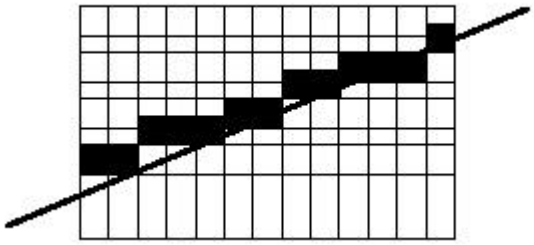
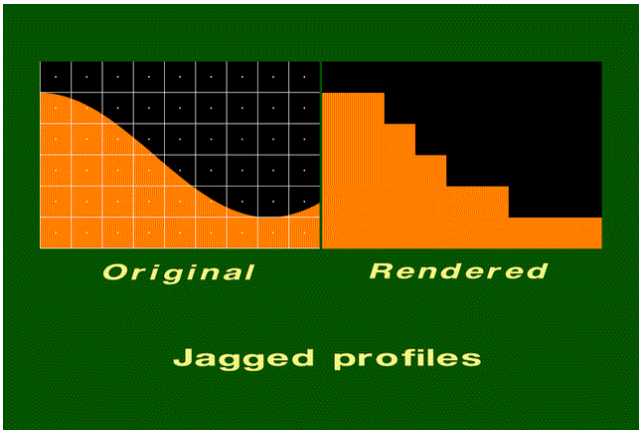
Aliasing

- Inherent property of raster displays
- problem:
    - staircases or jaggies
    - result of “all or nothing” approach (pixels are ON or OFF)
    - discrete sampling of a continuous primitive

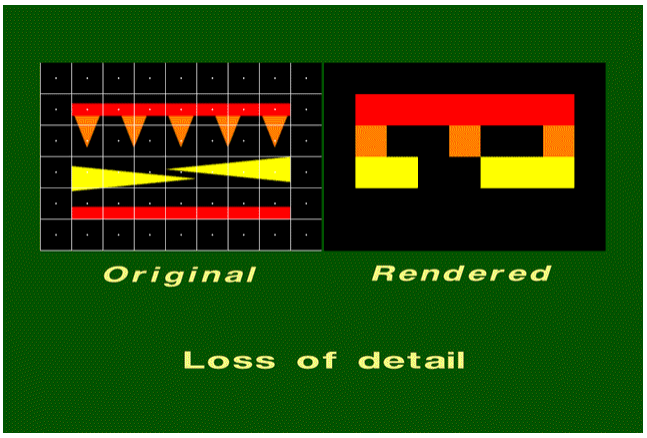


Errors caused by aliasing

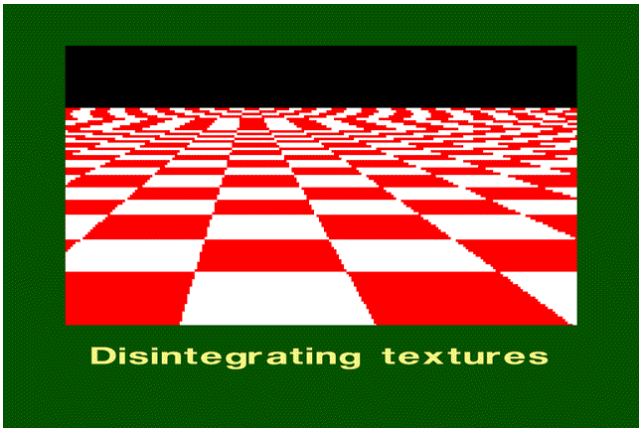
Jagged profiles



Improperly rendered detail:



Disintegrating textures:



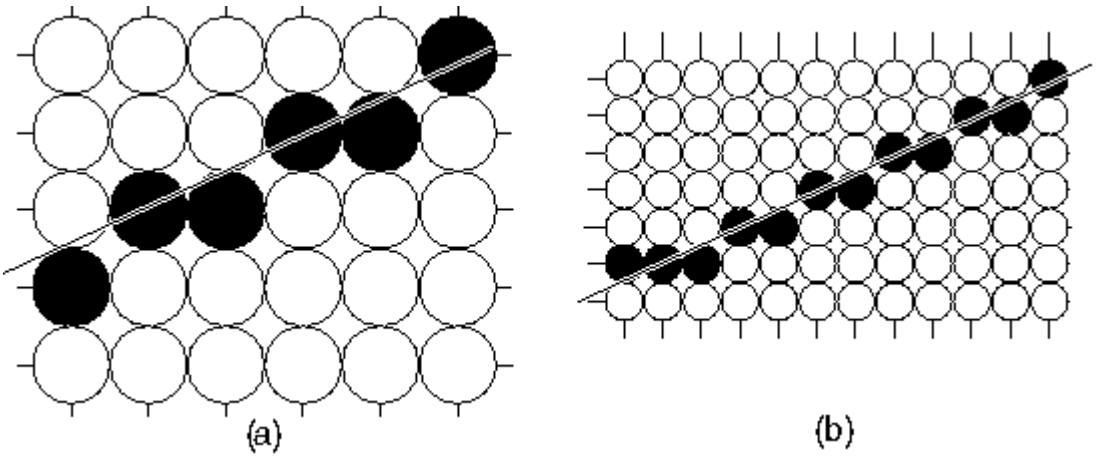
Antialiasing:

- involves “blurring” the edges to “smooth” the image
  - in case of a black rectangle on a white background, the sharp transition is softened by using a mix of gray pixels near the rectangles border
  - from afar, the eye tends to blend them together and thus see a smoother edge

Three main classes of antialiasing algorithms:

1. increase resolution
2. prefiltering
3. postfiltering

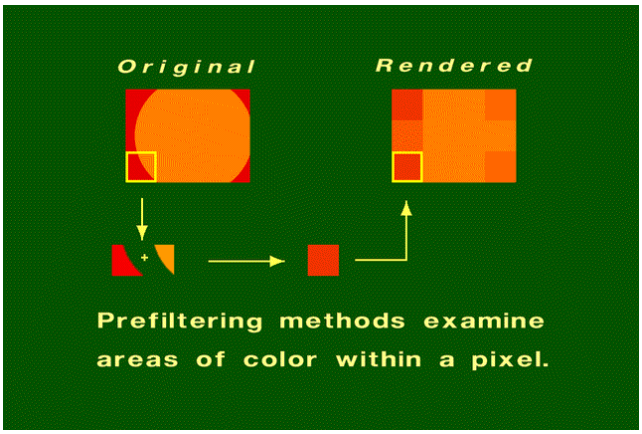
1. Increase resolution of display



- twice as many jaggies, but all are ½ the size so they are less noticeable
- BUT: expensive since 4X the memory, extra time for scan conversion …

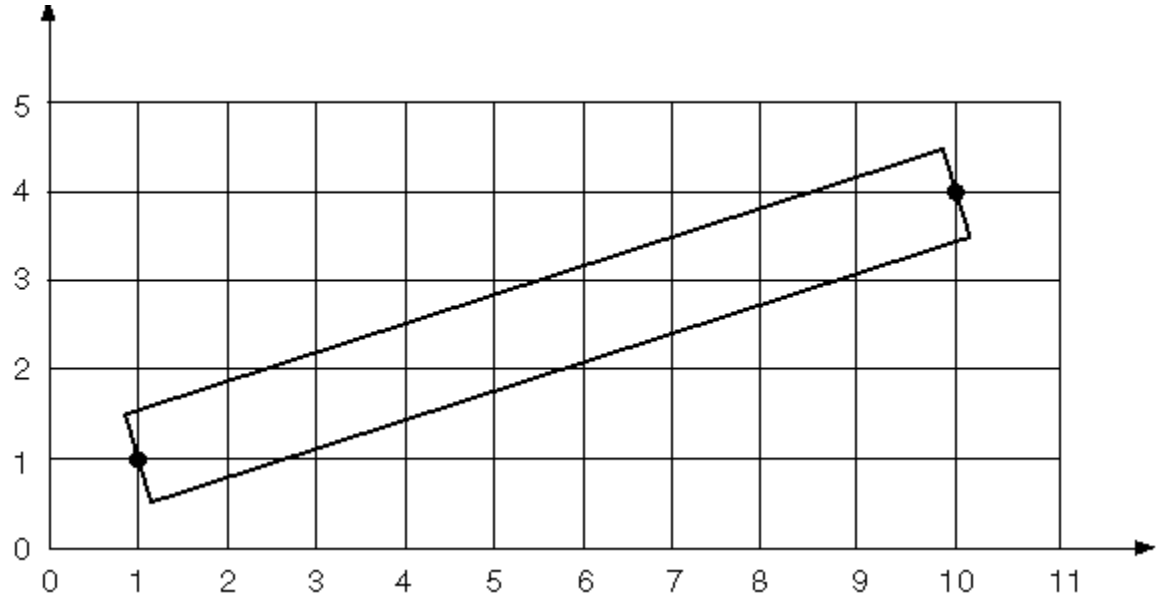
2. Prefiltering

- treat a pixel as an area and compute pixel colour based on the overlap of the scene's objects with a pixel's area

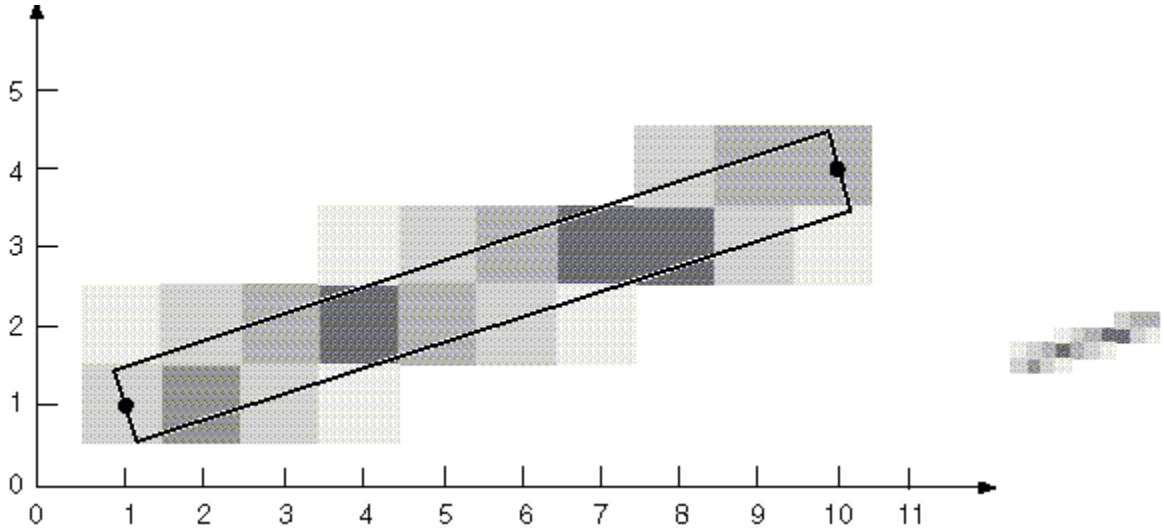


Unweighted area sampling

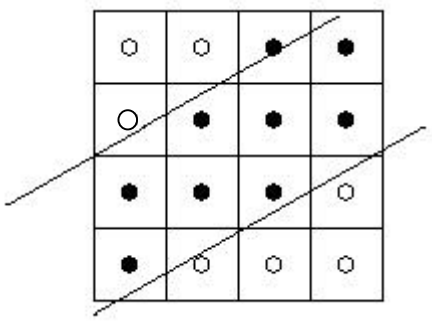
- Pitteway and Watkinson
- the intensity of a pixel is proportional to how much is covered by the line.



- so, determine the percentage of a pixel that is covered (P)
- if I is the full intensity then intensity of the pixel will be PxI.



- can approximation the area of overlap by subdividing into a finer grid of rectangular subpixels, then counting the number of subpixels inside the line



P = 9/16

Properties of unweighted sampling

1. intensity of a pixel decreases as the area of overlap decreases
2. a primitive cannot influence the intensity of a pixel which it does not intersect
3. equal areas contribute equal intensity

Original image:

Prefiltered image:

- along the characters border the colours are a mixture of the foreground and background colours

---

Supersampling

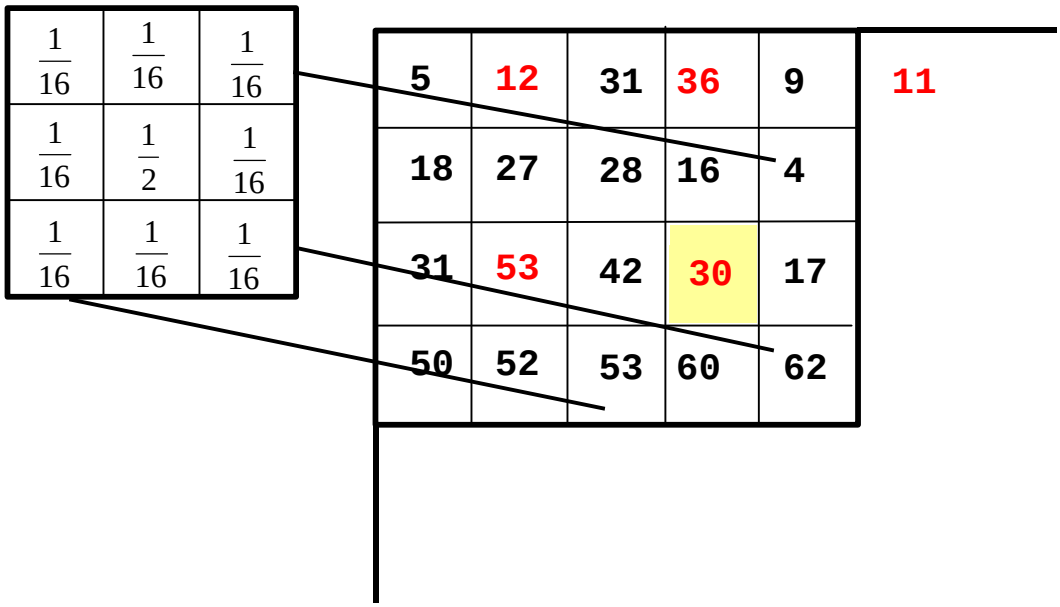
- try to improve the computational burden of prefiltering by sampling less densely, but still more than one sample per pixel
- taking more intensity samples of the scene than are displayed
- several samples are averaged together to compute each display pixel value

- Two steps in supersampling:
  1. sample the scene at n times the display resolution
  2. the colour of each pixel in the rendered image will be an average of several samples

3. Postfiltering

- computes each display pixel as a “weighted” average of an appropriate set of neighbouring samples
- similar to supersampling except a filter is used to weight each neighbouring sample

Example:



$$\frac{1}{2}(30)+\frac{1}{16}(28+16+4+42+17+53+60+62)=32.625$$

Common masks used:

(a)

1/8

0	1	0
1	4	1
0	1	0

(b)

1/16

1	2	1
2	4	2
1	2	1

(c)

1/81

1	2	3	2	1
2	4	6	4	2
3	6	9	6	3
2	4	6	4	2
1	2	3	2	1

(b) & (c) are approximations to the “Bartlett window”: the weights grow linearly from the edges towards the center