

Graphics en Game Technologie

1. Introductie en rasterizatie

Robert Belleman

Computational Science Lab

Universiteit van Amsterdam

R.G.Belleman@uva.nl

(voeg a.u.b. “[GGT]” toe aan subject)



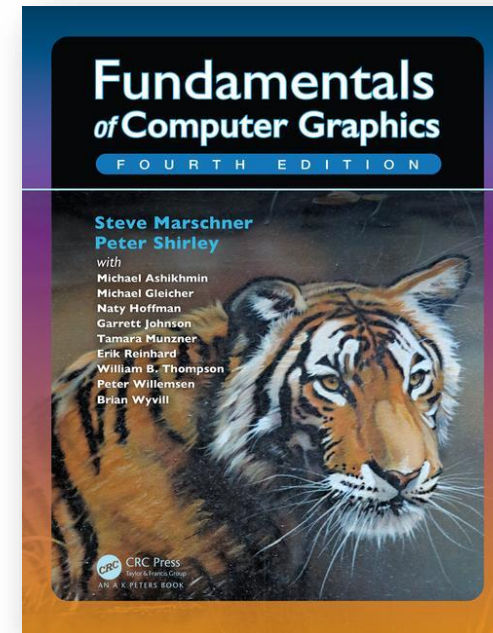
Utah teapot

Introductie tot dit vak

Inhoud, organisatie, cijfers en regels

Graphics en Game Technologie

- ▶ Docent: Robert Belleman
- ▶ Studentassistenten (“SA”):
 - ▶ Groep A: Duncan Bart
 - ▶ Groep B: Robin Slot
 - ▶ Groep C: Remco Zevering
- ▶ Boek:
 - ▶ Fundamentals of Computer Graphics (“FCG”), Steve Marschner, Peter Shirley, e.a; CRC Press/AK Peters



Colleges: eerste helft

- ▶ Collegestof week 1 t/m 3:
 1. Rasterization
 2. 2D Transformations
 3. 3D Transformations
 4. Viewing, the Graphics pipeline, curves
 5. Surface shading
 6. Raytracing
- ▶ Deeltoets in 4^e week op 26 september
 - ▶ Cijfer moet **minimaal 5.0** zijn (zo niet: hertentamen over de *hele* stof)

Colleges: tweede helft

- ▶ Collegestof week 5 t/m 7:

1. Solid modeling
2. Texture mapping
3. Data Visualization
4. Graphics Hardware
5. Interactive Graphics Applications
6. Animation

- ▶ Deeltoets in 8^e week op 24 oktober

- ▶ Cijfer moet **minimaal 5.0** zijn (zo niet: hertentamen over de *hele* stof)

Practicum

- ▶ **Lees** “Read this first!” (getting_started.pdf) op Canvas!!!
- ▶ Zes programmeeropgaven
 1. Raster graphics
 2. Lineaire transformaties
 3. Beziér curves
 4. Raytracing (twee weken)
 5. Shaders en isosurfaces
 6. Game in Unity3D
- ▶ Gemiddeld cijfer over alle practicumopgaven moet **minimaal 5.0** zijn.
 - ▶ Niet ingeleverde opgave betekent een 0
 - ▶ **Let op:** voor het practicum is géén herkansing!

Practicum

- ▶ Samenwerken **in tweetallen** aan de **hele** opdracht
 - ▶ Zoek een partner binnen je groep (A,B,C)
 - ▶ Registreer tweetal op Canvas (“People”)
- ▶ Assistentie alleen tijdens “laptopcolleges”
- ▶ Inleverdeadline: **vrijdag 23:59**
- ▶ Beoordeling door je SA **op de maandag daarna**
 - ▶ **Beide studenten** moeten aanwezig zijn
 - ▶ Wees voorbereid op vragen

Eindcijfer

- ▶ Gemiddeld tentamencijfer: $T = (DT1 + DT2) / 2$ (met $DT1, DT2 \geq 5.0$)
- ▶ Gemiddeld practicumcijfer: $P = (P1 + P2 + P3 + P4 + P5 + P6) / 6$
- ▶ Eindcijfer: $(T + P) / 2$ (met $P \geq 5.0$)

Verzin een tentamenvraag!

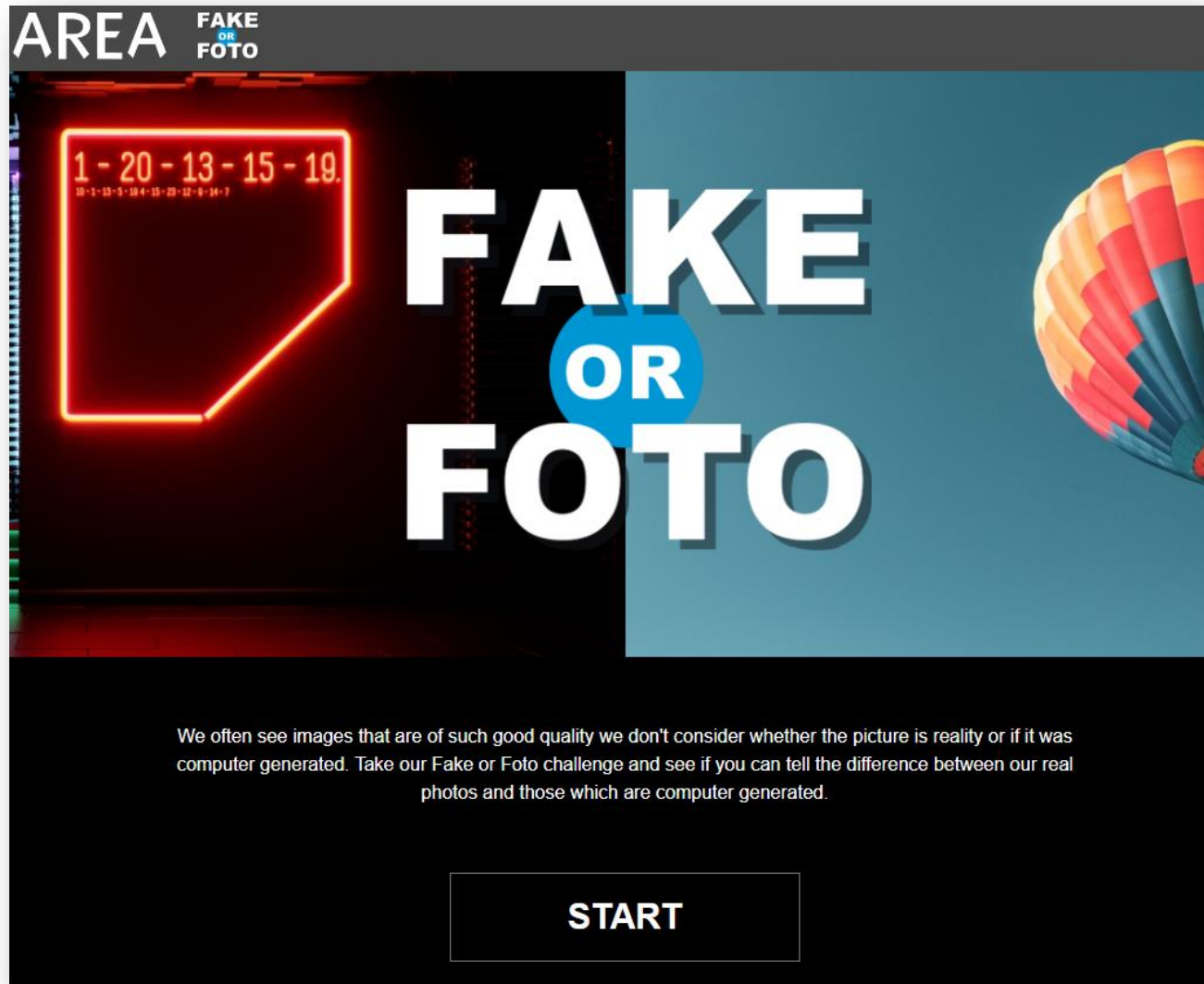
- ▶ Verzin een tentamenvraag n.a.v. dit college
- ▶ Dien in op [dit formulier](#)
- ▶ Als je vraag wordt gebruikt tijdens een (deel)tentamen ontvang jij een half punt extra op je eindcijfer voor dat (deel)tentamen (bv.: had je een 7, dan krijg je een 7,5)!*

*: Vragen die voorgaande jaren al eens op een (deel)tentamen voor dit vak zijn gesteld zijn van deze regeling uitgesloten.

Wil je meer?

- ▶ 8 weken is te kort...
- ▶ Externe bronnen:
 - ▶ [Cem Yuksel](#), University of Utah, [YouTube channel](#)
 - ▶ [Introduction to Computer Graphics](#) (Georgia Tech CS3451)
 - ▶ [3D User Interfaces](#) (Georgia Tech CS4803/8803)

Let's begin!



Computer Graphics: applications



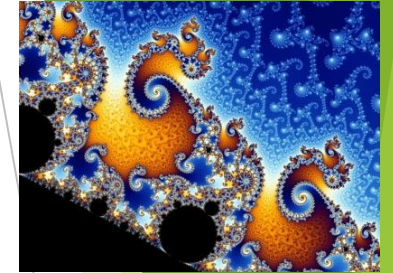
User interfaces



Games



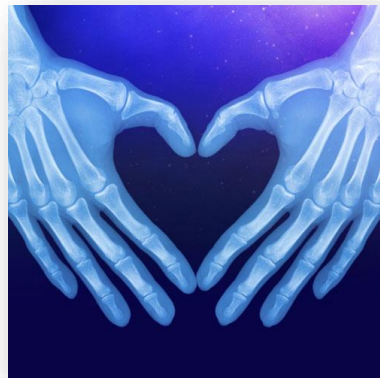
Modeling & fabrication



Art



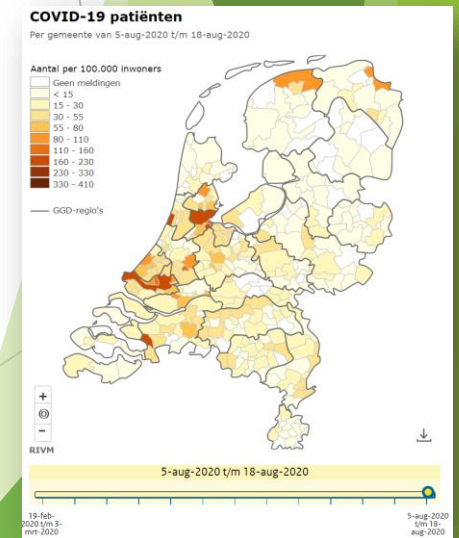
Graphics design



Imaging



Visual effects



Visualisation

Rasteralgorithmen

Computer Graphics

- ▶ **Modelling**: creëren van 3D virtuele wereld
- ▶ **Rendering**: creëren van 2D beelden van 3D modellen
 - ▶ Twee basismethoden:
 1. Projectieve methoden
 2. Ray tracing
- ▶ **Grafische pijplijn**: een reeks van transformaties die 3D modellen omzet in pixels op scherm

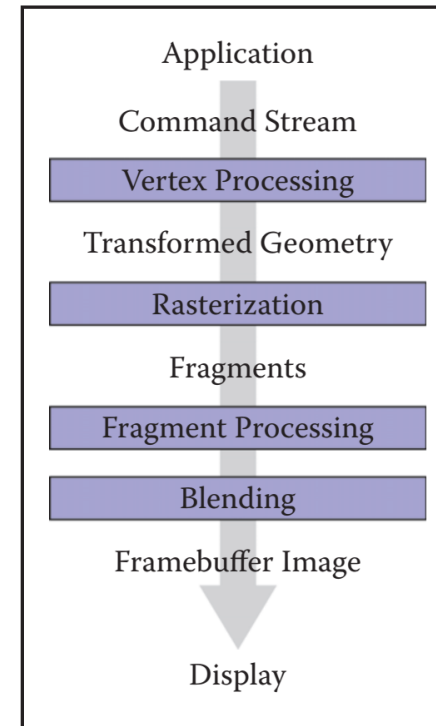


Figure 8.1. The stages of a graphics pipeline.

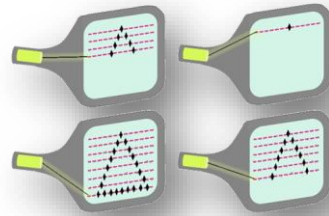
Raster Algoritmen (FCG chapter 8)

Overzicht:

1. Raster displays
2. Rasterizatie van lijnen
3. Barycentrische coördinaten
4. Rasterizatie van driehoek
5. Antialiasing

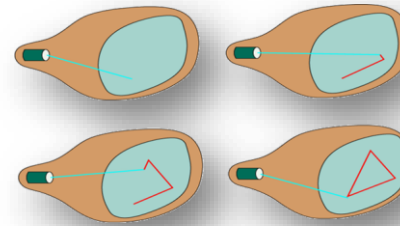
Raster versus random scan displays

Raster scan displays



Source: [techdifferences](http://techdifferences.com)

Random scan displays



Source: [javatpoint](http://javatpoint.com)

Hard copy displays



Printer



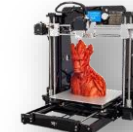
Laser printer



Pen plotter



Cutter plotter



3D printer

Transient displays



CRT monitor



LCD/TFT/LED monitor



Oscilloscope (XY mode)



Vectrex (1982)

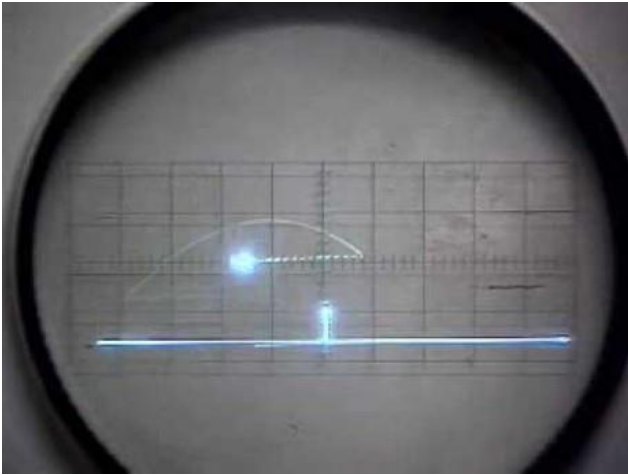


Laser beam projector

VECTOR VS RASTER



Vector graphics displays



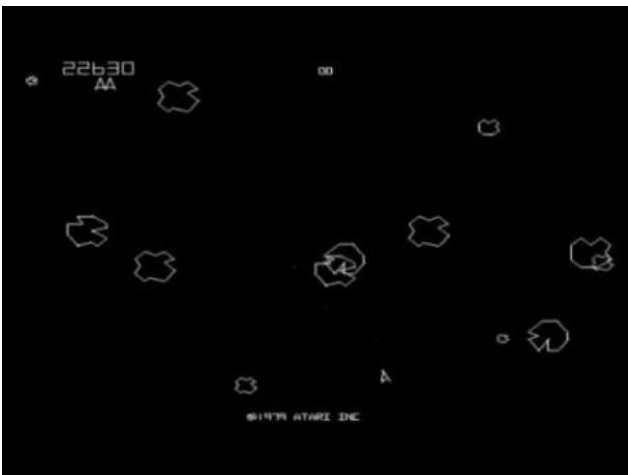
Tennis for two(1958) (source: [YouTube](#))



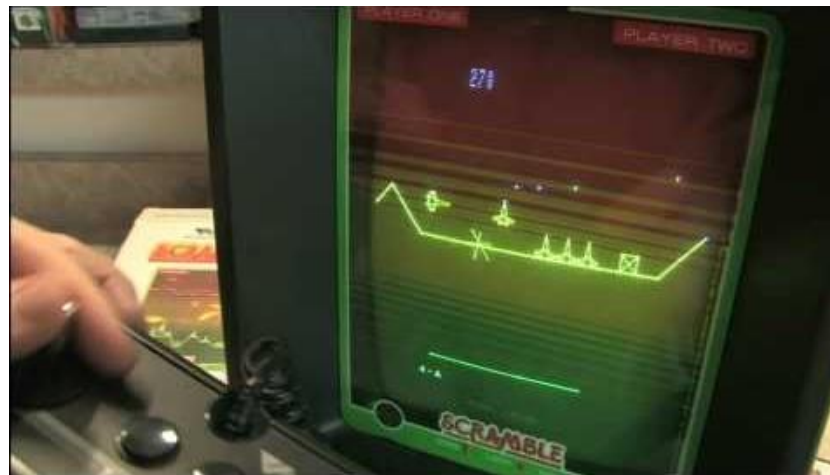
Battlezone (Atari, 1980) (source: [YouTube](#))



Star Wars (Atari, 1983) (source: [YouTube](#))



Asteroids (Atari, 1979) (source: [YouTube](#))



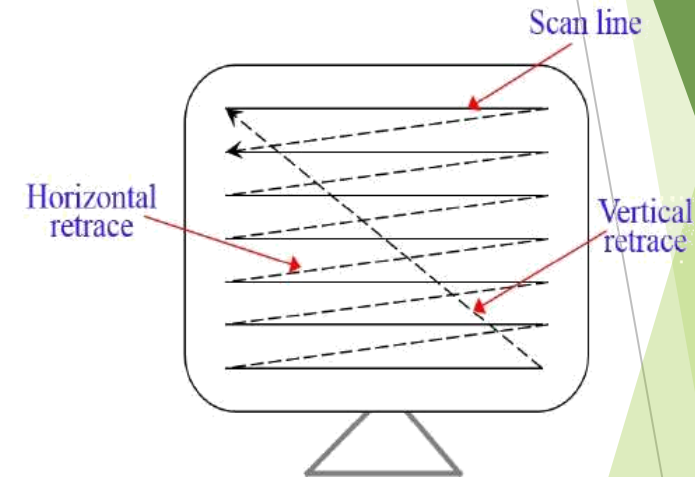
Vectrex game console (1982) (source: [YouTube](#))

Raster display

Een **raster display** is een TV of computer monitor die de “raster scanmethode” gebruikt om een beeld op een scherm te genereren

- ▶ raster display is array van pixels
- ▶ **pixel** is kleinste eenheid op scherm
- ▶ scherm opgebouwd uit punten
- ▶ **intensiteit** en kleur van elke pixel is variabel

Van 2D model naar pixels op scherm noemen we **rasterizatie**.



Schermcoördinaten

- Pixels aangegeven door indices voor kolom en rij: (i, j)
- Beeld bevat n_x kolommen en n_y rijen
- Pixel links onder $(0, 0)$, pixel rechts boven $(n_x - 1, n_y - 1)$

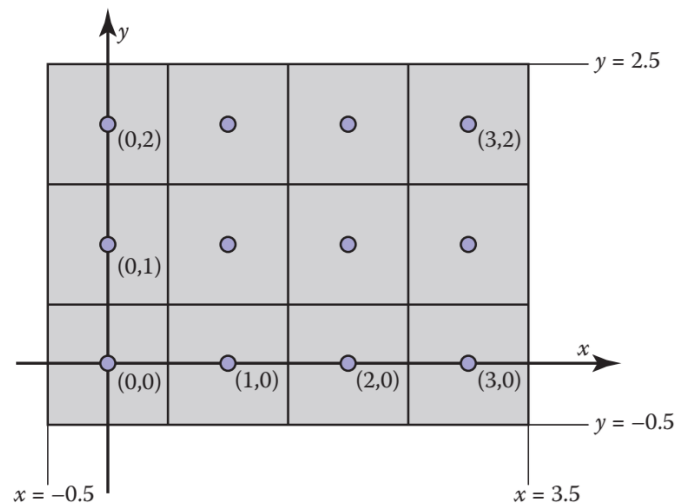


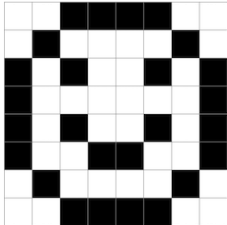
Figure 3.10. Coordinates of a four pixel \times three pixel screen. Note that in some APIs the y-axis will point downward.

0	■ (0,0)	Eerste rij
1	■ (1,0)	
2	■ (2,0)	
3	■ (3,0)	
4	■ (0,1)	Tweede rij
5	■ (1,1)	
6	■ (2,1)	
7	■ (3,1)	
8	■ (0,2)	Derde rij
9	■ (1,2)	
10	■ (2,2)	
11	■ (3,2)	

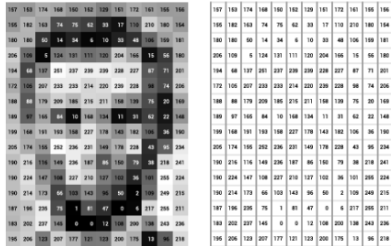
■ Eerste kolom
■ Tweede kolom
■ Derde kolom
■ Vierde kolom

Pixel format

1 1 0 0 0 0 1 1
1 0 1 1 1 1 0 1
0 1 0 1 1 0 1 0
0 1 1 1 1 1 1 0
0 1 0 1 1 0 1 0
0 1 1 0 0 1 1 0
1 0 1 1 1 1 0 1
1 1 0 0 0 0 1 1

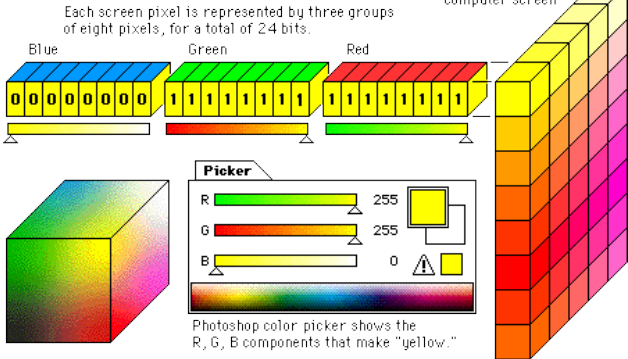


Bitmap image

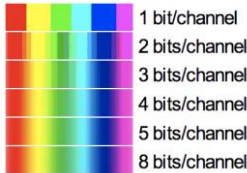


Grayscale image

24-bit "true color" displays



Colour image



+ zwart en wit

Aantal kleuren: 2^{3n} , met n aantal bits/channel

Rasterisatie van lijnen

- ▶ Lijn tussen twee eindpunten (x_0, y_0) en (x_1, y_1)
- ▶ x_0, y_0, x_1, y_1 vaak integers (pixel centra)
- ▶ Tekenen gebaseerd op vergelijking lijn, twee typen:
 - ▶ **impliciete** vergelijking van lijn
 - ▶ **parameter** voorstelling van lijn (doen we later)

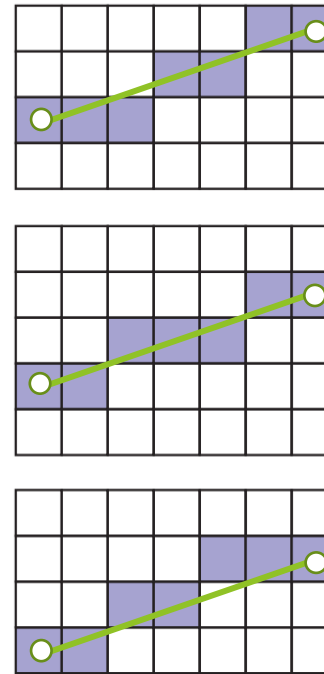
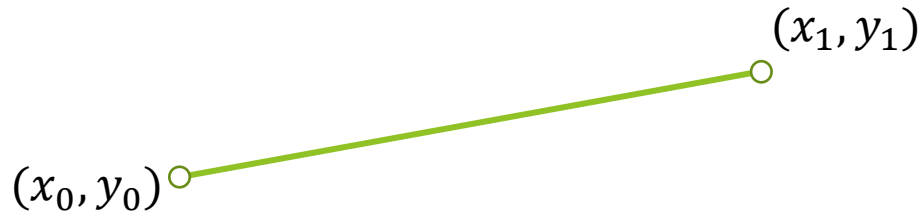


Figure 8.2. Three “reasonable” lines that go seven pixels horizontally and three pixels vertically.

,
of
?

Vergelijking van een lijn

Expliciete vergelijking lijn: $y = mx + b$

Voorbeeld: $y = 2x - 2$

Vraag: wat is m voor de lijn $x = 0$?

Impliciete vergelijking lijn: $f(x, y) = Ax + By + C = 0$

Vraag: wat is richtingscoëfficiënt m ?

Voorbeeld:

1. Lijn door oorsprong met hoek 45° : $x - y = 0$
2. Lijn door oorsprong en $(2, 3)$: $3x - 2y = 0$

Vul de tabel in en teken de lijnen

Expliciete vergelijking $y = mx + b$	Impliciete vergelijking $Ax + By + C = 0$	Normaalvergelijking $n_1x + n_2y = c$	Normaalvector (n_1, n_2)	Richtingsvector $(-n_2, n_1)$
$y = 2x - 2$	$2x - y - 2 = 0$	$\frac{2}{\sqrt{5}}x - \frac{1}{\sqrt{5}}y = \frac{2}{\sqrt{5}}$	$(2, -1)$	$(1, 2)$
$y = 2x$				
$y = -x + 3$				
	$x + 3y - 2 = 0$			
		$\frac{1}{2}\sqrt{2}x - \frac{1}{2}\sqrt{2}y = \sqrt{2}$		
$y = 0$				
	$x = 5$			

Vergelijking van een lijn door (x_0, y_0) en (x_1, y_1)

Vraag: Wat is impliciete vergelijking lijn door (x_0, y_0) en (x_1, y_1) ?

Richtingsvector $\begin{pmatrix} x_1 - x_0 \\ y_1 - y_0 \end{pmatrix}$, normaalvector $\begin{pmatrix} y_0 - y_1 \\ x_1 - x_0 \end{pmatrix}$

Impliciete vergelijking lijn is

$$f(x, y) = (y_0 - y_1)x + (x_1 - x_0)y + C = 0$$

Bepalen van C door (x_0, y_0) in te vullen:

$$f(x_0, y_0) = (y_0 - y_1)x_0 + (x_1 - x_0)y_0 + C = 0$$

$$C = -x_0y_0 + x_0y_1 - x_1y_0 + x_0y_0 = x_0y_1 - x_1y_0$$

Vergelijking van een lijn door (x_0, y_0) en (x_1, y_1)

$$f(x, y) = (y_0 - y_1)x + (x_1 - x_0)y + x_0y_1 - x_1y_0 = 0$$

Vraag: Gaat deze lijn door (x_0, y_0) en (x_1, y_1) ?

Richtingscoëfficiënt: $m = \frac{y_1 - y_0}{x_1 - x_0}$

Deze vergelijking gebruiken bij tekenen pixels op scherm

Lijnen zo dun mogelijk

Probleem: gegeven begin- en eindpunt lijn, teken zo'n dun mogelijke lijn. Welke pixels moeten getekend worden?

- ▶ $m \in (0,1]$
- ▶ Begin links, teken pixel op dezelfde hoogte of één hoger
- ▶ Precies één pixel per kolom

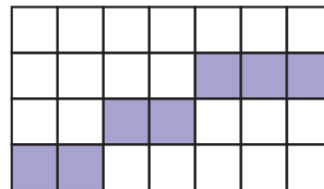
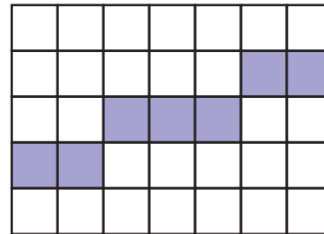
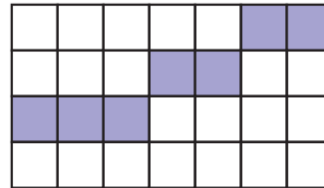
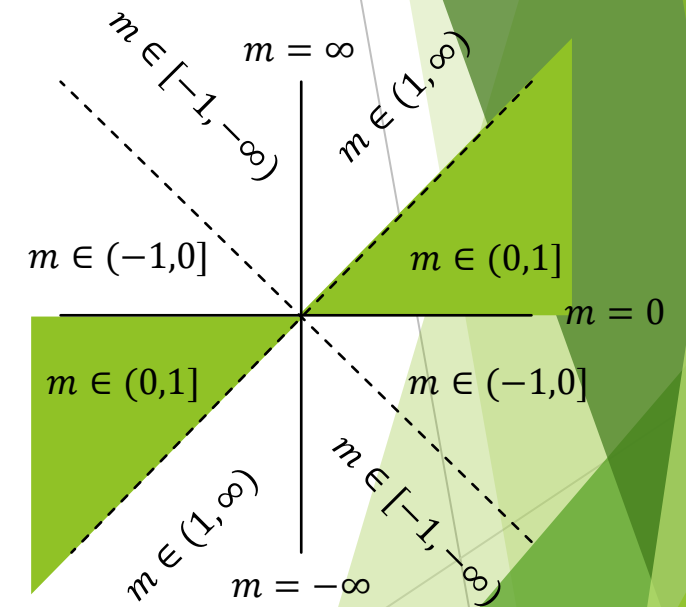


Figure 8.2. Three “reasonable” lines that go seven pixels horizontally and three pixels vertically.



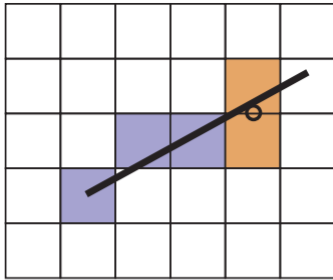
Basisvorm van Middelpunt algoritme

```
y = y0  
for x = x0 to x1 do  
    draw(x, y)  
    if (some condition) then  
        y = y + 1
```

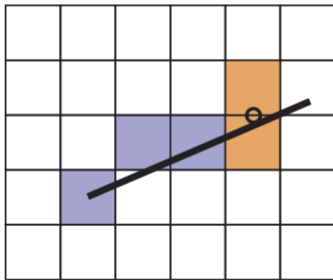
x en y zijn integers

Welke conditie moet er in de **if**-statement?

Middelpunt $(x, y + 0.5)$ tussen 2 potentiële pixels



Lijn **boven** middelpunt:
top pixel getekend



Lijn **onder** middelpunt:
bottom pixel getekend

Figure 8.3. Top: the line goes above the midpoint so the top pixel is drawn. Bottom: the line goes below the midpoint so the bottom pixel is drawn.

Boven of onder de lijn?

$$f(x, y) = (y_0 - y_1)x + (x_1 - x_0)y + x_0y_1 - x_1y_0 = 0$$

Bepalende term is $(x_1 - x_0)y$:

1. $(x_1 - x_0) > 0$ voor $m \in (0, 1]$
2. $(x_1 - x_0)y$ wordt groter als y groter
3. Dus $f(x, +\infty) > 0$

Punten boven lijn zijn **positief**

$$f(x, y + 0.5) < 0 \text{ of } f(x, y + 0.5) > 0?$$

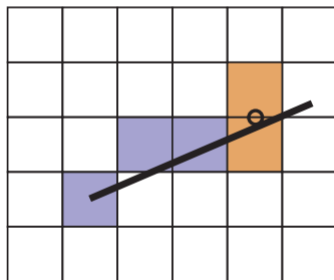
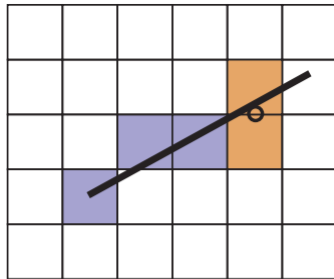


Figure 8.3. Top: the line goes above the midpoint so the top pixel is drawn. Bottom: the line goes below the midpoint so the bottom pixel is drawn.

Lijn **boven** middelpunt:

$$f(x, y + 0.5) < 0$$

top pixel getekend

$$y = y + 1$$

Lijn **onder** middelpunt:

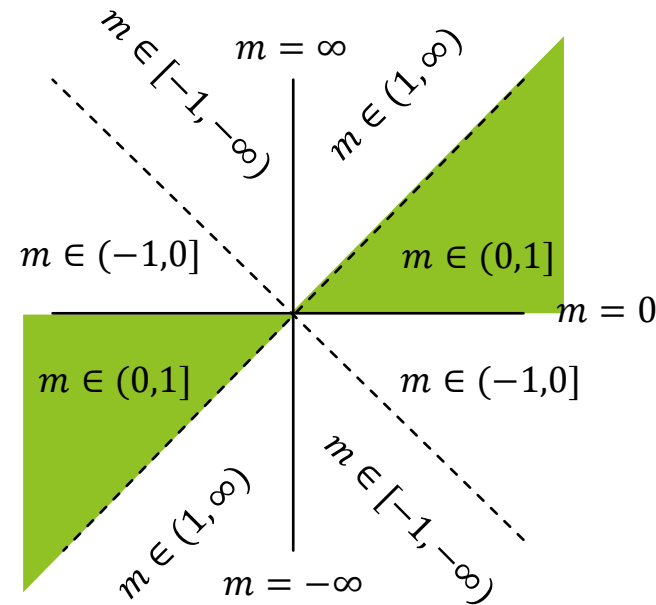
$$f(x, y + 0.5) > 0$$

bottom pixel getekend

Middelpunt algoritme

```
y = y0
for x = x0 to x1 do
  draw(x, y)
  if (f(x + 1, y + 0.5) < 0) then
    y = y + 1
```

Let op: geldt alleen voor lijnen met $m \in (0,1]$



Incrementele methode middelpunt algoritme

Incrementele methode hergebruikt berekeningen in loop

Meeste rekenwerk bepalen $f(x + 1, y + 0.5)$

Vorige berekening was:

$f(x, y - 0.5)$ of

$f(x, y + 0.5)$

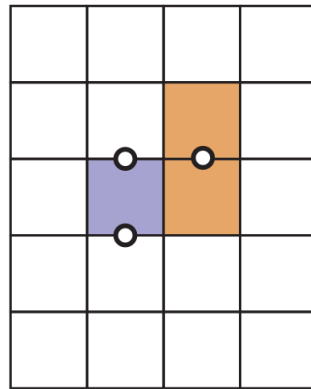


Figure 8.4. When using the decision point shown between the two orange pixels, we just drew the blue pixel, so we evaluated f at one of the two left points shown.

Berekening nieuwe $f(x + 1, y + 0.5)$

$$f(x, y) = (y_0 - y_1)x + (x_1 - x_0)y + x_0y_1 - x_1y_0 = 0$$

Berekening van f voor twee pixels in kolom:

- ▶ $f(x + 1, y) = f(x, y) + (y_0 - y_1)$
- ▶ $f(x + 1, y + 1) = f(x, y) + (y_0 - y_1) + (x_1 - x_0)$

Daaruit volgt:

1. $f(x + 1, y + 0.5) = f(x, y + 0.5) + (y_0 - y_1)$
2. $f(x + 1, y + 0.5) = f(x, y - 0.5) + (y_0 - y_1) + (x_1 - x_0)$

Beslissingsvariabele **d**

Definieer: **d** = $f(x + 1, y + 0.5)$

In **if**-statement beslissingsvariabele **d**:

- ▶ startwaarde $y = y_0$
startwaarde **d** = $f(x_0 + 1, y_0 + 0.5)$
- ▶ als **d** < 0, $d_{new} = d_{old} + (x_1 - x_0) + (y_0 - y_1)$
- ▶ als **d** >= 0, $d_{new} = d_{old} + (y_0 - y_1)$

Incrementeel middelpunt algoritme

```
y = y0
d = f(x0 + 1, y0 + 0.5)
for x = x0 to x1 do
    draw(x, y)
    if (d < 0) then
        y = y + 1
        d = d + (x1 - x0) + (y0 - y1)
    else
        d = d + (y0 - y1)
```

Incrementeel integer algoritme

Algoritme gebruikt bijna alleen maar **integers**, behalve voor startwaarde van **d** = $f(x_0 + 1, y_0 + 0.5)$

Zelfs **d** kan **integer** gemaakt worden (zie FCG)

Middelpunt algoritme meest efficiënt als:

1. incrementeel
2. alleen integer operaties

Verzin een tentamenvraag!

- ▶ Verzin een tentamenvraag n.a.v. dit college
- ▶ Dien in op [dit formulier](#)
- ▶ Als je vraag wordt gebruikt tijdens een (deel)tentamen ontvang jij een half punt extra op je eindcijfer voor dat (deel)tentamen (bv.: had je een 7, dan krijg je een 7,5)!*

*: Vragen die voorgaande jaren al eens op een (deel)tentamen voor dit vak zijn gesteld zijn van deze regeling uitgesloten.