

Anti-aliasing and Graphics Formats

Eric C. McCreath

School of Computer Science
The Australian National University
ACT 0200 Australia

ericm@cs.anu.edu.au

- Nyquist sampling frequency
- supersampling
- area sampling
- filtering
- Raw image formats
- PNG
- JPEG

- Nyquist–Shannon sampling theorem states:

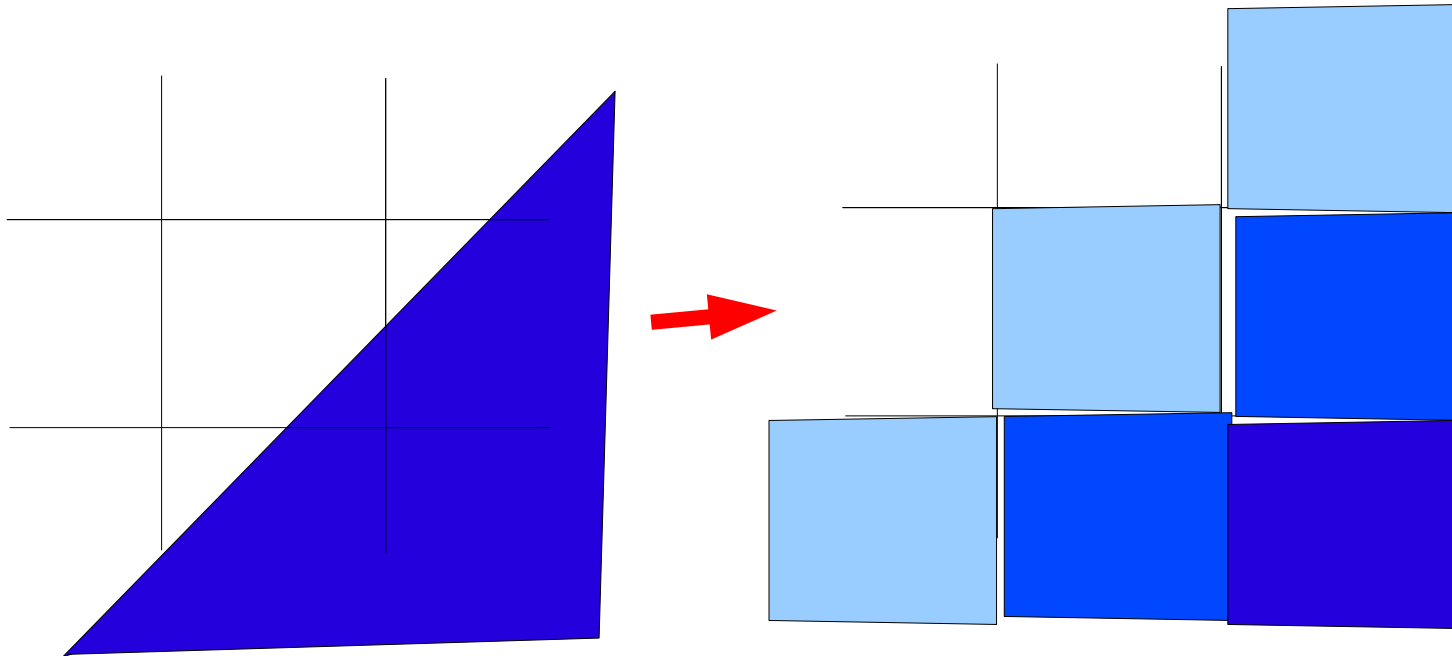
If a function $x(t)$ contains no frequencies higher than B hertz, it is completely determined by giving its ordinates at a series of points spaced $1/(2B)$ seconds apart.

C. E. Shannon, "Communication in the presence of noise", Proc. Institute of Radio Engineers, 37:1, pages 10–21, 1949.

- *Patterns which have 'high' frequency along with sharp changes in colour(object boundaries) cause the most problems in Computer Graphics.*
- *The problem cause by under sampling is called aliasing.*

- Super sampling is an anti-aliasing approach that calculates intensities at sub-pixel grid positions and uses the average of these pixels to determine the pixel intensity.
- Super sampling is very costly both in terms of memory and processing requirements. One approach for reducing this cost is to use adaptive supersampling. Adaptive supersampling only supersamples pixels that are on boundaries.
- There are a number of variations on the sub-pixel position for obtaining samples, these include:
 - Grid
 - Random
 - Poisson disk
 - Jitter
 - Rotated Grid

- Rather than supersampling it is possible to calculate pixel intensity by working out the overlapping area within pixels of parts of the scene. The proportion of area within a pixel is used to calculate a weighted average over the contributing colours.



- Aliasing problems can be addressed by applying some form of filter to the image. A common filter to use is a Gaussian blur.
- Gaussian blur is achieved by convolving a 2D Gaussian function with the image. This has the effect of replacing an image intensity with that of the average intensity of surrounding pixels.

$$g(x, y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}}$$

Gaussian function

$$f_{new}(x, y) = f_{old}(x, y) * g(x, y)$$
$$= \iint f(u, v) g(x-u, y-v) du dv$$

Convolution operator

- Raw Image Formats are lossless formats that store data that closely maps the sensors/pixels of the camera/display device. These will often be in either RGB or YUV colour spaces.
- The meta data will include information like: resolution, byte/bit ordering, the number of bits per intensity, colour space used, palette (if one is used), etc
- Often raw images will be 2-6 times larger than compressed formats like jpeg.
- The below is an example of a 3x3 image with 8 bit RGB intensity values.

p0	p1	p2
p3	p4	p5
p6	p7	p8



▲
Data will be $3 \times 3 \times 3 = 27$ bytes long

- PNG is a lossless data compression format for storing bitmap images.
- Jpeg is a lossy data compression format also for storing bitmap images.

Storage size and quality of a 200x200 image

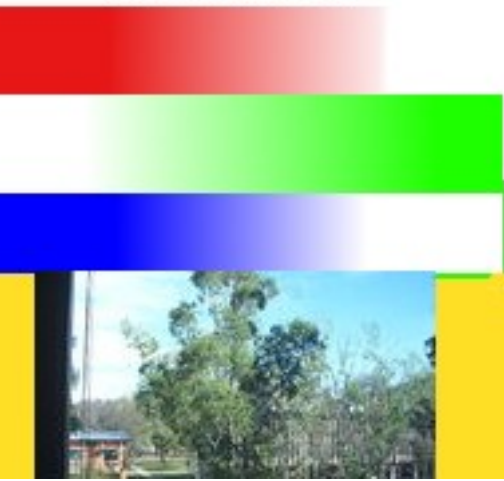
Raw(RGB 8bit/ch) 117k
PNG 18k

JPEG (q=2) 1k

JPEG (q=10) 2k

JPEG (q=90) 10k

Some Text



Some Text



Some Text



Some Text



- JPEG is great for photos with textures and smooth changing colour, however, it is not as good for text, icons, or line drawing with sharp changing colours.
- JPEG files are made up of a sequence of segments divided by markers these indicate the type of the next segment.

```
ericm@ericm-desktop:~/courses/cg/notes/formats$ od -c SimpleImage2.jpeg
0000000 377 330 377 340 \0 020 J F I F \0 001 001 001 \0 H
0000020 \0 H \0 \0 377 376 \0 023 C r e a t e d
0000040 w i t h   G I M P 377 333 \0 C \0 377 377
0000060 377 377 377 372 377 377 377 377 377 377 377 377 377 377
0000100 377 377 377 377 377 377 377 377 377 377 377 377 377 377
*
```

Marker to indicate the start of a comment.
In hex they are bytes FF FE

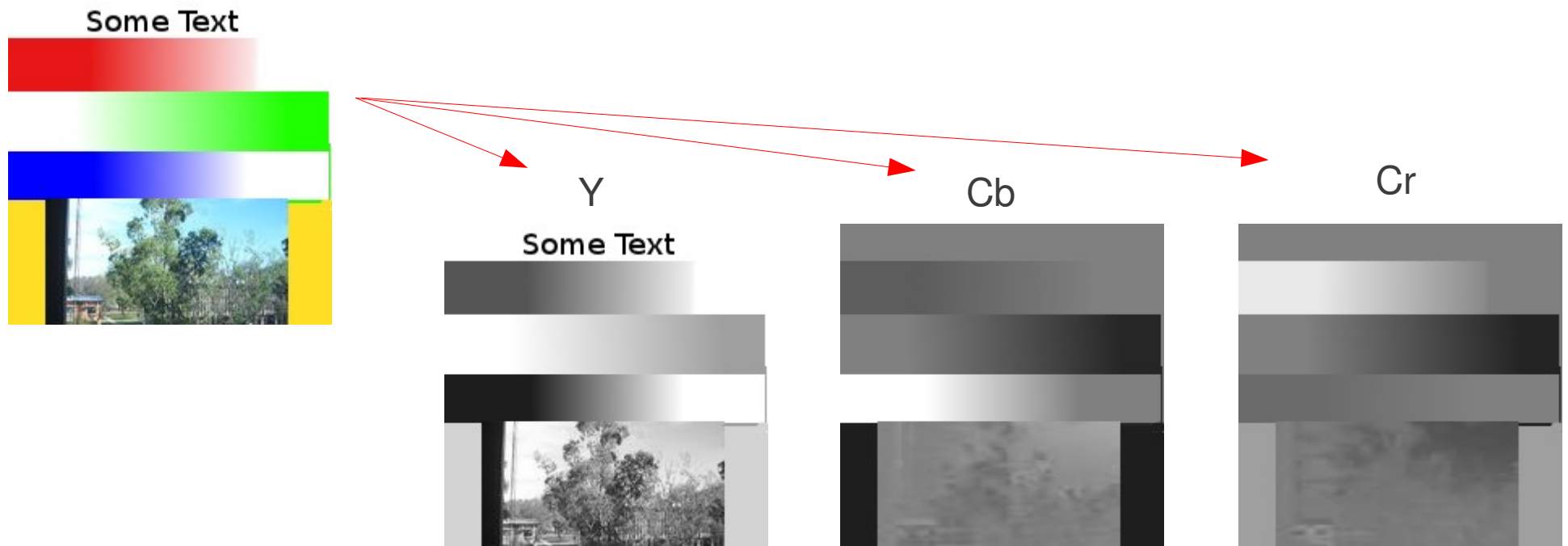
- Images would normally be converted to YCbCr and the chroma components downsampled.

$$Y = 0.299 R + 0.587 G + 0.114 B$$

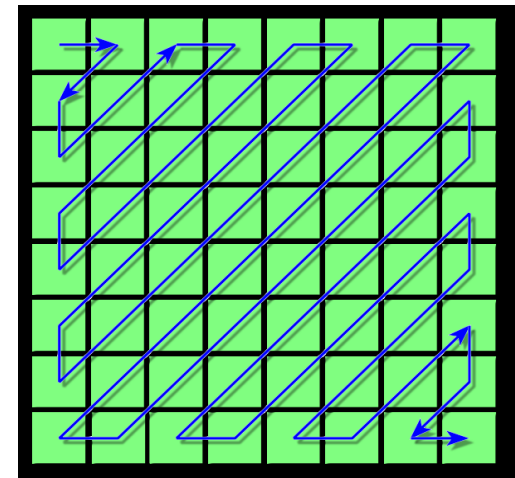
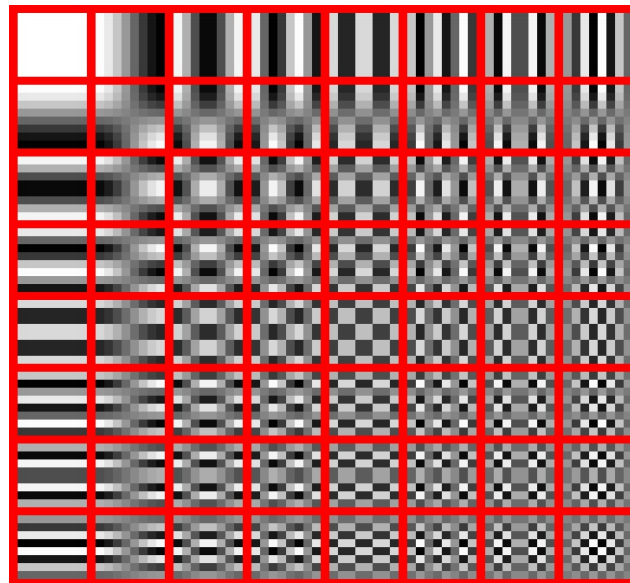
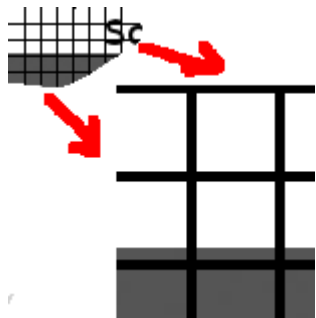
$$Cb = -0.1687 R - 0.3313 G + 0.5 B + 128$$

$$Cr = 0.5 R - 0.4187 G - 0.0813 B + 128$$

From "JPEG File Interchange Format"
V1.02, Eric Hamilton C-Cube Microsystem, 1992



- Y, Cb, and Cr are divided into 8x8 blocks, these blocks are then transformed into a 'frequency domain'(a DCT). This produces an 8x8 block of numbers representing a linear combination of the different frequencies. These are quantised by dividing by constant values and rounded (this is the step which governs compression and quality). Lossless Huffman encoding is then used on these blocks.



From http://en.wikipedia.org/wiki/File:JPEG_ZigZag.svg,
Alex Khristov – Public Domain

From <http://en.wikipedia.org/wiki/File:Dctjpeg.png> - public domain

- PNG uses lossless data compression to store bitmap information. PNG includes RGB, RGBA, and greyscale colour spaces. Either a pallet or intensity channel approach is used.
- Image data first undergoes pre-compression filter and then the DEFLATE (combining LZ77 and Huffman) compression algorithm is used.

- PNG files start with a unique file header. After the header is a number of 'chunks' of data. These 'chunks' contain: length, type, data, and crc parts.
- The IHDR chunk appears first and contains (from RFC2083):
 - Width: 4 bytes
 - Height: 4 bytes
 - Bit depth: 1 byte
 - Color type: 1 byte
 - Compression method: 1 byte
 - Filter method: 1 byte
 - Interlace method: 1 byte

```
ericm@ericm-desktop:~/courses/cg/notes/formats$ od -t x1 SimpleImage.png | head
0000000 89 50 4e 47 0d 0a 1a 0a 00 00 00 0d 49 48 44 52
0000020 00 00 00 c8 00 00 00 c8 08 06 00 00 00 ad 58 ae
```

- PNG files start with a unique file header. After the header is a number of 'chunks' of data. These 'chunks' contain: length, type, data, and crc parts.
- Chunks have length(4 bytes), type (4 bytes), data(variable), crc (4 bytes). Key chunks in PNG are: IHDR – header info; PLTE – pallet information; IDAT – image data; IEND – end marker.

```
ericm@ericm-desktop:~/courses/cg/notes/formats$ od -c SimpleImage.png | head
0000000 211  P  N  G  \r  \n 032  \n  \0  \0  \0  \r  I  H  D  R
0000020  \0  \0  \0 310  \0  \0  \0 310  \b 006  \0  \0  \0 255  X 256
0000040 236  \0  \0  \0 001  s  R  G  B  \0 256 316 034 351  \0  \0
0000060  \0 006  b  K  G  D  \0 377  \0 251  \0  = 275 346  V  \r
0000100  \0  \0  \0  \t  p  H  Y  s  \0  \0  \v 023  \0  \0  \v 023
0000120 001  \0 232 234 030  \0  \0  \0  \a  t  I  M  E  \a 333  \n
0000140 033  \0  - 6 226 025  A 005  \0  \0  \0 031  t  E  X  t
0000160  C  o  m  m  e  n  t  \0  C  r  e  a  t  e  d
0000200  w  i  t  h  \0  G  I  M  P  W 201 016 027  \0  \0
0000220  \0  I  D  A  T  x 332 354 275  y 220  $ 347  Y 356 373
```