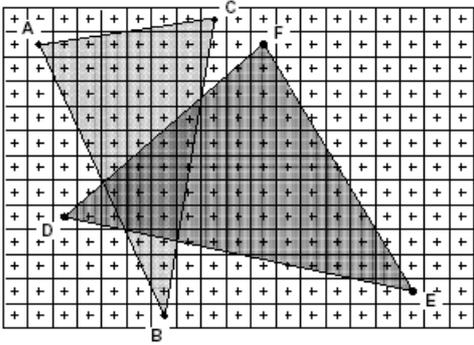
Scan Conversion of Lines

G. Zachmann
 Clausthal University, Germany
zach@tu-clausthal.de




Einordnung in die Pipeline

- Rasterisierung der Objekte in Pixel
- Ecken-Werte interpolieren (Farbe, Tiefenwert, ...)



Modell Transformation

Illumination (Shading)

Viewing Transformation (Perspective / Orthographic)

Clipping

Projektion (in Screen Space)

Scan Conversion (Rasterization)

Visibility / Display

G. Zachmann Computer-Graphik 1 – WS 10/11 Scan Conversion: Lines & Co 2

Zeichnen von Linien

- **Scan-conversion** oder **Rasterisierung**: Bestimme welche Pixel von dem Primitiv überdeckt werden
 - Der Name kommt von der Scan-Technik der Rasterdisplays
- Algorithmus ist grundlegend für 2D und 3D Computergraphiken
- (Linien zeichnen war einfacher bei Vektor Displays ;-)

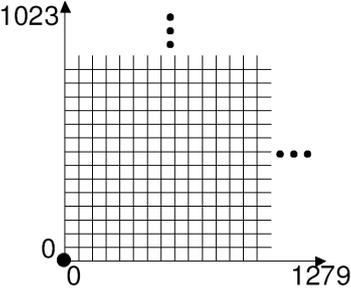


"La Linea"

G. Zachmann Computer-Graphik 1 – WS 10/11 Scan Conversion: Lines & Co 3

Das Frame-Buffer-Modell

- Adressierung als 2D-Array

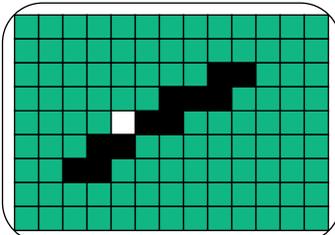


- Startadresse:
 - links oben (X11, Java AWT)
 - links unten (Open GL)

G. Zachmann Computer-Graphik 1 – WS 10/11 Scan Conversion: Lines & Co 4

Was ist ein Pixel?

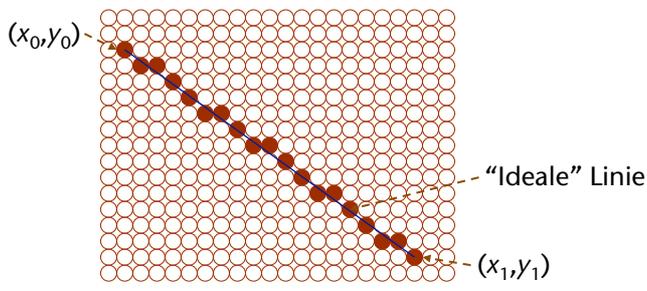
- Wir haben Ausgabegeräte mit endlicher, fester Auflösung und damit ein festes Raster
- In Wirklichkeit ist ein Pixel kein Punkt sondern ein kleines Quadrat
 - Daher der Name: "*pixel*" = "*picture cell*" oder "*picture element*"
- Und auch das ist nur eine Näherung
- **Rasterisierung** bedeutet: ein graphisches Objekt, z.B. eine Linie, durch die Zellen eines Gitters annähern



G. Zachmann Computer-Graphik 1 – WS 10/11 Scan Conversion: Lines & Co 5

Die Ideale Linie

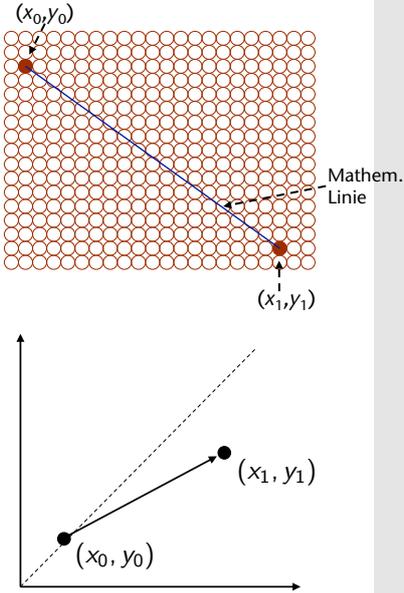
- Keine Unterbrechungen (diagonale Schritte sind erlaubt)
- Einheitliche Stärke und Helligkeit
- Fehlerfreiheit (setze nur den nächsten Pixel an der idealen Linie)
- Geschwindigkeit (wie schnell kann die Linie gezeichnet werden?)
- Invarianz gegenüber Zeichenrichtung



G. Zachmann Computer-Graphik 1 – WS 10/11 Scan Conversion: Lines & Co 6

Die Aufgabe

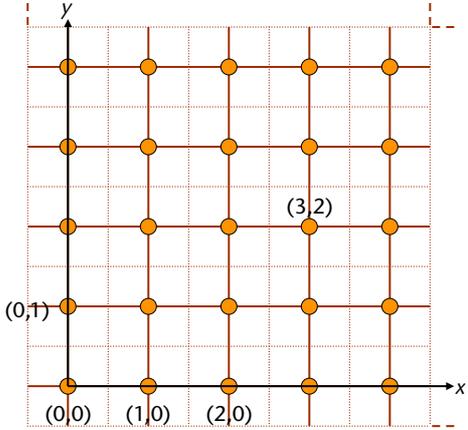
- Gegeben: Endpunktkoordinaten einer Linie
- Vereinfachungen:
 - Ganzzahlige Koordinaten
 - Geradengleichung $y = mx + b$
 - Mit $0 \leq m \leq 1$ und $x_0 < x_1$
 - Alle übrigen Fälle bekommt man durch Vertauschen / Spiegeln
- Ausgabe: Folge von Pixeln (= rasterisierte Linie)



G. Zachmann Computer-Graphik 1 – WS 10/11 Scan Conversion: Lines & Co 7

Bildschirmkoordinaten im Frame-Buffer-Modell

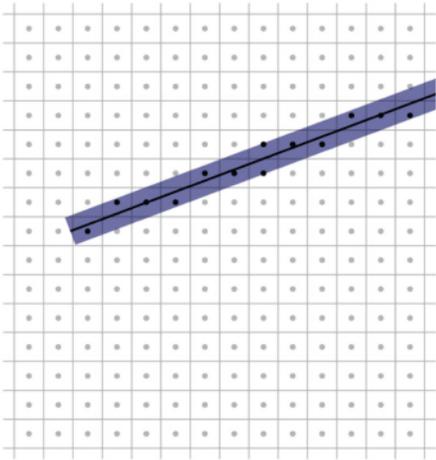
- Wir verwenden folgende 2D Bildschirmkoordinaten
 - Ganzzahlige Werte für Mittelpunkte der Pixel
 - Senkrecht = Y-Achse, Horizontal = X-Achse



G. Zachmann Computer-Graphik 1 – WS 10/11 Scan Conversion: Lines & Co 8

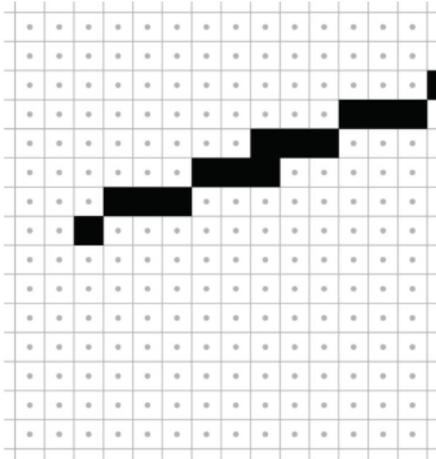
Erster Versuch der Scan-Konv. einer Linie

- Betrachte Linie als schmales Rechteck
- Zeichne alle Pixel, deren Zentrum im Inneren liegt



G. Zachmann Computer-Graphik 1 – WS 10/11 Scan Conversion: Lines & Co 9

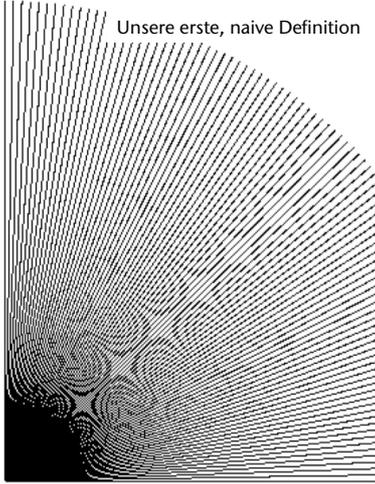
- Betrachte Linie als schmales Rechteck
- Zeichne alle Pixel, deren Zentrum im Inneren liegt
- Problem: manchmal werden vertikal übereinander liegende Pixel gesetzt → unterschiedliche scheinbare Linienstärke



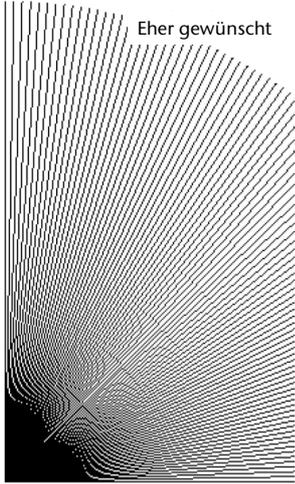
G. Zachmann Computer-Graphik 1 – WS 10/11 Scan Conversion: Lines & Co 10

■ Problem:

Unsere erste, naive Definition



Eher gewünscht



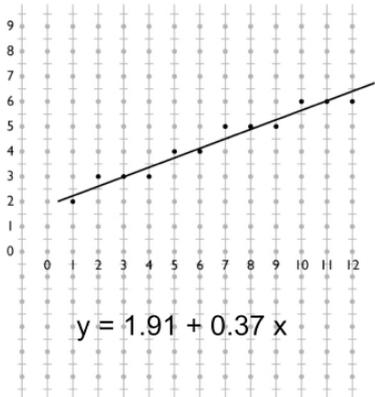
G. Zachmann Computer-Graphik 1 – WS 10/11 Scan Conversion: Lines & Co 11

■ Erster einfacher Algorithmus

■ Einfacher Algorithmus: werte Gleichung der Linie pro Spalte (pro x-Koord.) ein Mal aus

```
for x = ceil(x0) .. floor(x1):
  y = b + m*x
  setPixel( x, round(y) )
```

■ Problem: Floating-Point, Mult. und Runden sind (rel.) langsam



G. Zachmann Computer-Graphik 1 – WS 10/11 Scan Conversion: Lines & Co 12

Enter Bresenham

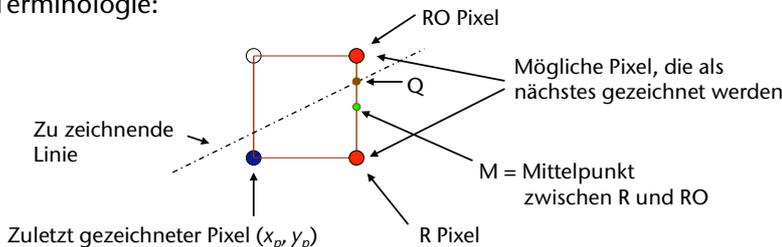


Clip from Bresenham's Keynote at WSCG'03

G. Zachmann Computer-Graphik 1 – WS 10/11 Scan Conversion: Lines & Co 13

Der Midpoint-Algorithmus

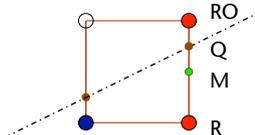
- Bei jedem X-Schritt gibt es nur zwei Möglichkeiten für die Y-Koordinate:
 - entweder bleibt die Y-Koord gleich;
 - oder die Y-Koord erhöht sich um genau 1 Pixel
- Terminologie:


- $Q = \text{Schnittpunkt der Linie mit Gridline } x_p + 1$

G. Zachmann Computer-Graphik 1 – WS 10/11 Scan Conversion: Lines & Co 14

Zwei Varianten des Linien-Zeichen-Algos

- Ursprünglich "Bresenham-Algorithmus" [1962]:
 - Bestimme Distanz zwischen RO und Q und zwischen R und Q
 - Bestimme Differenz zwischen den Distanzen
 - Vorzeichen des Ergebnisses legt fest, welcher Pixel eingefärbt wird
- Heute Midpoint-Algo:
 - Bestimme, auf welcher Seite der Linie liegt Mittelpunkt M
 - M = oberhalb → färbe Pixel R
 - M = unterhalb → färbe Pixel RO



G. Zachmann Computer-Graphik 1 – WS 10/11 Scan Conversion: Lines & Co 15

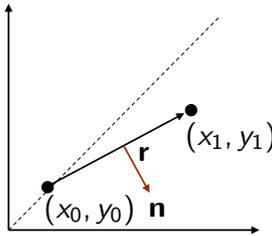
Wie kann man einfach bestimmen, auf welcher Seite der Linie man sich befindet?

- Verwende implizite Form der Linie:

$$F(x, y) := \mathbf{n} \cdot \begin{pmatrix} x \\ y \end{pmatrix} - c = 0$$

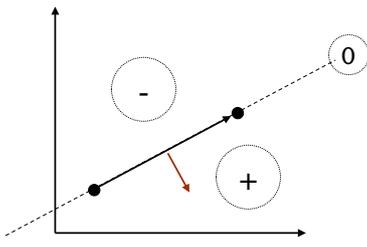
$$\mathbf{r} = \begin{pmatrix} x_1 - x_0 \\ y_1 - y_0 \end{pmatrix} = \text{Richtungsvektor}$$

$$\mathbf{n} = \begin{pmatrix} n_1 \\ n_2 \end{pmatrix} = \begin{pmatrix} y_1 - y_0 \\ x_0 - x_1 \end{pmatrix} \text{ ist senkrecht dazu}$$

$$F(x_0, y_0) = 0 \text{ liefert } c = x_0 y_1 - y_0 x_1$$


G. Zachmann Computer-Graphik 1 – WS 10/11 Scan Conversion: Lines & Co 16

- Gegeben sei (x, y) . Dann ist
 - $F(x,y) = 0$, wenn (x,y) auf der Linie liegt
 - $F(x,y) < 0$, wenn (x,y) oberhalb der Linie liegt
 - $F(x,y) > 0$, wenn (x,y) unterhalb der Linie liegt

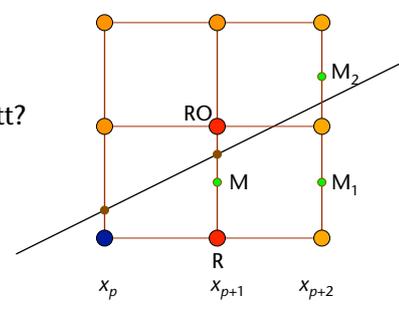


G. Zachmann Computer-Graphik 1 – WS 10/11 Scan Conversion: Lines & Co 17

- Für den Midpoint-Algorithmus, beachte das Vorzeichen von

$$F(M) = F(x_p + 1, y_p + \frac{1}{2})$$
- Definiere „Entscheidungsvariable“ d :

$$d = F(x_p + 1, y_p + \frac{1}{2})$$
 - Wenn $d > 0$, färbe RO
 - Wenn $d < 0$, färbe R
- Was ist mit dem nächsten Schritt?
- Annahme:
wir haben $d = F(M)$



G. Zachmann Computer-Graphik 1 – WS 10/11 Scan Conversion: Lines & Co 18

1. Fall: R wurde ausgewählt → nächstes M ist M_1

$$d_{old} = F(M)$$

$$= F\left(x_p + 1, y_p + \frac{1}{2}\right)$$

$$= n_1(x_p + 1) + n_2\left(y_p + \frac{1}{2}\right) + c$$

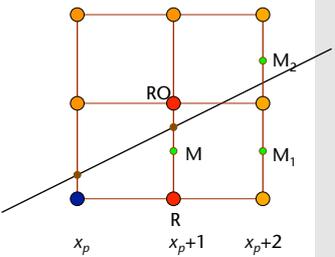
und

$$d_{new} = F(M_1)$$

$$= F\left(x_p + 2, y_p + \frac{1}{2}\right)$$

$$= n_1(x_p + 2) + n_2\left(y_p + \frac{1}{2}\right) + c$$

somit

$$d_{new} = d_{old} + n_1$$


G. Zachmann Computer-Graphik 1 – WS 10/11 Scan Conversion: Lines & Co 19

2. Fall: RO wurde ausgewählt → nächstes M ist M_2

$$d_{old} = F(M)$$

$$= F\left(x_p + 1, y_p + \frac{1}{2}\right)$$

$$= n_1(x_p + 1) + n_2\left(y_p + \frac{1}{2}\right) + c$$

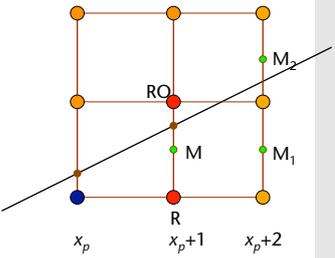
und

$$d_{new} = F(M_2)$$

$$= F\left(x_p + 2, y_p + \frac{3}{2}\right)$$

$$= n_1(x_p + 2) + n_2\left(y_p + \frac{3}{2}\right) + c$$

somit

$$d_{new} = d_{old} + n_1 + n_2$$


G. Zachmann Computer-Graphik 1 – WS 10/11 Scan Conversion: Lines & Co 20

- Pseudo-Code des Midpoint-Algo:

```

berechne n1, n2, c
x, y ← x0, y0
d ← F(M) = F(x0 + 1, y0 + 1/2) = n1 + n2/2
setze d1 ← n1, d2 ← n1 + n2
while x ≤ x1:
    zeichne Pixel (x,y)
    x += 1
    if d > 0:
        y += 1
        d += d2
    else:
        d += d1
    
```

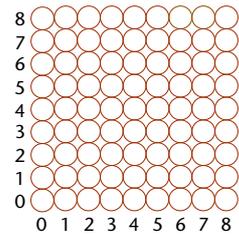
- Achtung: in obigen Pseudo-Code ist evtl. $d = \frac{k}{2}$
- Lösung: ...

Beispiel

- Zeichne Linie von (1,2) nach (5,5)

```

x, y ← x0, y0
d ← F(M) = F(x0 + 1, y0 + 1/2) = n1 + n2/2
setze d1 ← n1, d2 ← n1 + n2
while x ≤ x1:
    zeichne Pixel (x,y)
    x += 1
    if d < 0:
        y += 1
        d += d2
    else:
        d += d1
    
```



n ₁	n ₂	d ₁	d ₂	d	x	y

Demo

http://www.cs.rit.edu/~ncs/whatsInALine/whatsInALine.html

G. Zachmann Computer-Graphik 1 – WS 10/11 Scan Conversion: Lines & Co 23

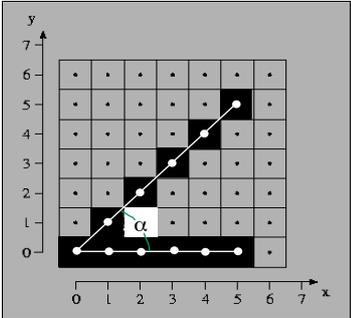
Geschwindigkeitssteigerung

- Sind rasterisierte Linien symmetrisch?
- Abhängig von der Länge:
 - Gerade # an Pixel → ja
 - Ungerade # an Pixel → ja, bis auf 1 Pixel
- Idee: zeichne von beiden Seiten [Rokne et al., 1990]
- Man kann 2 Pixel zeichnen mit:
 - 1 Vergleich
 - 1 Update der Entscheidungsvariable d
- Weiterhin: mit 1 Test kann man die nächsten 2 Pixel entscheiden [Wyvill et al., 1990]

G. Zachmann Computer-Graphik 1 – WS 10/11 Scan Conversion: Lines & Co 24

Weitere Überlegungen

- Einheitliche Stärke und Helligkeit
 - Bei gleicher Pixelzahl sind schräge Linien länger als horizontale
 - Ändere Intensität der Linie gemäß der Steigung
 - Skaliere den Grauwert um den Faktor $\cos(45^\circ - \alpha)$, $\alpha = 0^\circ \dots 45^\circ$



- Was ist bei gemusterten Linien? (gestrichelte Linie, etc.)

G. Zachmann Computer-Graphik 1 – WS 10/11 Scan Conversion: Lines & Co 25

Historische Randnotiz

- Bekannt als DDA (digital differential analyzer)



MADDIDA (Magnetic Drum Digital Differential Analyzer, Northrop Aircraft) 1952

G. Zachmann Computer-Graphik 1 – WS 10/11 Scan Conversion: Lines & Co 26