Antialiasing

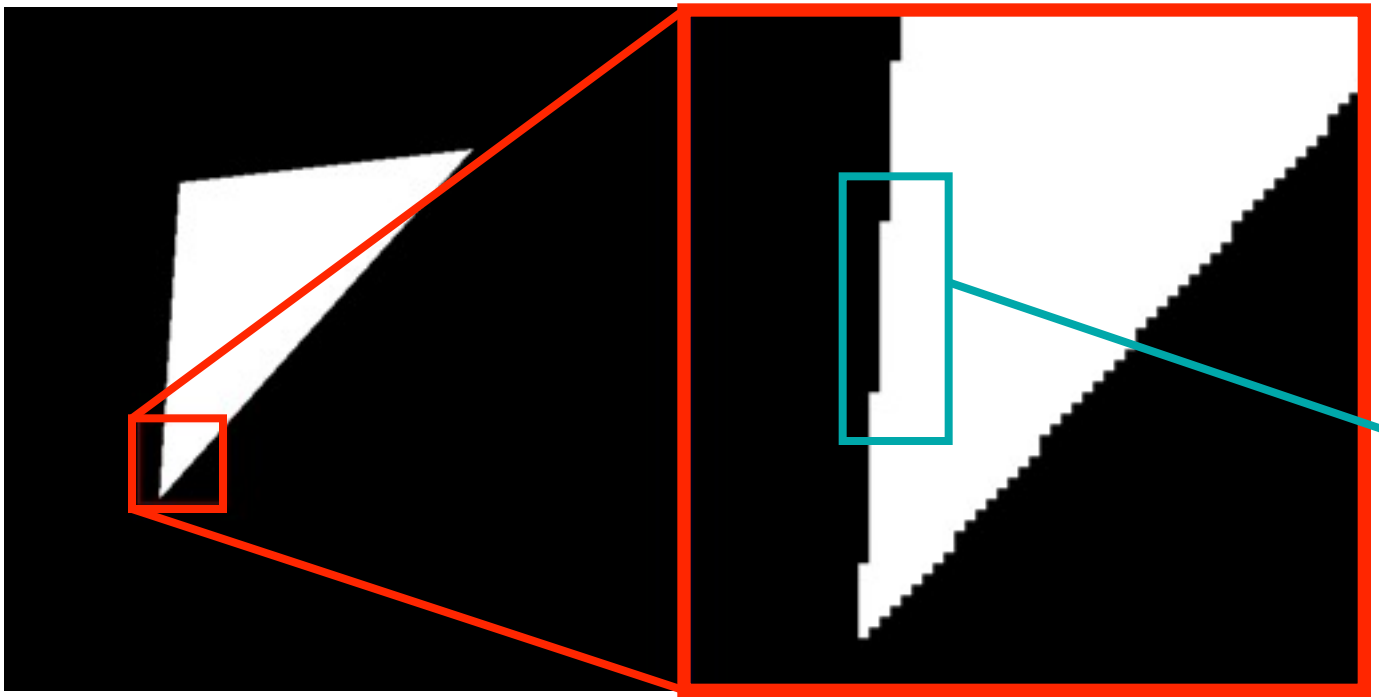
Michael Doggett

Department of Computer Science

Lund University

Demo of aliasing

- Aliasing (eng) = vikning (swe)



Called the ”jaggies”

DEMO

Shows that aliasing is *very* noticable when animated: called ”crawlies” in that case...

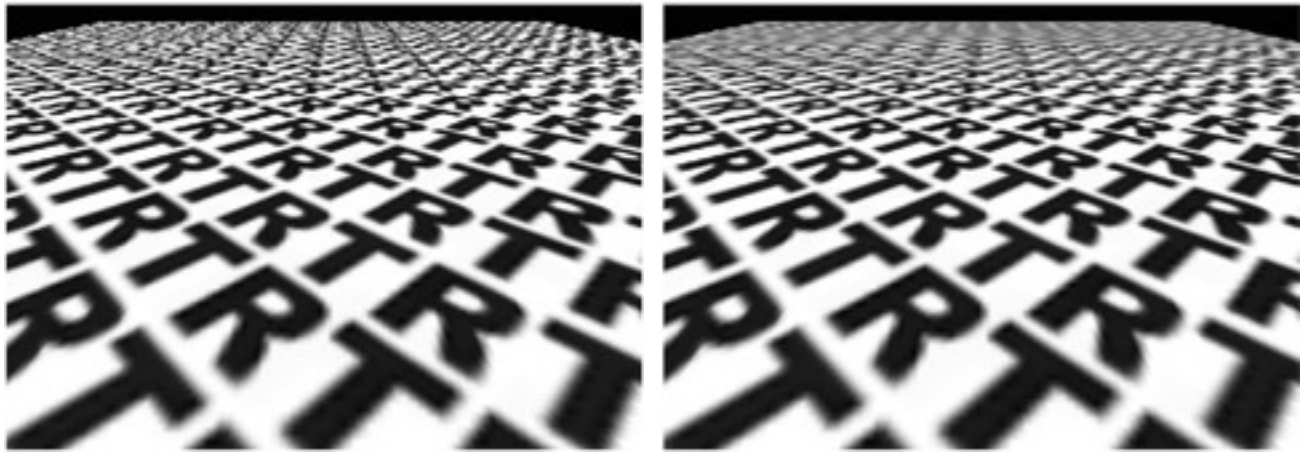
CG is a sampling and filtering process

- Pixels

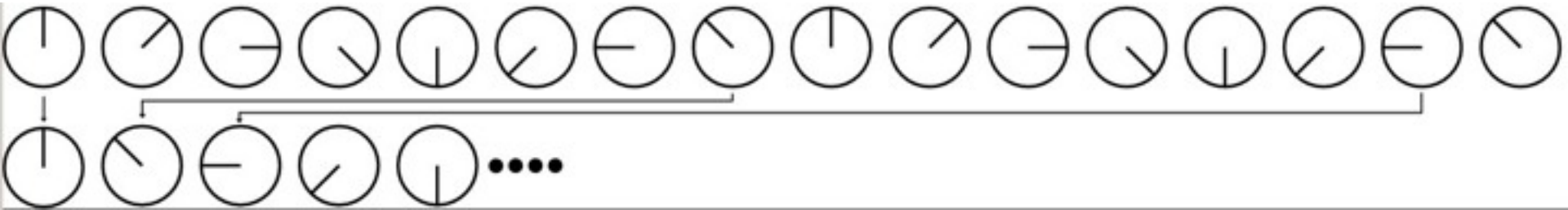
This is what we will study now

- Texture

A reasonable solution is mipmapping

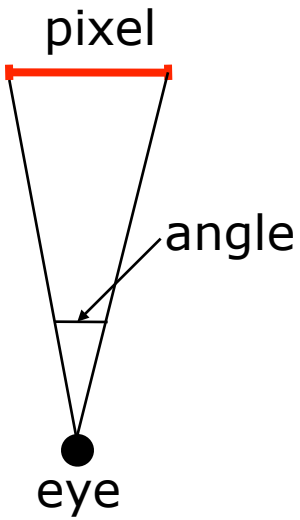


- Time



Mobile case

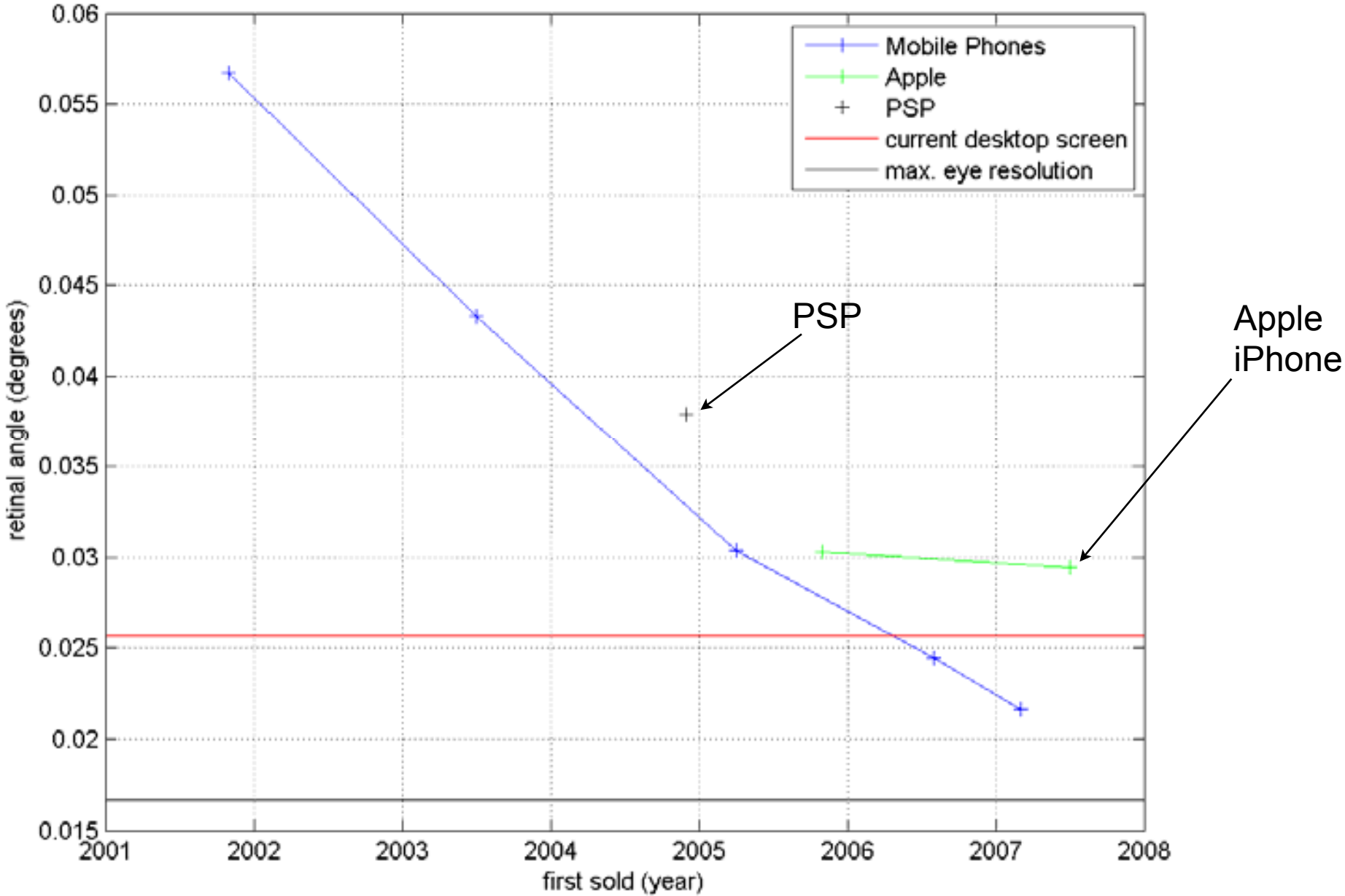
- **Small display, but held very close to your eyes**



- Our measurements in 2003:
  - Average eye-to-pixel angle is 1—4 times larger for mobile than for a laptop/desktop

- "These display conditions implies that every pixel on a mobile phone should ultimately be rendered with higher quality than on a PC system."
  - from "Graphics for the Masses: A Hardware Rasterization Architecture for Mobile Phones", SIGGRAPH 2003, Akenine-Möller and Ström

Today: dramatic changes since 2003



So, we do not need to care about antialiasing? Or what?

- Even though display technology has changed a lot, **aliasing** is still
  - visible,
  - disturbing, and
  - even on a PC, it is visible and disturbing.
- Example:
  - Playing a game on the PSP. First thing you notice is jaggies and crawlies...

What should we do?

- Well, if you have ”aliasing”, all you got to do, is to ”turn on ***anti***aliasing”...
- We want the ”average color” seen in the gray pyramid below [integral]

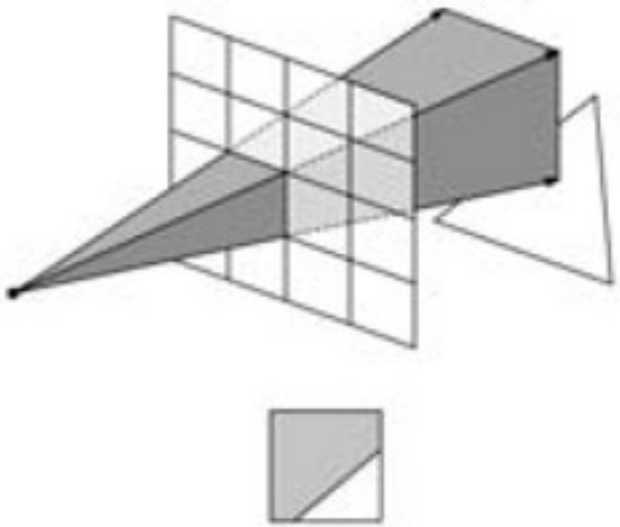
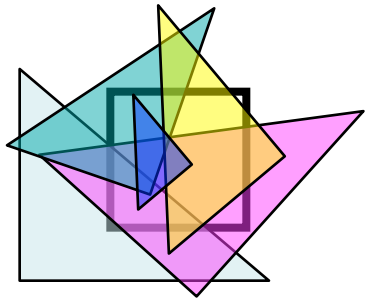


Image courtesy of Bill Mark, University of Austin, Texas

That's simple: solve it analytically...

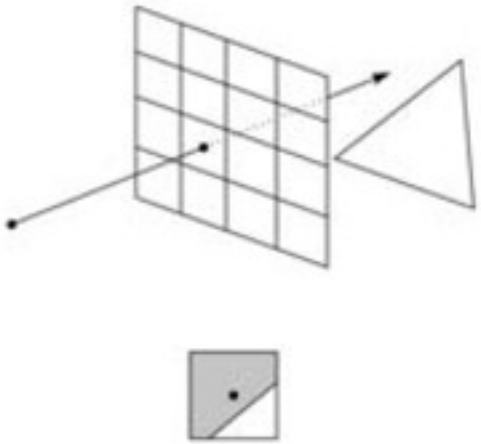
- In this case, it is simple: 
- How about this case:



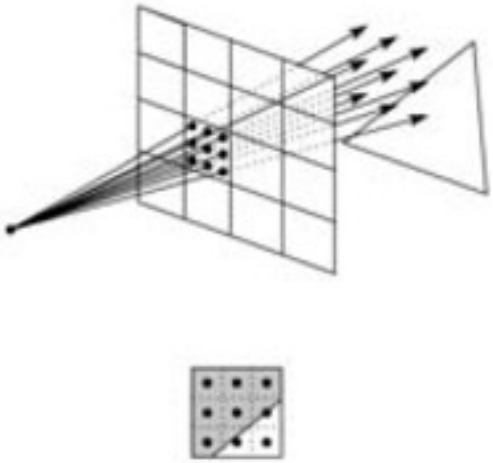
- It is actually very very hard!
  - Especially if you want to use limited computing resources...

Standard solution in graphics:

- Turn to ”point sampling”
  - i.e., evaluate ”color” in certain points



Point sampling using a single point is what we’ve done so far... Using a sample at the center of each pixel

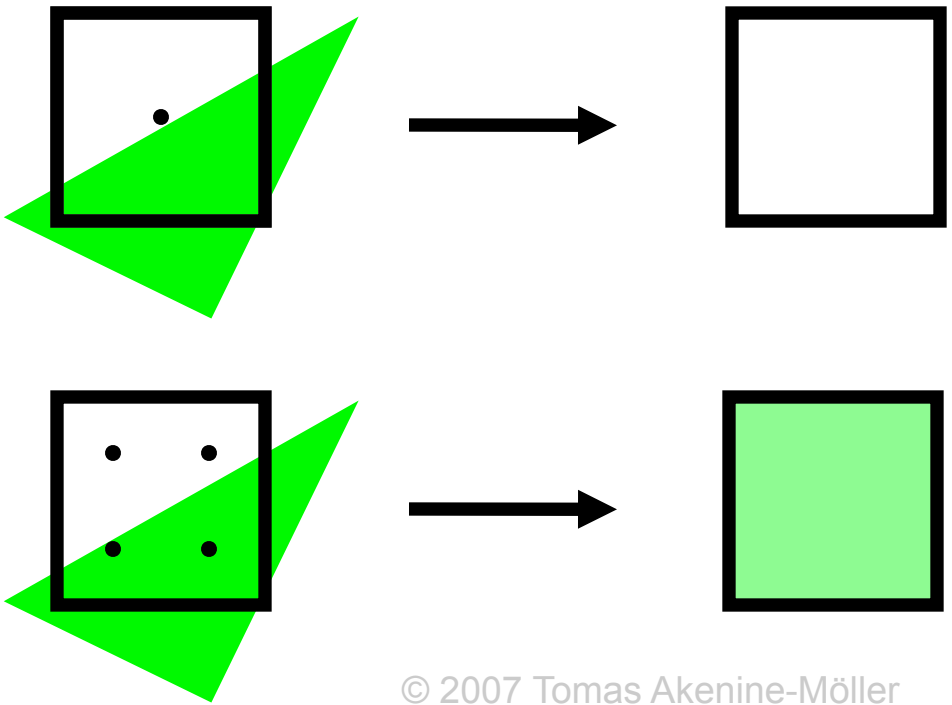


Using more samples per Pixel can give a more accurate estimate of the pixel’s color

Images courtesy of Bill Mark, University of Austin, Texas

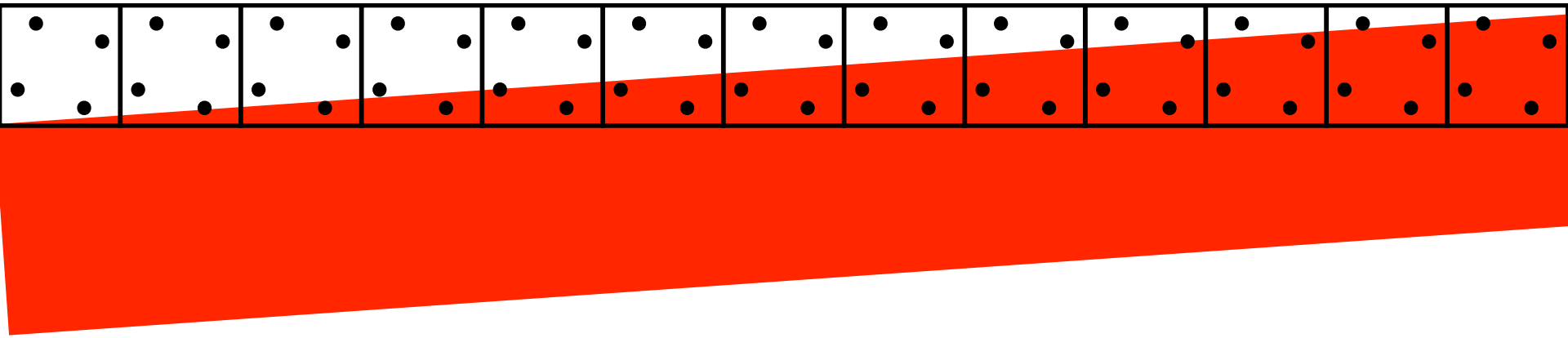
Screen-based Antialiasing

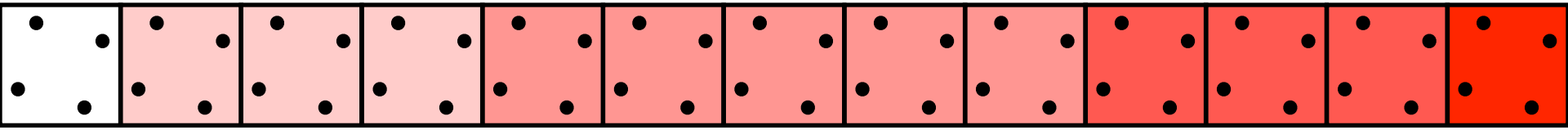
- One sample per pixel is not enough
- Hard case: an edge has infinite frequency content
  - Means no sample rate can fix this for us...
- Supersampling techniques: use more samples



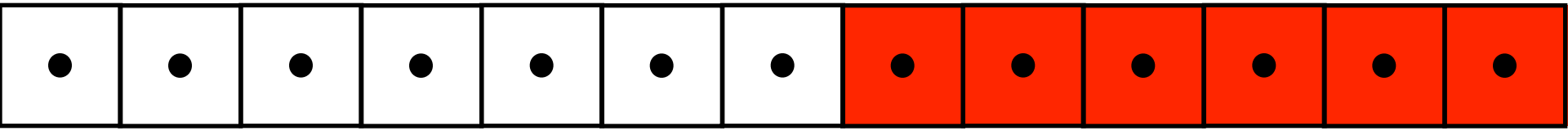
NOTE: frame buffer needs to be 4x as big!

Another example

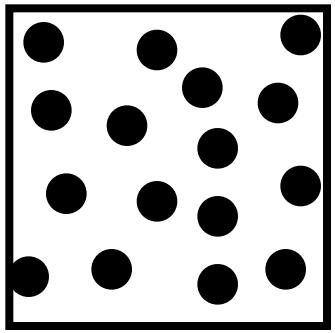




Instead of:



How to compute the color of a pixel from samples?

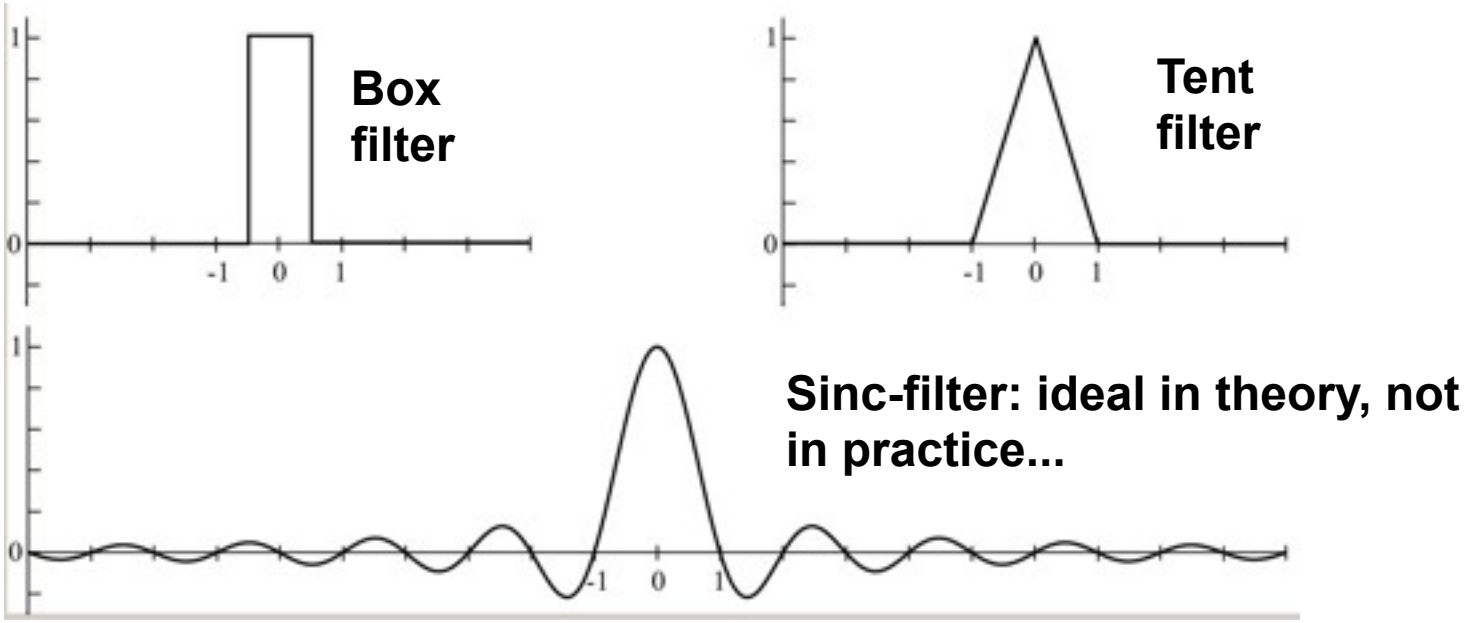


$$\mathbf{p}(x,y) = \sum_{i=1}^{n} w_i \mathbf{c}(i,x,y)$$

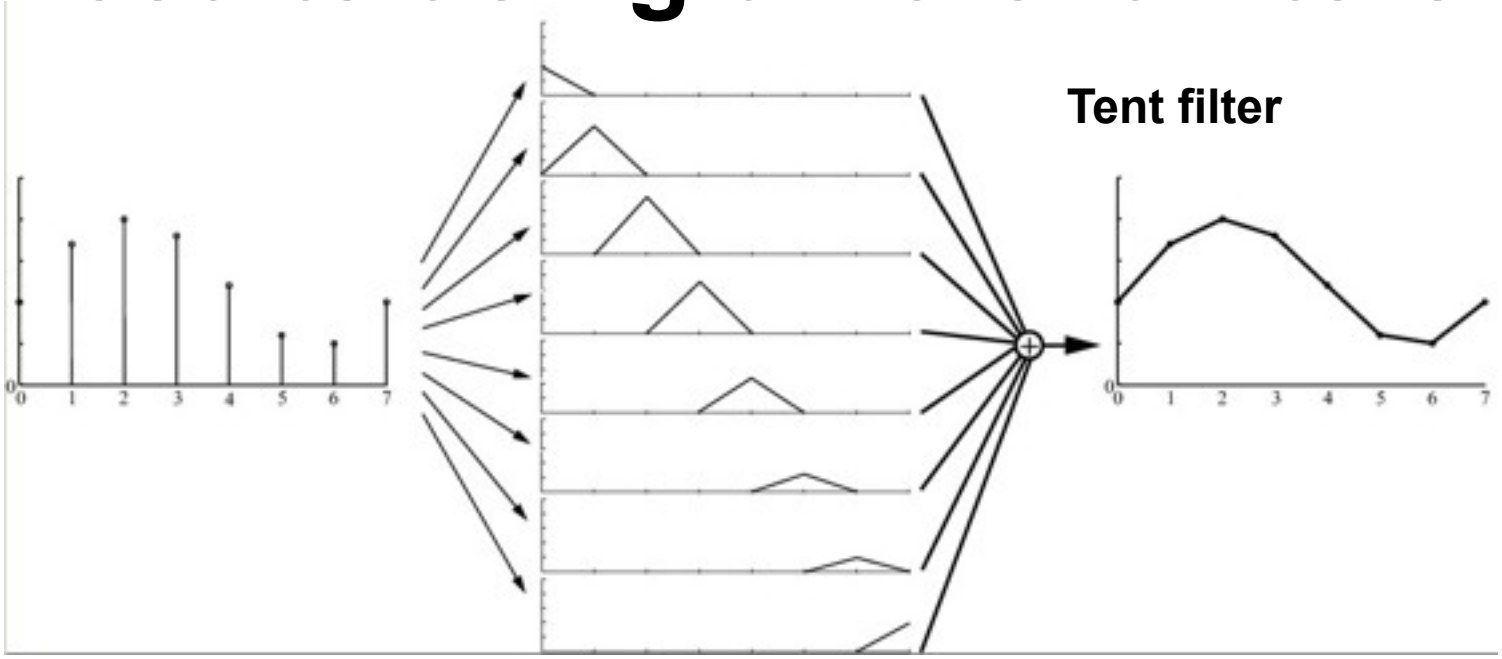
- w_i are the weights in [0,1]
  - Depends on the filter you use!
- $\mathbf{c}(i,x,y)$ is the color of sample i inside pixel
- $\mathbf{p}(x,y)$ is the color of the pixel

Sampling theory

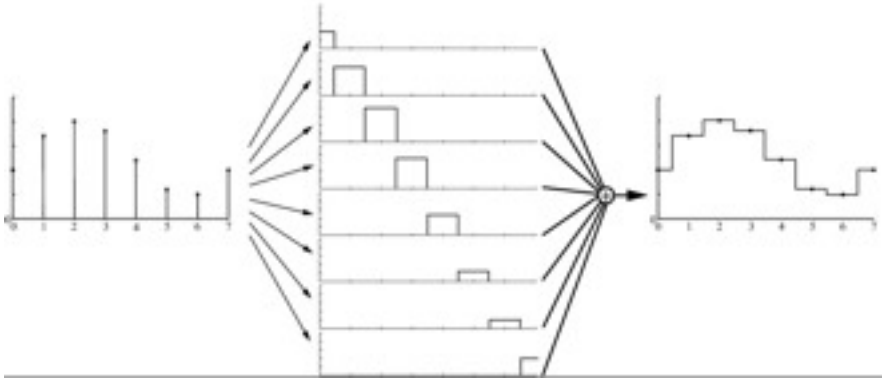
- The filter weights, w_i
  - The figure below assume a pixel starts at -0.5 and ends at +0.5



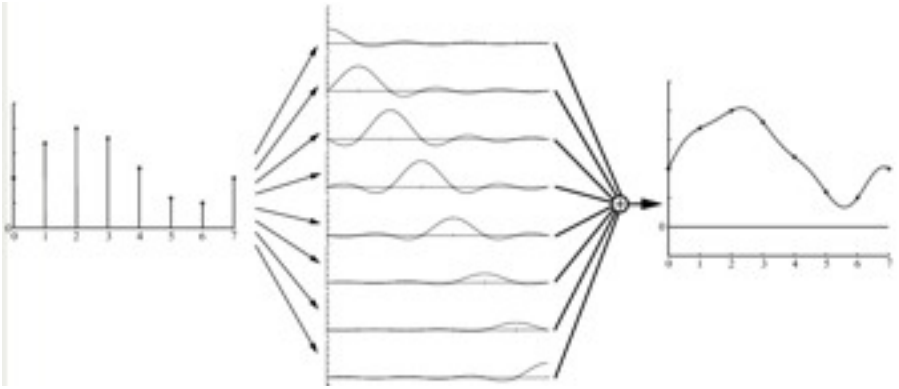
Results using different filters



Box filter



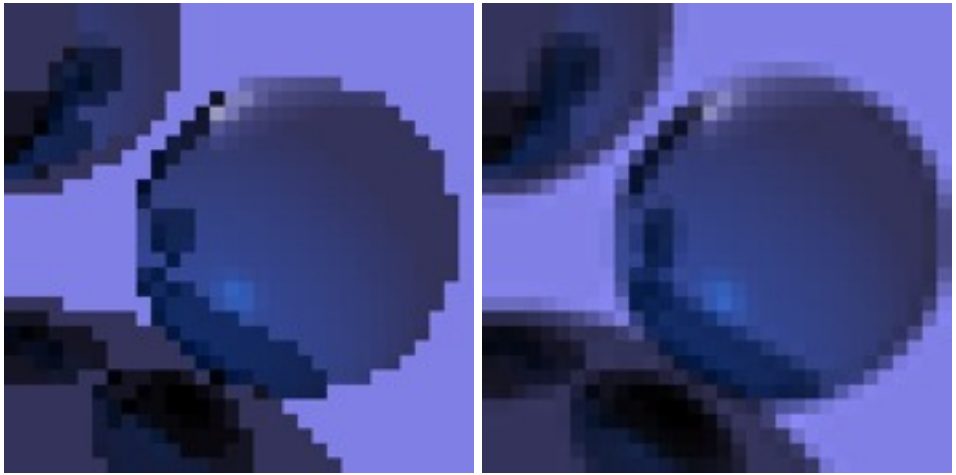
Sinc filter



Mobile supersampling

- Increase the sampling rate, and hope for the best
  - There are good ways and bad ways...
- What can we afford in the mobile case?
  - As little as possible...
  - Still want good quality!

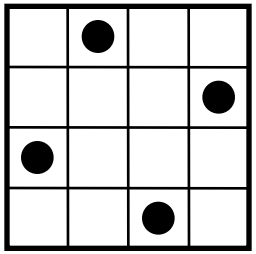
1 sample per pixel

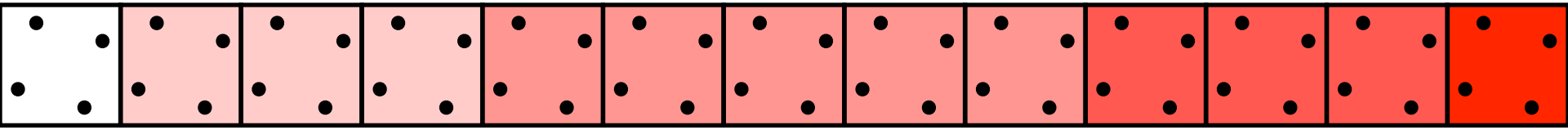


4x4 samples per pixel

Standard supersampling schemes

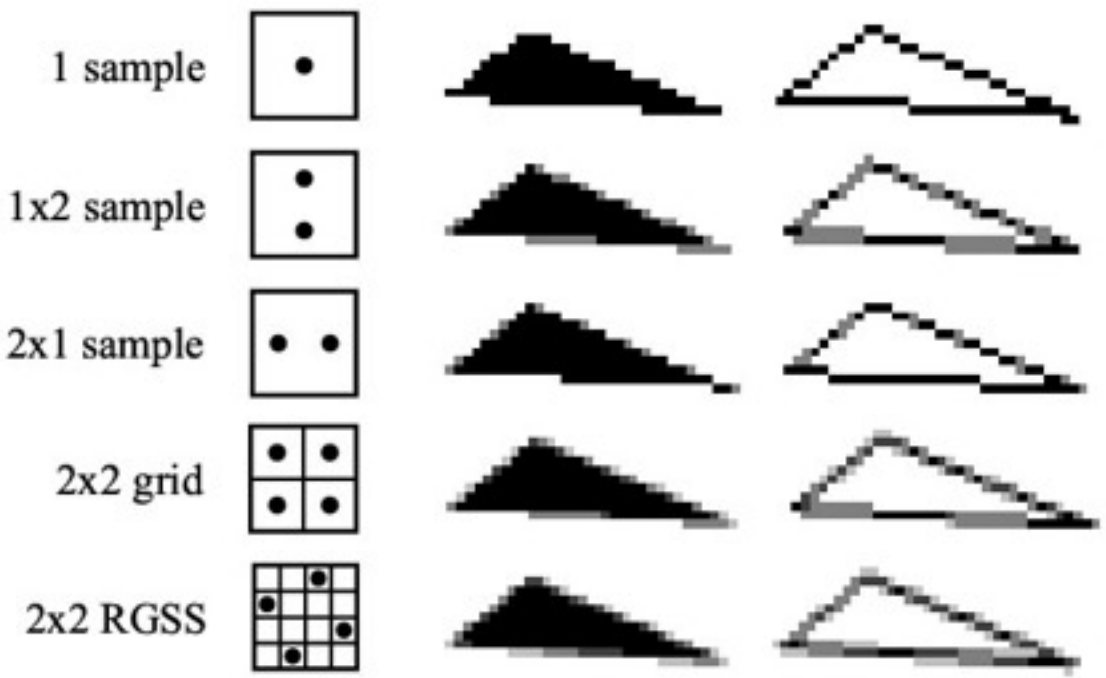
- Rotated Grid Supersampling (RGSS)
  - Good for near horizontal and vertical edges
  - Cost: 4 samples/pixel





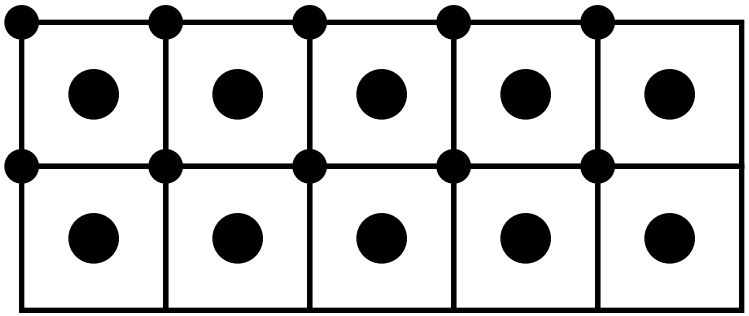
- A study by a researcher called Naiman:
  - Near-horizontal and near-vertical edges are the most annoying to humans
    - Then comes near-45-degree edges...
  - RGSS: is very good for those cases!

Standard supersampling schemes



- Bad: 1, 1x2, 2x1, 2x2...
- Good: RGSS
  - Quite expensive though!

NVIDIA's Quincunx sampling



2 generated samples per pixel
5 samples used to
compute color of pixel

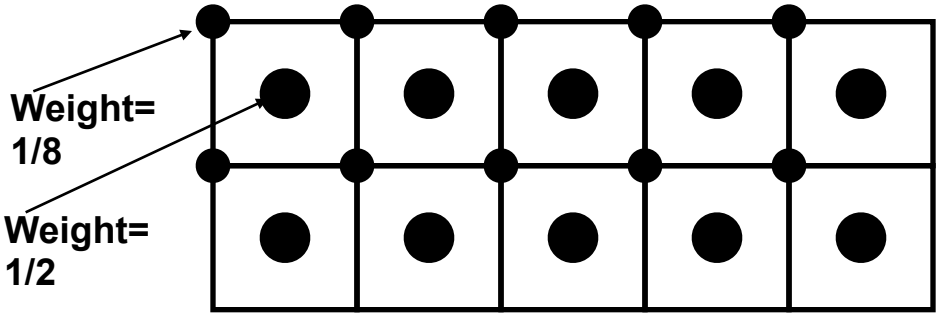
- Good news: cost is only 2 samples/pixels
  - The rest comes from sharing samples with neighboring pixels

- Still, RGSS quality is much better!
  - Check bottom edge of black triangle

DEMO

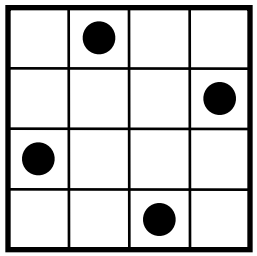
A new inexpensive multisampling scheme

- Combine good features of two existing schemes
- Quincunx scheme by NVIDIA:

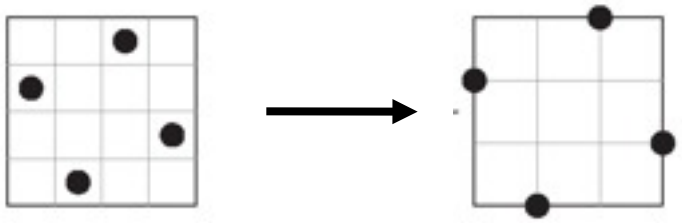


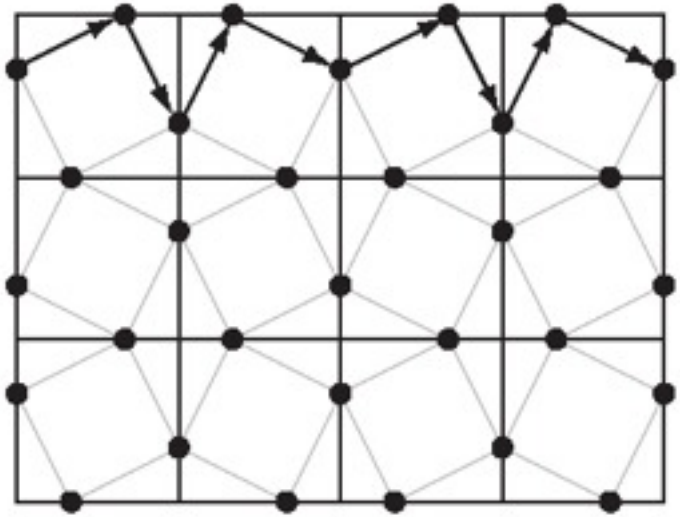
2 generated samples per pixel
5 samples used to
compute color of pixel

- [Weights sum to one!]
- Rotated Grid Supersampling (RGSS)
  - Good for near horizontal and vertical edges



FLIPQUAD supersampling

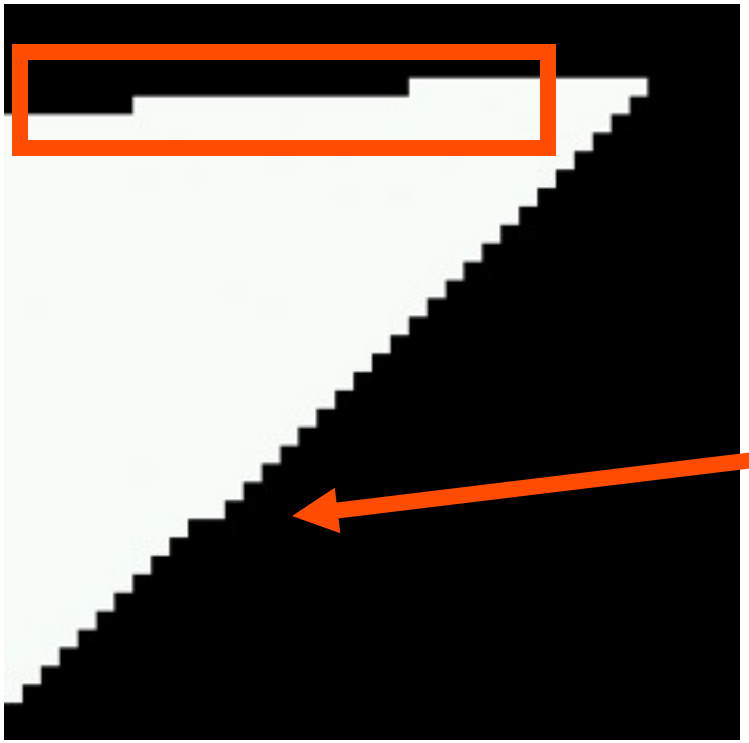




Quality: quite near RGSS (costing 4 samples)

Visual Results: FLIPQUAD

Concentrate on two things:



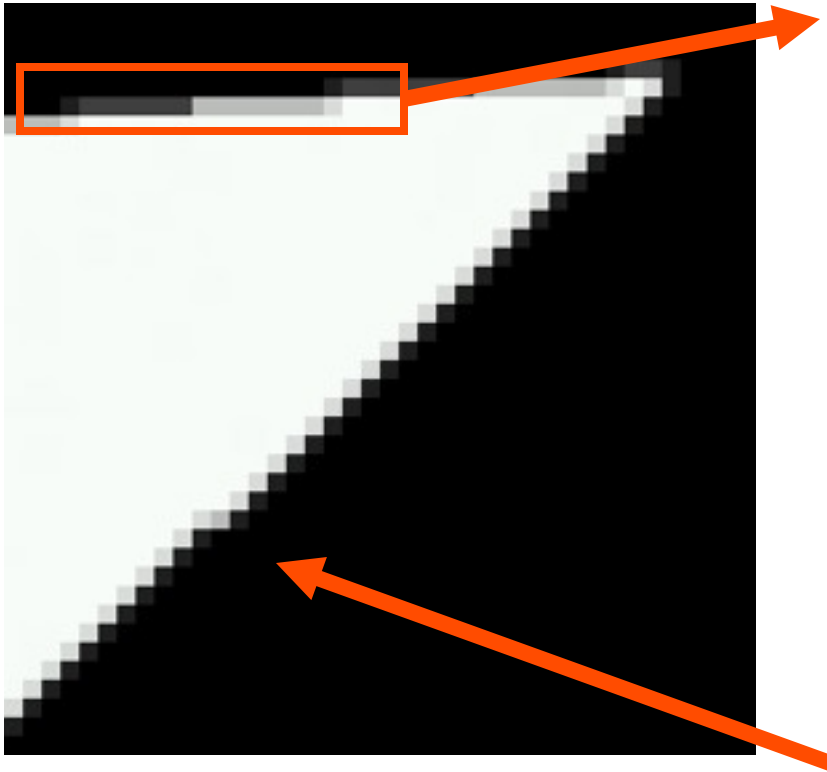
Single Sample

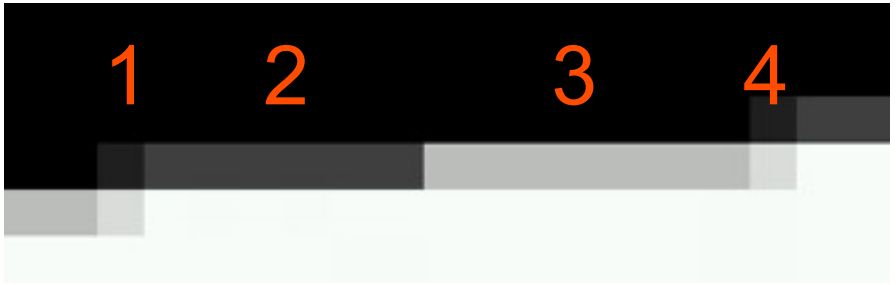
Aliasing near horizontal edge

"jerk" when angle is near 45°

Visual Results: FLIPQUAD

NVIDIA Quincunx



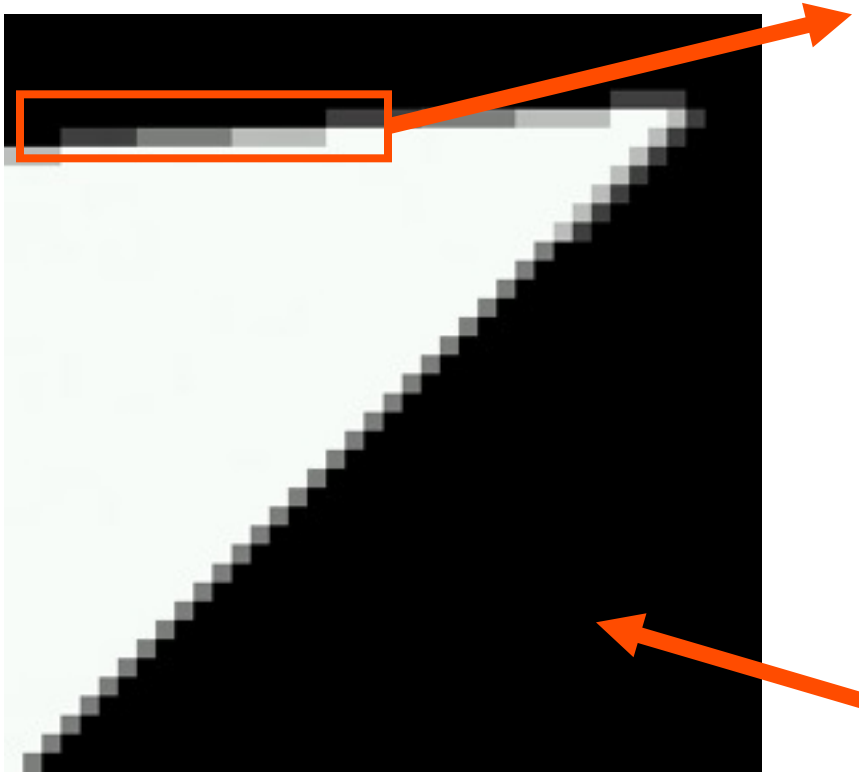


Four mid gray levels, but only two effective for near horizontal edges

Note "jerk" at 45° angle still visible

Visual Results: FLIPQUAD

FLIPQUAD (proposed scheme):

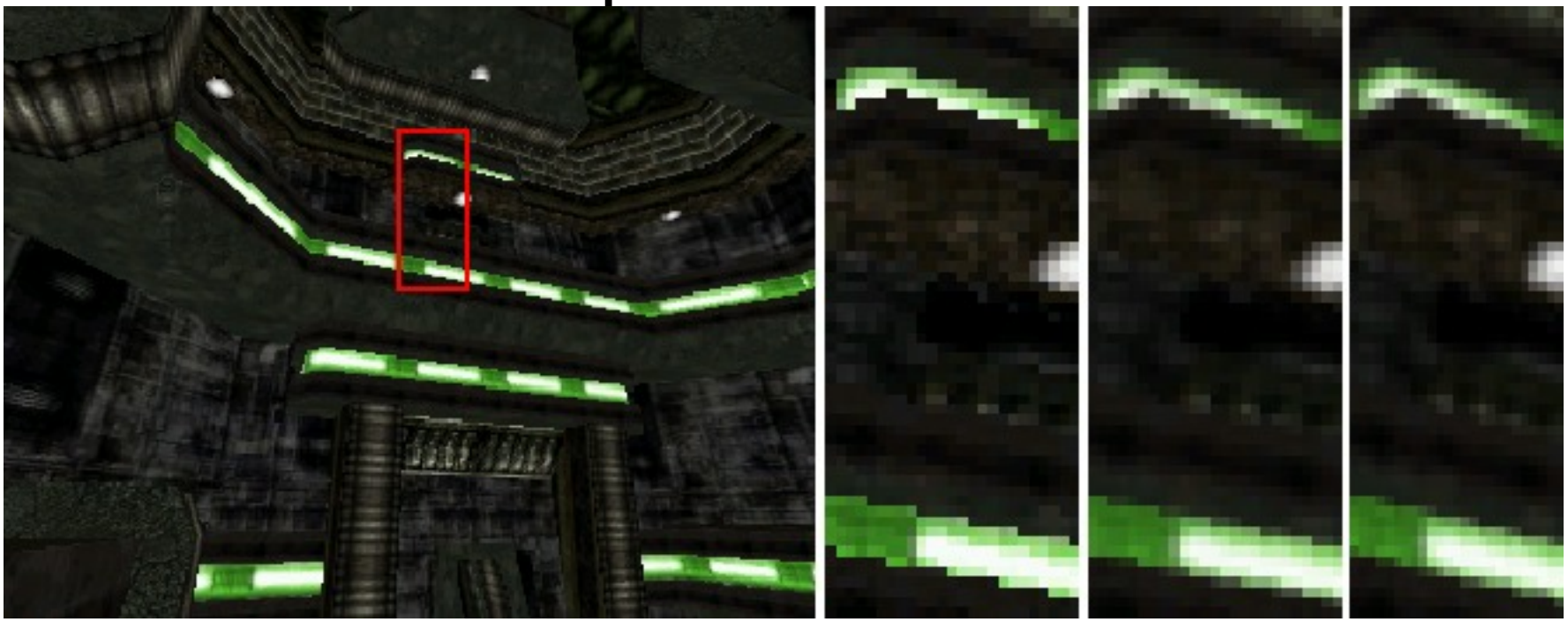


Three mid gray levels, but effective since evenly spaced out

No “jerk”

Visual results: FLIPQUAD

Full scene example:



Single Sample | Quincunx | FLIPQUAD

Finnish research group has shown that *FLIPQUAD* is the best sampling scheme at 2 samples/pixel

DEMO

FLIPQUAD implemented in ATI/Bitboy's architectures

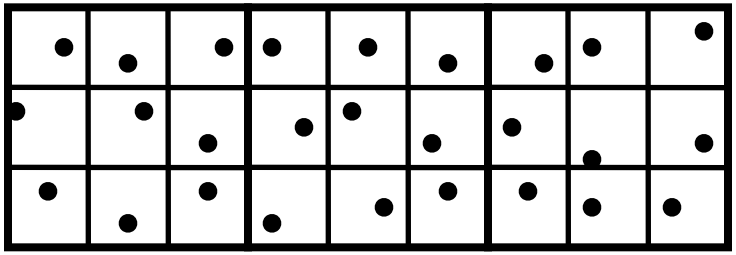
Supersampling vs multisampling

- There are actually two types:
  - **Supersampling**: any type of sampling that uses more than one sample per pixel
  - **Multisampling**: is supersampling, but with the restriction that the fragment shader (texturing etc) is only evaluated once per pixel.
    - Cheaper but lower quality
    - Only affects edges, not shader antialiasing..
    - Hence, supersampling can provide better texture sampling

High-quality antialiasing

- Use jittered sample points
  - Replaces aliasing with noise
    - Humans easily accept a bit of noise, rather than aliasing, which is disturbing

Divide pixel into nxn subpixels, random position inside subpixel



High-quality antialiasing (2)

- Sample positions may differ spatially but not temporally
  - Each pixel must use same sample locations every frame

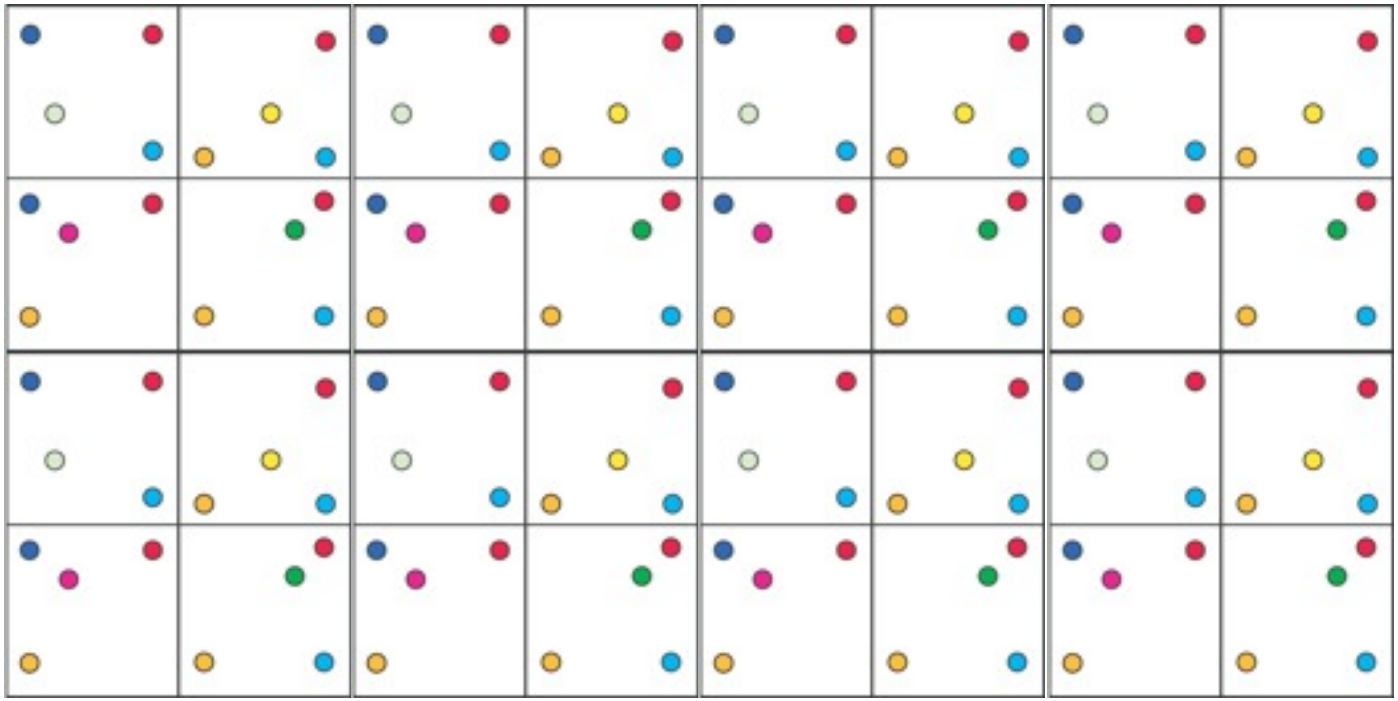
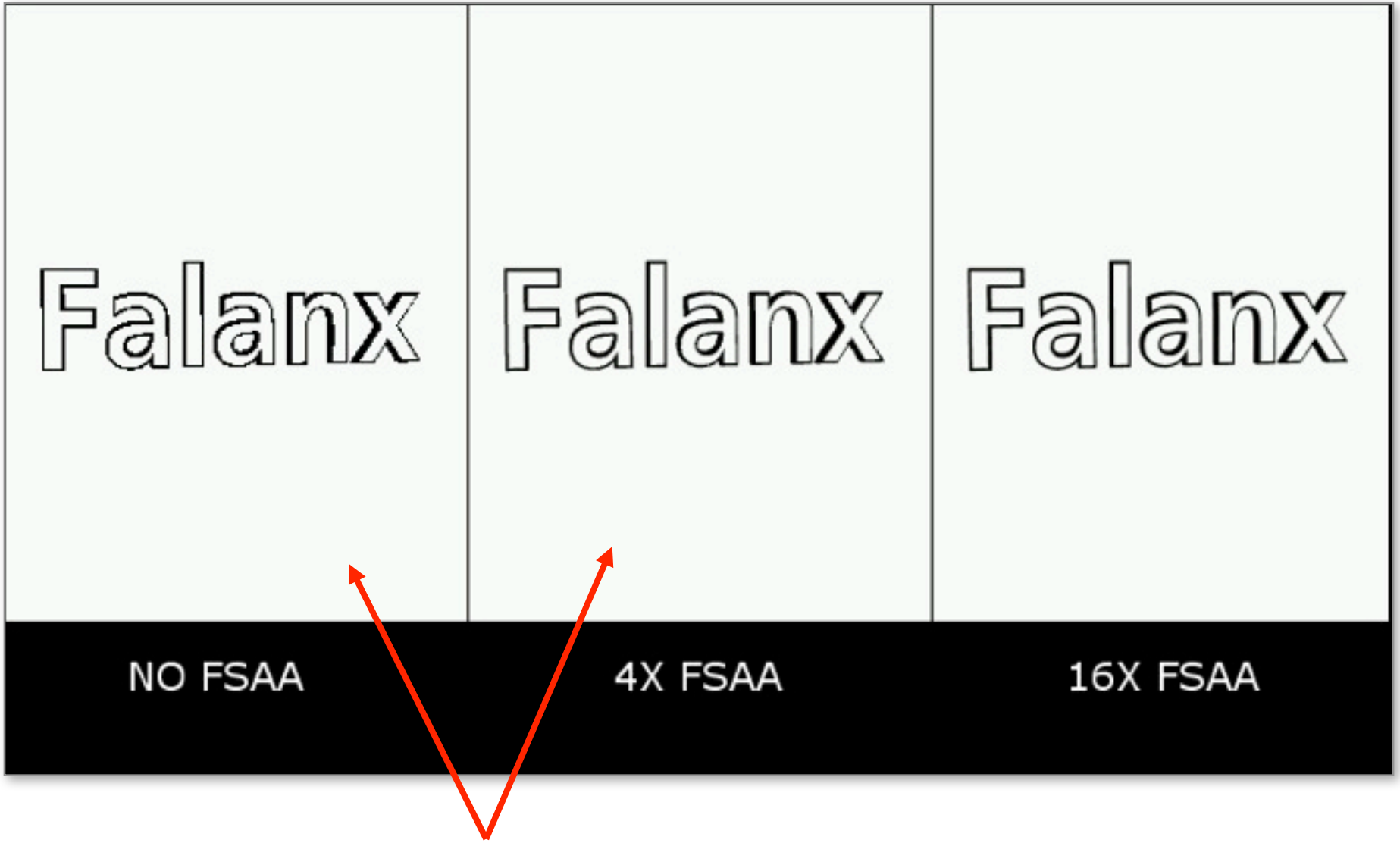


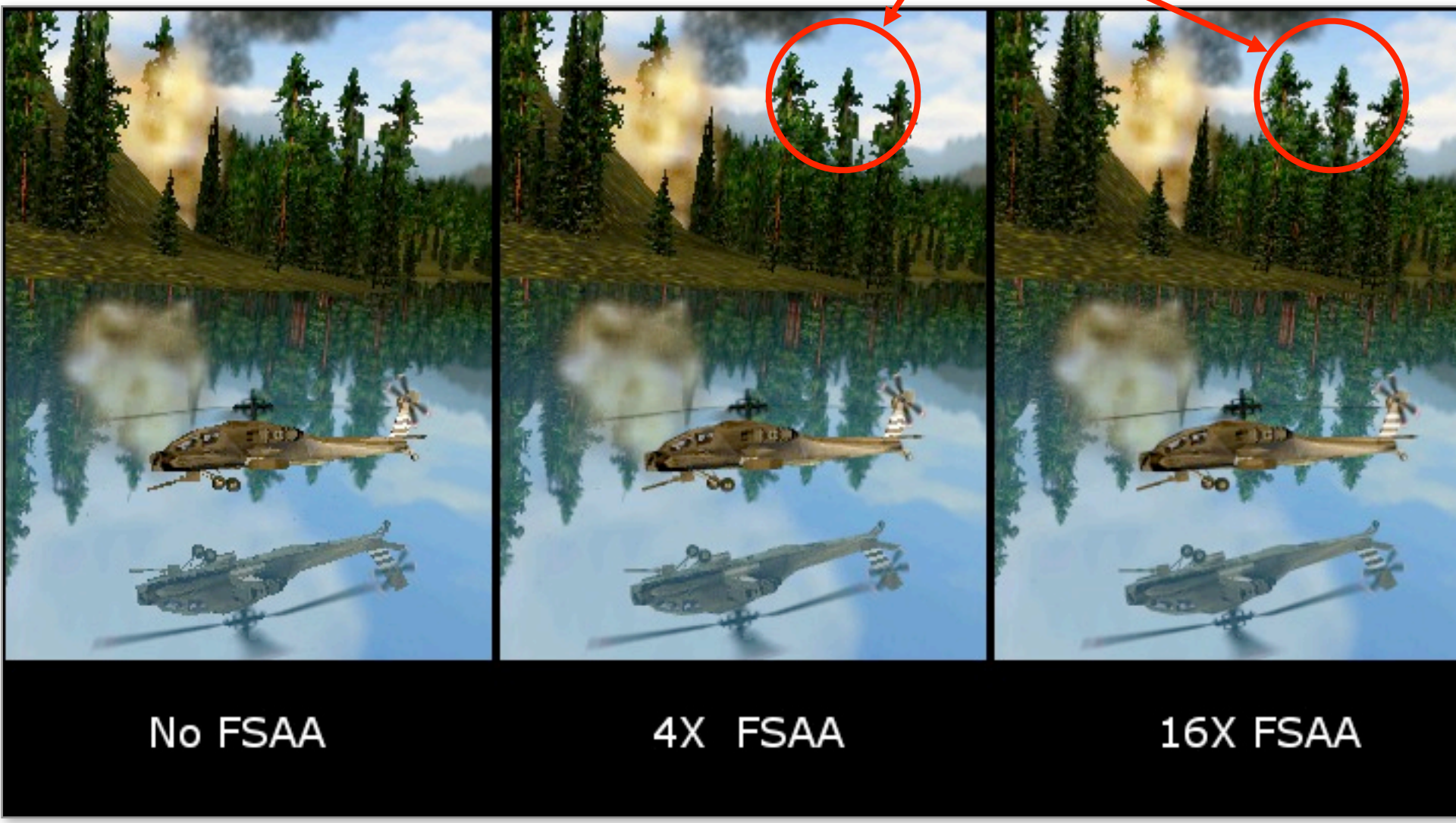
Image from
ATI's SMOOTHVISION sampling
[Note: this is not exactly jittered sampling, rather random/interleaved sampling]
Though, positions of samples are programmable

Pattern is pseudo-random, and repeatable → manageable for HW



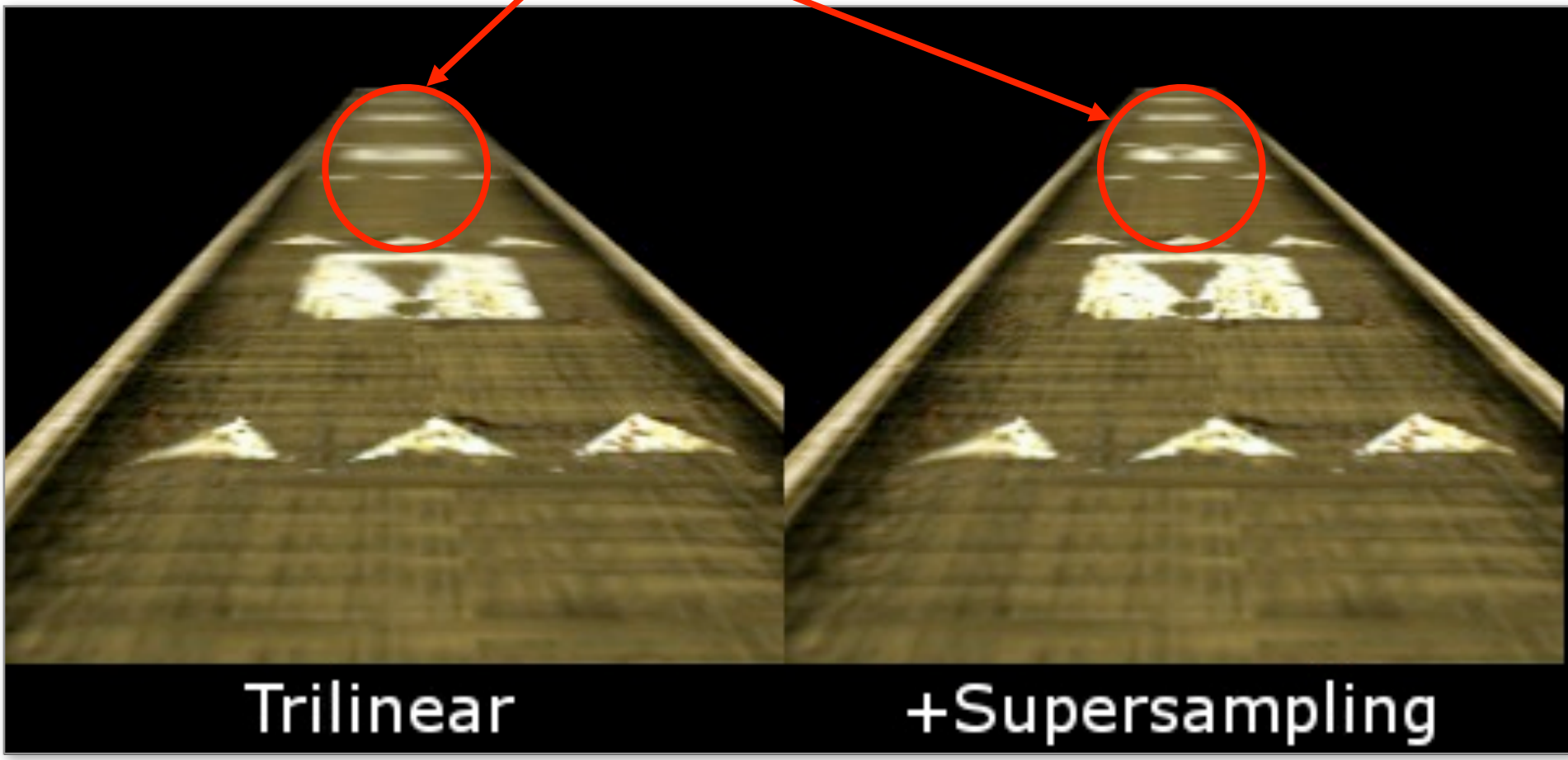
ARM/Falanx architecture designed so that these two modes run at approximately the same speed

Check out the power of antialiasing of transparency



Images courtesy of Falanx

Texture sampling becomes better with supersampling techniques



Images courtesy of Falanx

Next time...

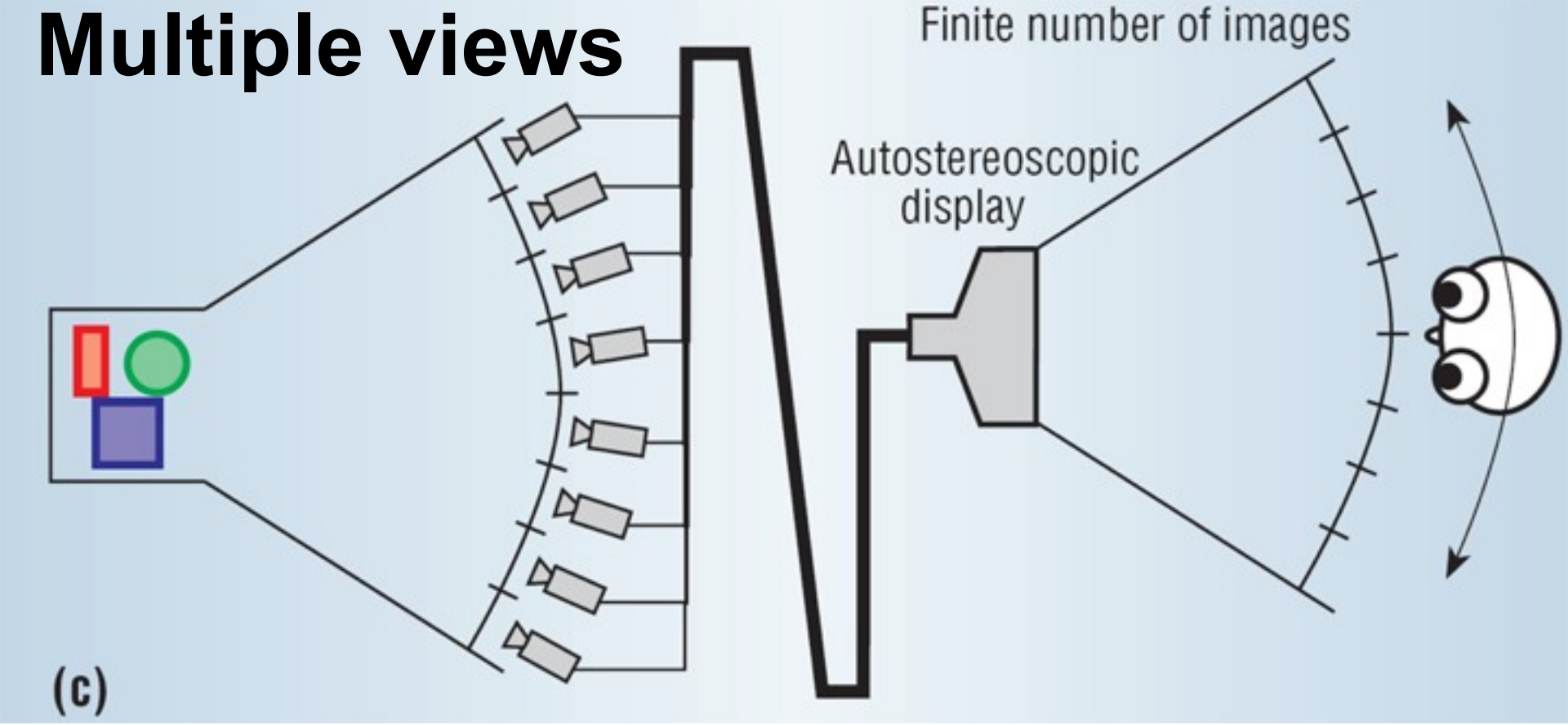
- Is the last lecture
  - Jury will be here to judge the ones competing
    - Is anyone part of the competitions?
      - iPhone Project?
      - Graphics Hardware optimization?
    - You need to notify me whether you want to be part of the competition
- Project deadline is on Friday this week
  - You shall deliver: report + source code
  - Send by email to me and Magnus!
- Check out the list of required readings (papers etc) on website.

Intro to another topic: Real 3D Graphics...

- LG predicts 3D TV market > 30 million units by 2012
- Korea starts 3D Full HD broadcast next year
- Need a new type of displays for *real* 3D
- Known as autostereoscopic displays
- No need for extra glasses or other peripherals

Stereo graphics used to be painful

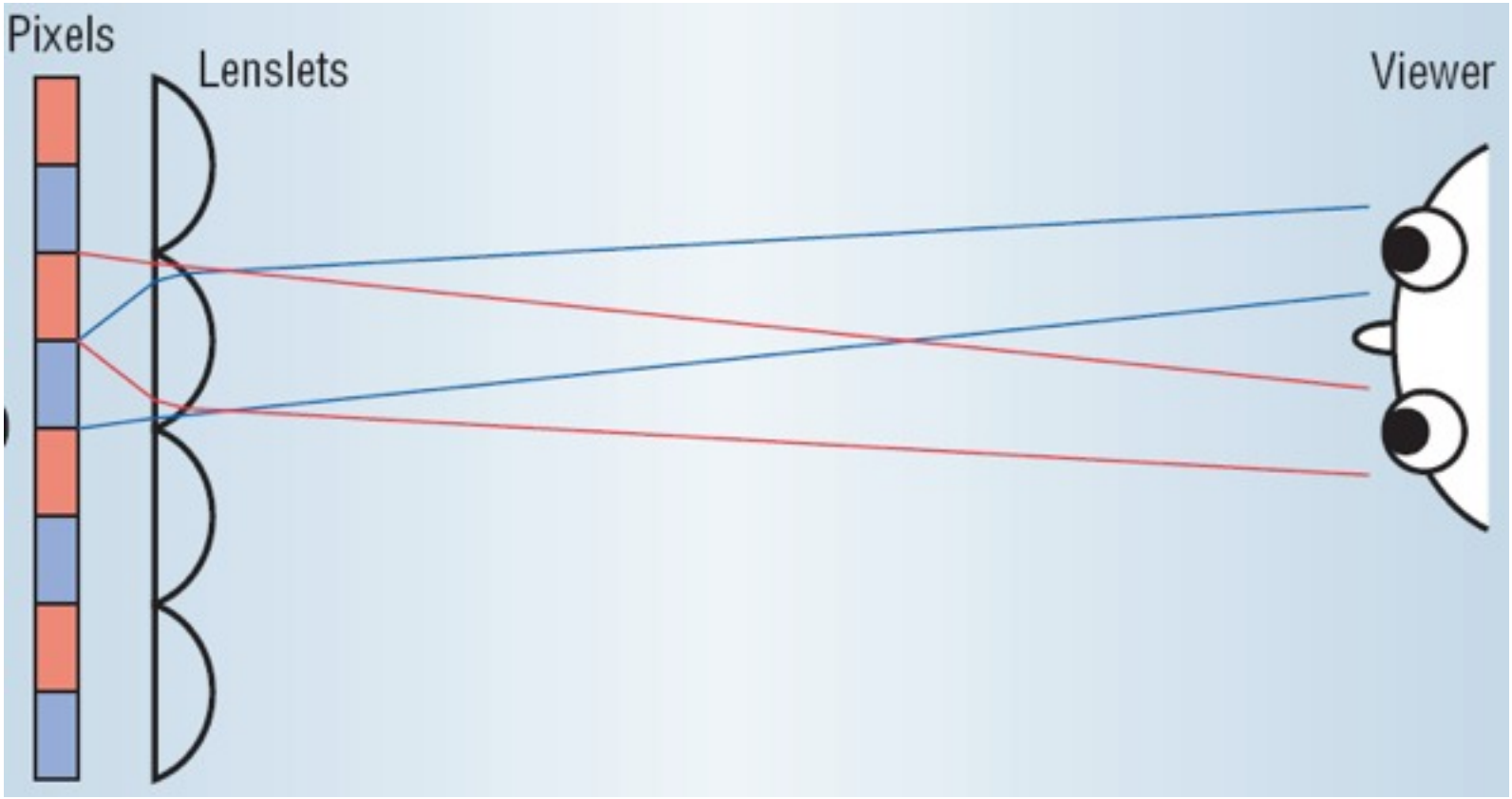
Multiple views



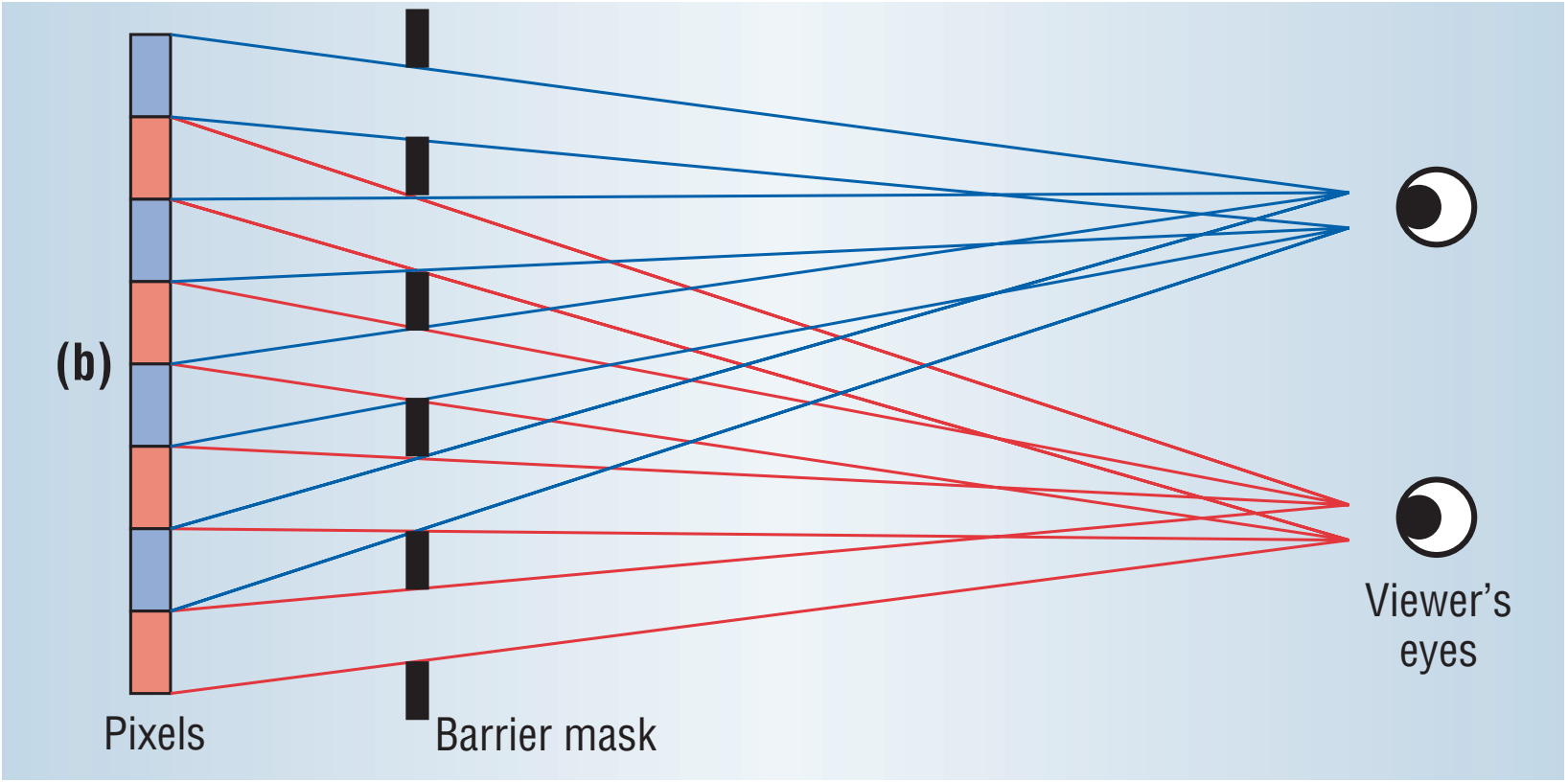
- Philips manufactures one display with 9 views
- Displays for mobile phones are available
- Systems for 3D TV and video have been built
  - 16 views
  - 80 views

Display type 1: lenticular displays

Stereo principle

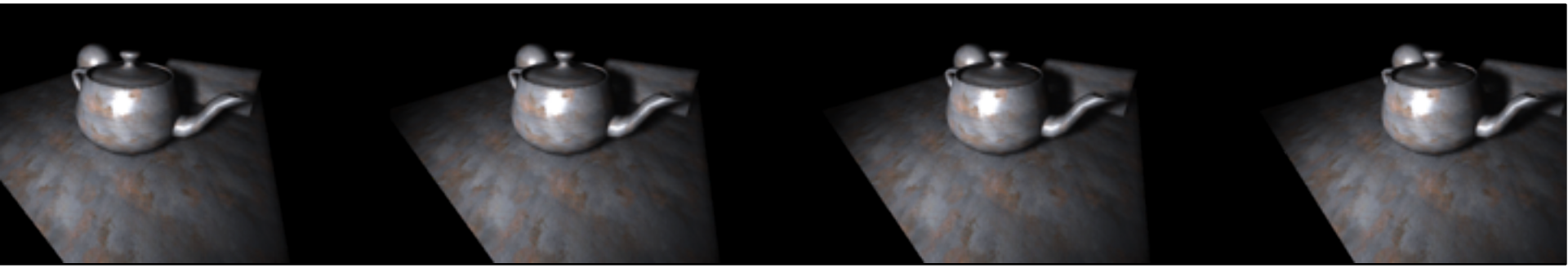


Display type 2: parallax barriers

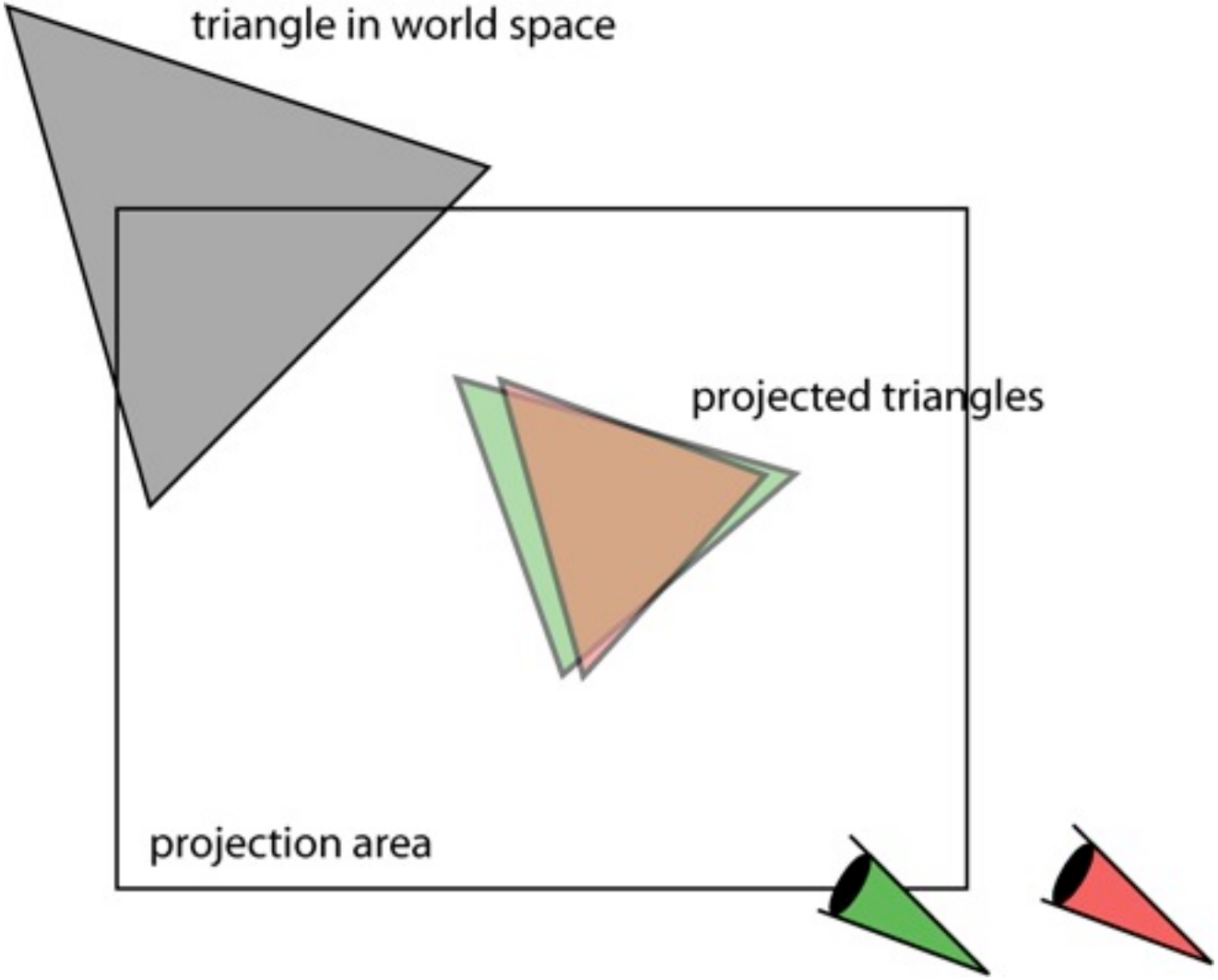


How much does it cost?

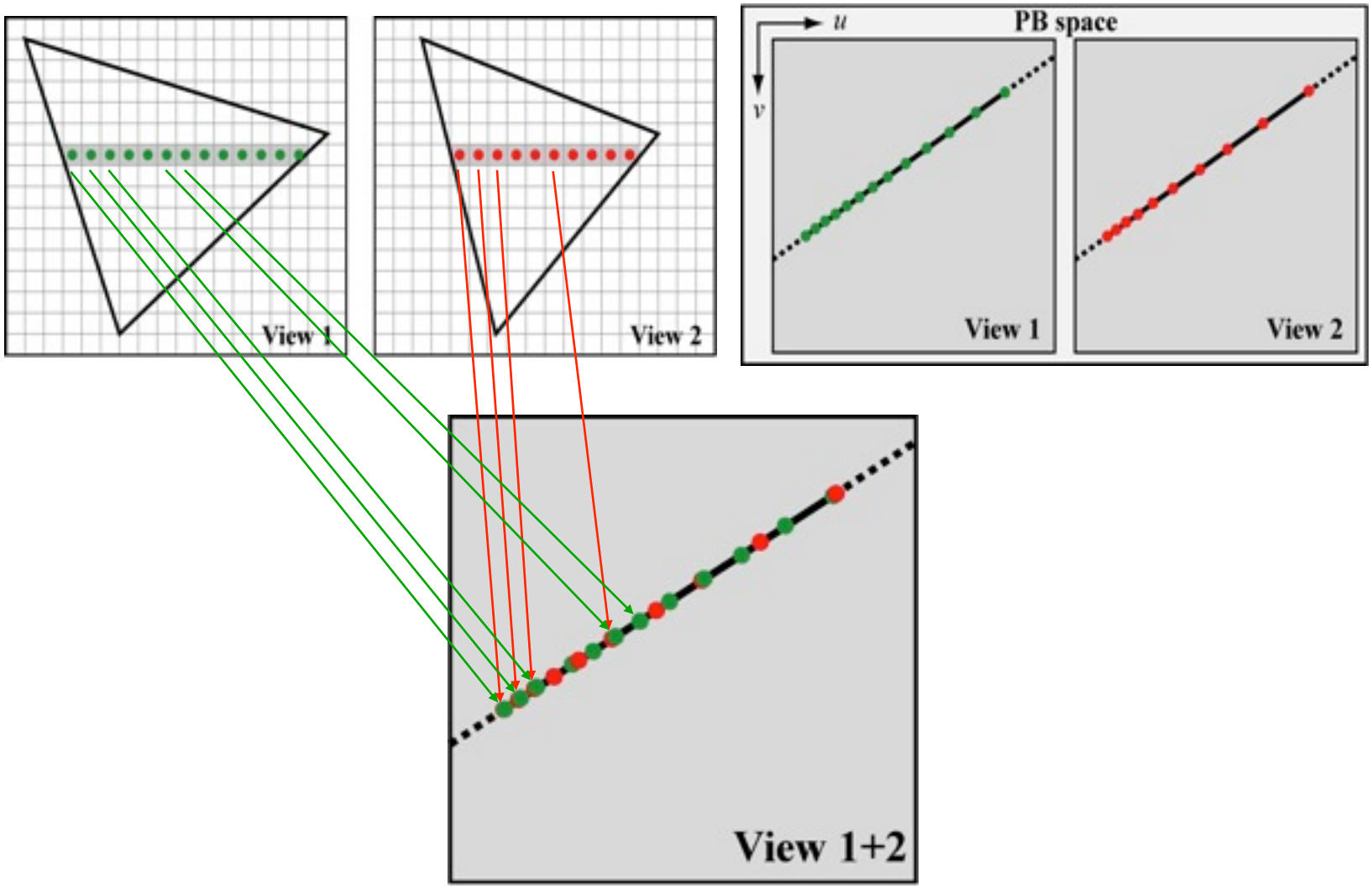
- Brute-force
  - With N views, it costs N times as much as a single image
  - Our display has 9 views...
- Can we do something smarter?
  - As shown below, the images are quite similar
  - Let's exploit this...



Hasselgren and Akenine-Möller sorted traversal



Sorted traversal



Results

Green pixels are approximated from centre view

Our algorithms

Future...

- These displays are likely to be a big hit
- Need to mature a bit more

- Lots of new research to be done in this field

The end