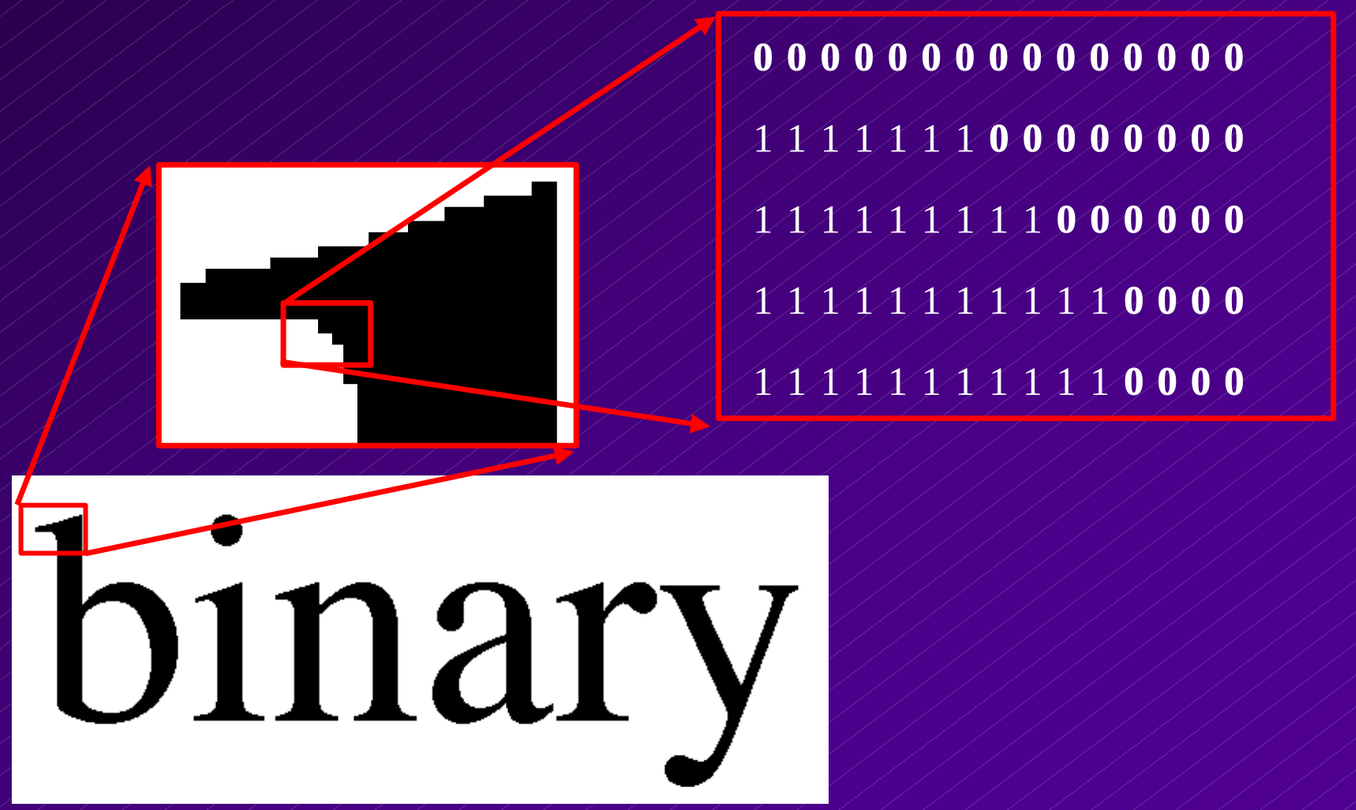
Binary Images

- The simplest digital images are *binary* images. Binary images contain only *one bit per pixel*, so they can only represent two gray values. For example;

0 = black

1 = white



Computer Memory & Storage

- If we want an image that has more than two gray levels, we have to increase the number of *'bits per pixel'*

binary: just white or black

grayscale: many shades of gray

Computer Memory & Storage

1 bit / pixel

0
1

2 gray levels

2 bits / pixel

0 0
0 1
1 0
1 1

2^2 = 4 gray levels

Computer Memory & Storage

3 bits / pixel

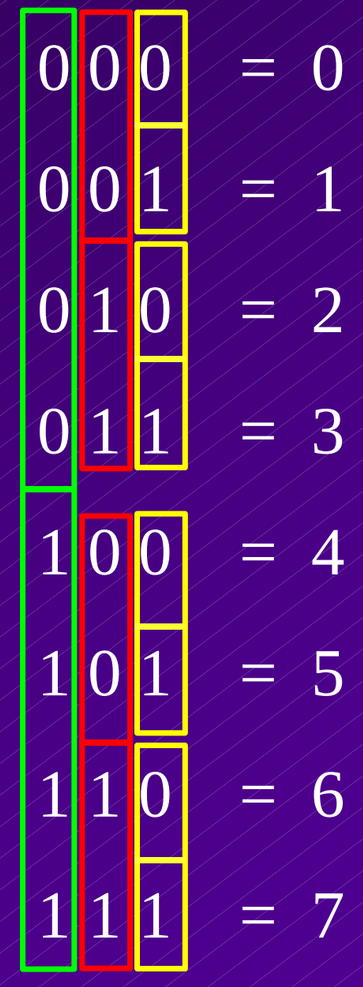
0	0	0
0	0	1
0	1	0
0	1	1
1	0	0
1	0	1
1	1	0
1	1	1

$2 \times 2 \times 2 = 8$ gray levels

Computer Memory & Storage

- We started to look at the bits as tokens to represent different values, but we ended up with a binary counting system.

- The largest number we can count to (and the number of different gray levels we can have) depends on how many bits we use.



Grayscale images

- To get more than two gray values, we need a code with more than one *bit per pixel*.

1 bit	2 bits	3 bits
0	00	000
1	01	001
	10	010
	11	011
		100
		101
		110
		111
(2 values)	(4 values)	(8 values)

Binary Arithmetic

- In binary arithmetic, we can only count from *0* to *1* before we have to ‘carry’

- To increase the number of different values that can be represented with a binary number, (and the number of gray levels in a digital image) we have to increase the number of bits

binary	decimal	
0	0	1 bit
1	1	
10	2	2 bits
11	3	
100	4	
101	5	
110	6	
111	7	3 bits
1000	8	
1001	9	
1010	10	
1011	11	
1100	12	
1101	13	
1110	14	
1111	15	4 bits
10000	16	
...	...	

Computer Memory & Storage

3 bits/pixel: 8 gray levels

000 ® 111

(0 ® 7)

4 bits/pixel: 16 gray levels

0000 ® 1111

(0 ® 15)

Computer Memory & Storage

5 bits/pixel: 32 gray levels

00000 ® 11111

(0 ® 31)

8 bits/pixel: 256 gray levels

00000000 ® 11111111

(0 ® 255)

Grayscale Images

- The number of gray levels that can be represented is fixed by the *bit depth*, the number of bits per pixel used to store the gray value.

1 bit/pixel : 2 values (‘binary’)
[0, 1]

2 bits/pixel : 4 values
[00, 01, 10, 11]

3 bits/pixel : 8 values
[000, 001, 010, …]

4 bits/pixel : 16 values
[0000, 0001, 0010, …]

Grayscale Images

- The number of gray levels that can be represented is fixed by the *bit depth*, the number of bits per pixel used to store the gray value.

5 bits/pixel : 32 values

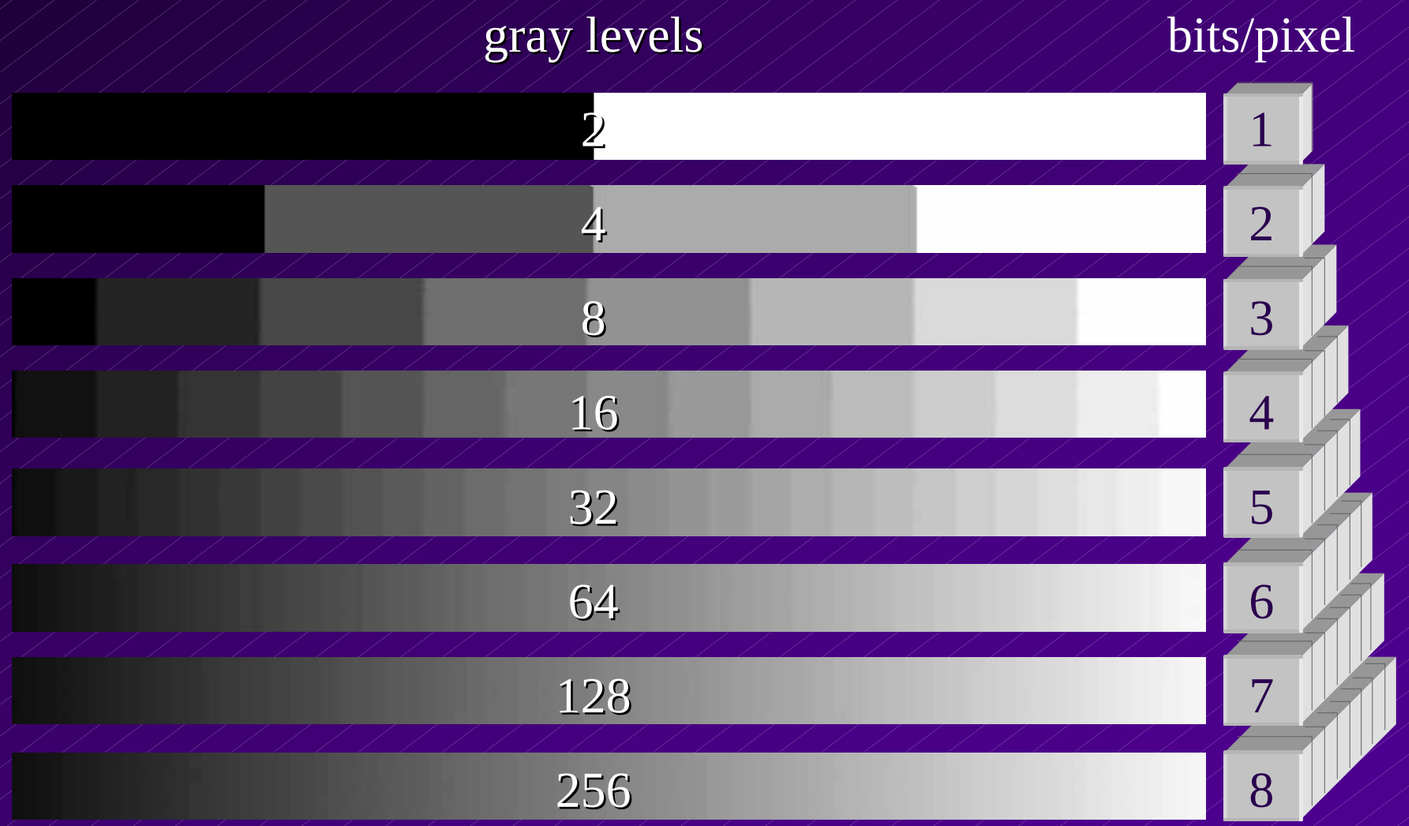
6 bits/pixel : 64 values

7 bits/pixel:128 values

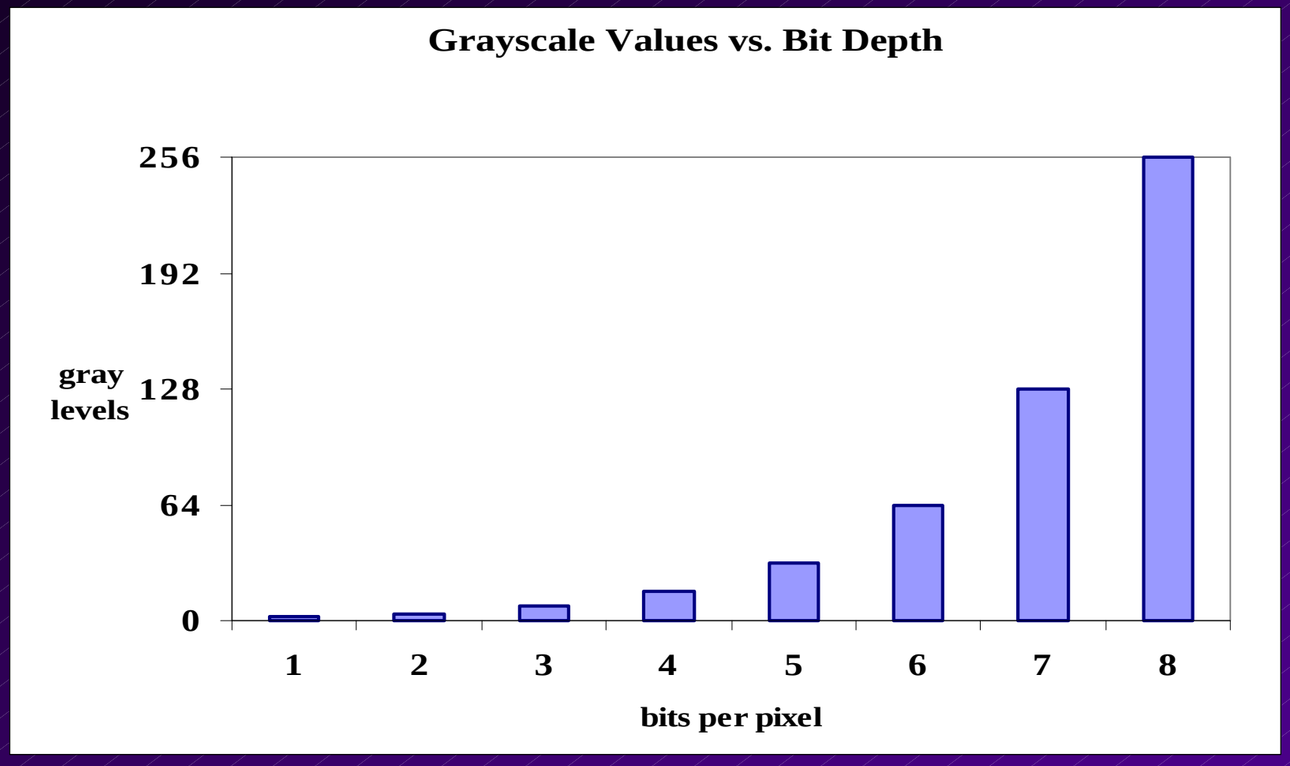
8 bits/pixel : 256 values

Bit depth: bits per pixel

- The number of possible gray levels is controlled by the number of bits/pixel, or the *‘bit depth’* of the image

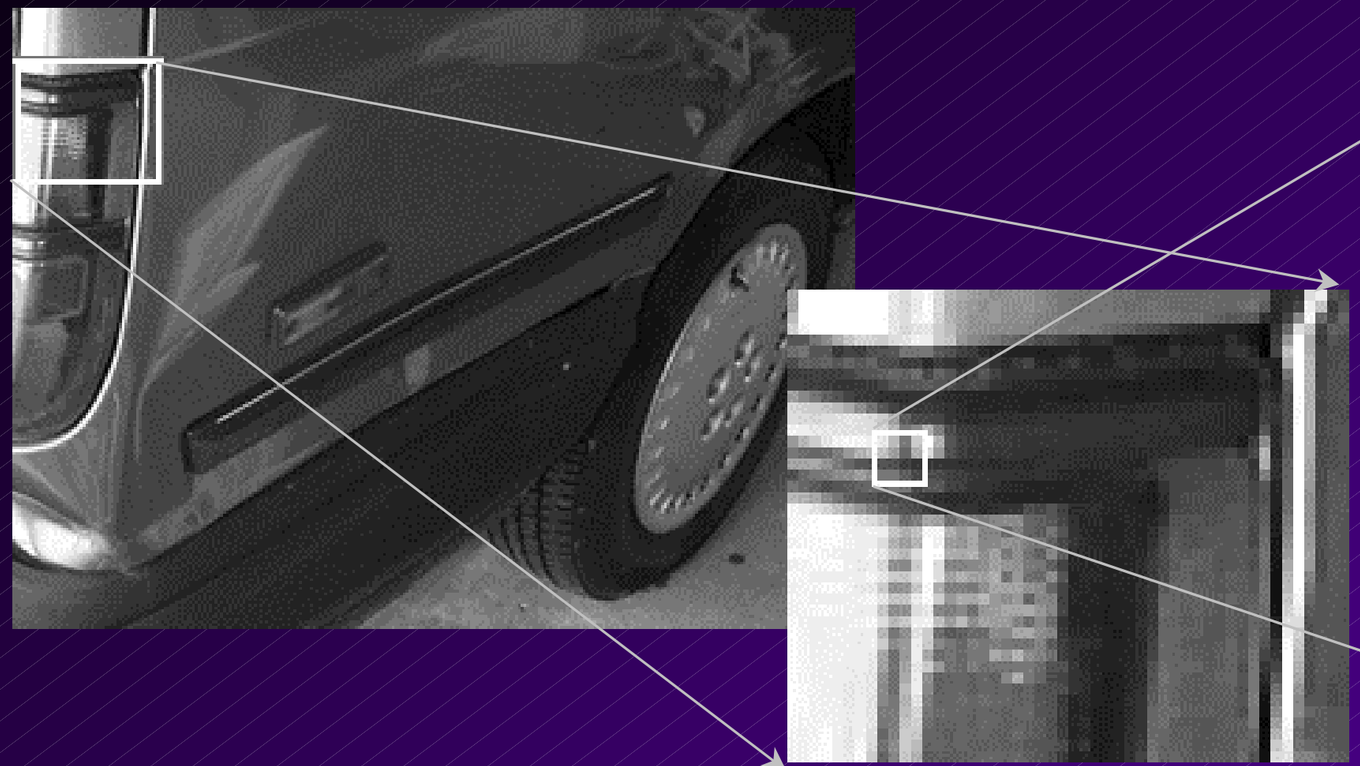


Memory requirements: Bit depth



- Adding more gray levels is ‘cheap’ in terms of memory requirements. Every added *bit* doubles the number of gray levels

Digital images: Fundamentals



228	178	106	193
183	143	84	162
64	62	45	39
101	99	75	56

- A digital image is an ‘ordered array’ of numbers
- Each pixel (picture element) in a grayscale digital image is a number that describe the pixel’s lightness

  (e.g., 0 = black 255 = white)

Grayscale Images

- Grayscale images commonly have 256 different gray values, numbered 0 - 255. Each pixel can then be stored in 8 bits, or 1 byte. [00000000 → 11111111]

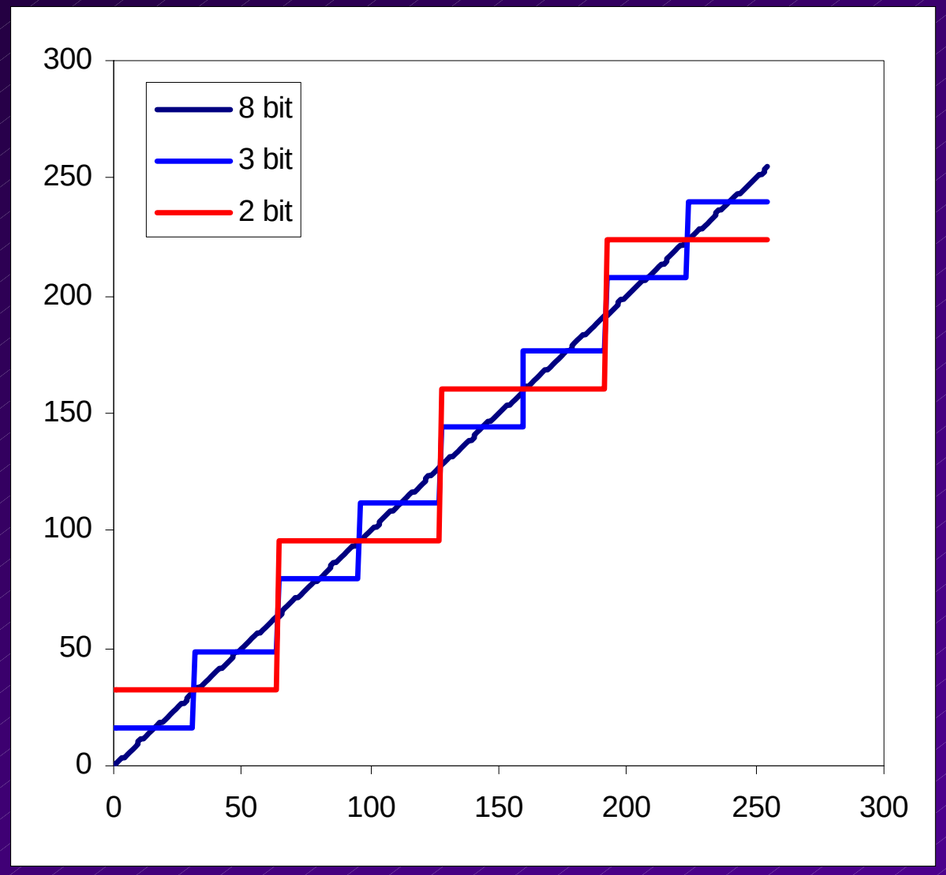
0 = black 255 = white

- Grayscale pixels are sometimes stored with as many as 1024 gray values (10 bits) or 4096 gray values (12 bits) This doesn’t make the image ‘look better’ but it increases the lightness range that can be captured

Bit depth & spatial resolution

The bit depth describes the ‘grayscale resolution’

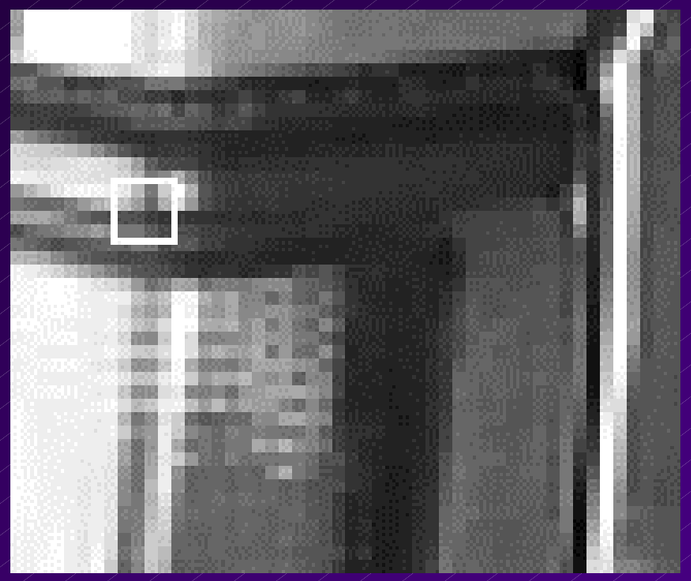
- with what precision are gray values distinguished?



Bit depth & spatial resolution

Spatial resolution

- with what precision are spatial variations reproduced?



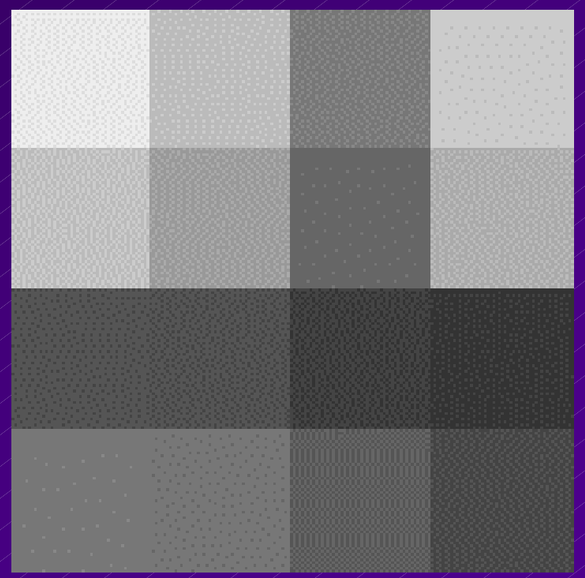


Image Resolution: 4 x 3 Pixels

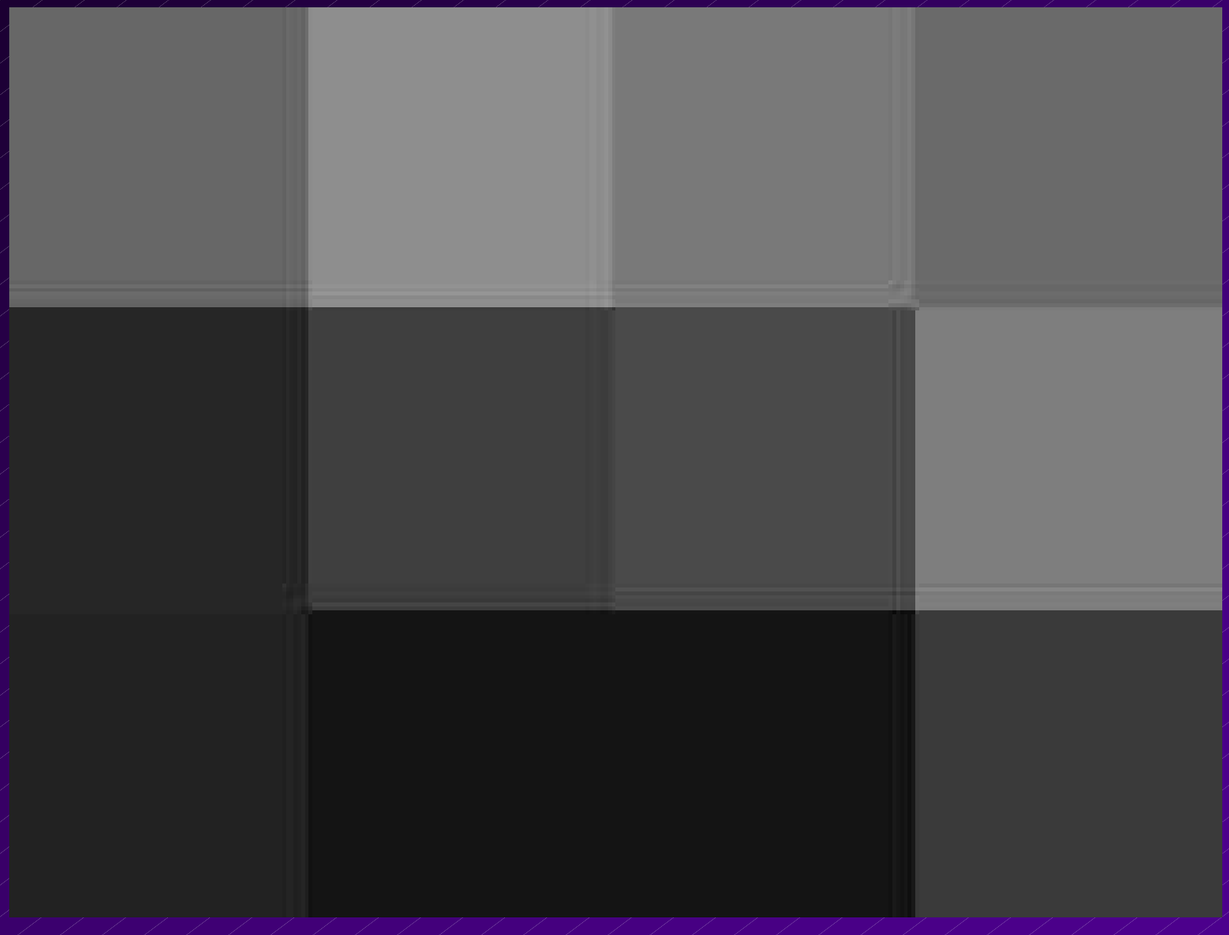


Image Resolution: 8 x 6 Pixels

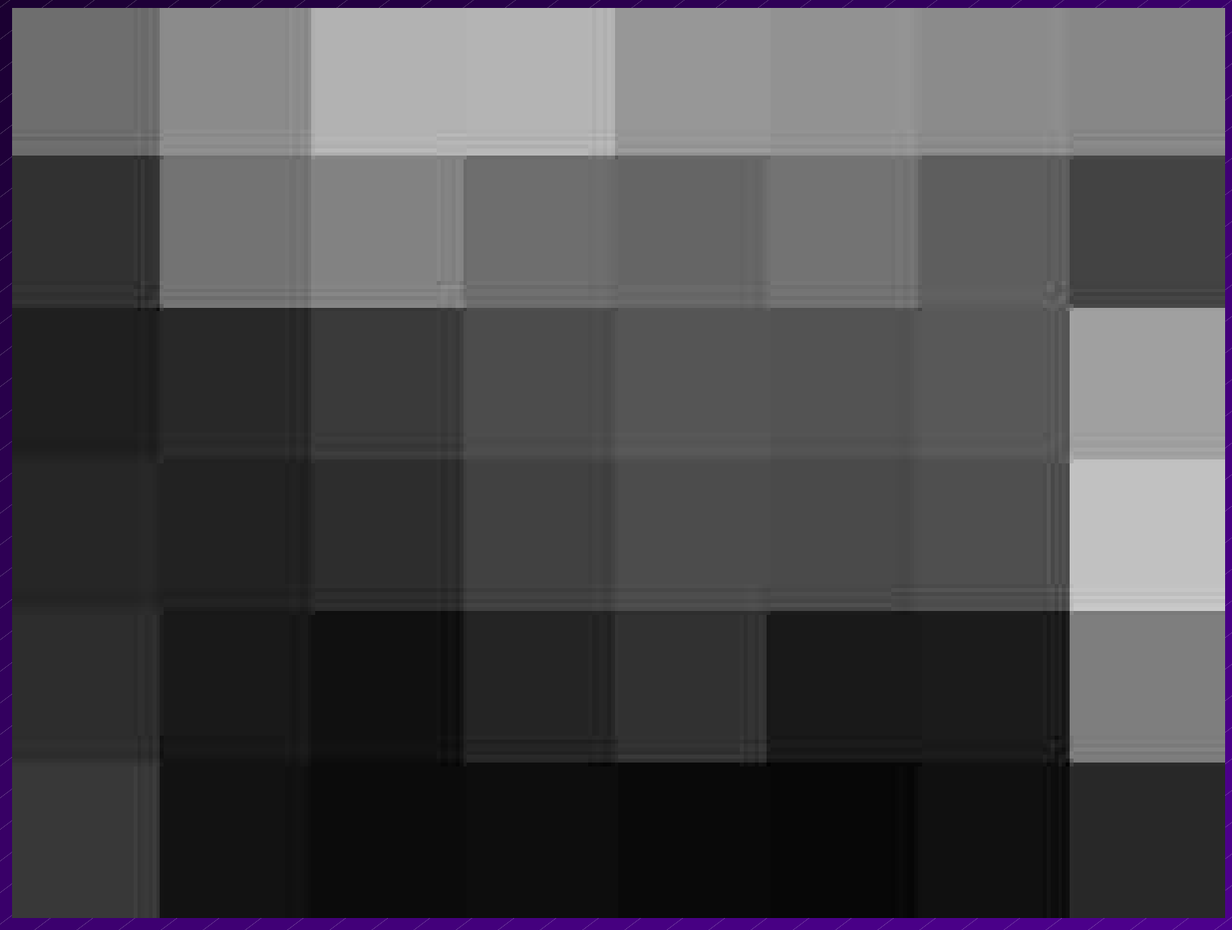


Image Resolution: 16 x 12 Pixels

Image Resolution: 32 x 24 Pixels

Image Resolution: 64 x 48 Pixels

Image Resolution: 128 x 96 Pixels

Image Resolution: 160 x 120 Pixels

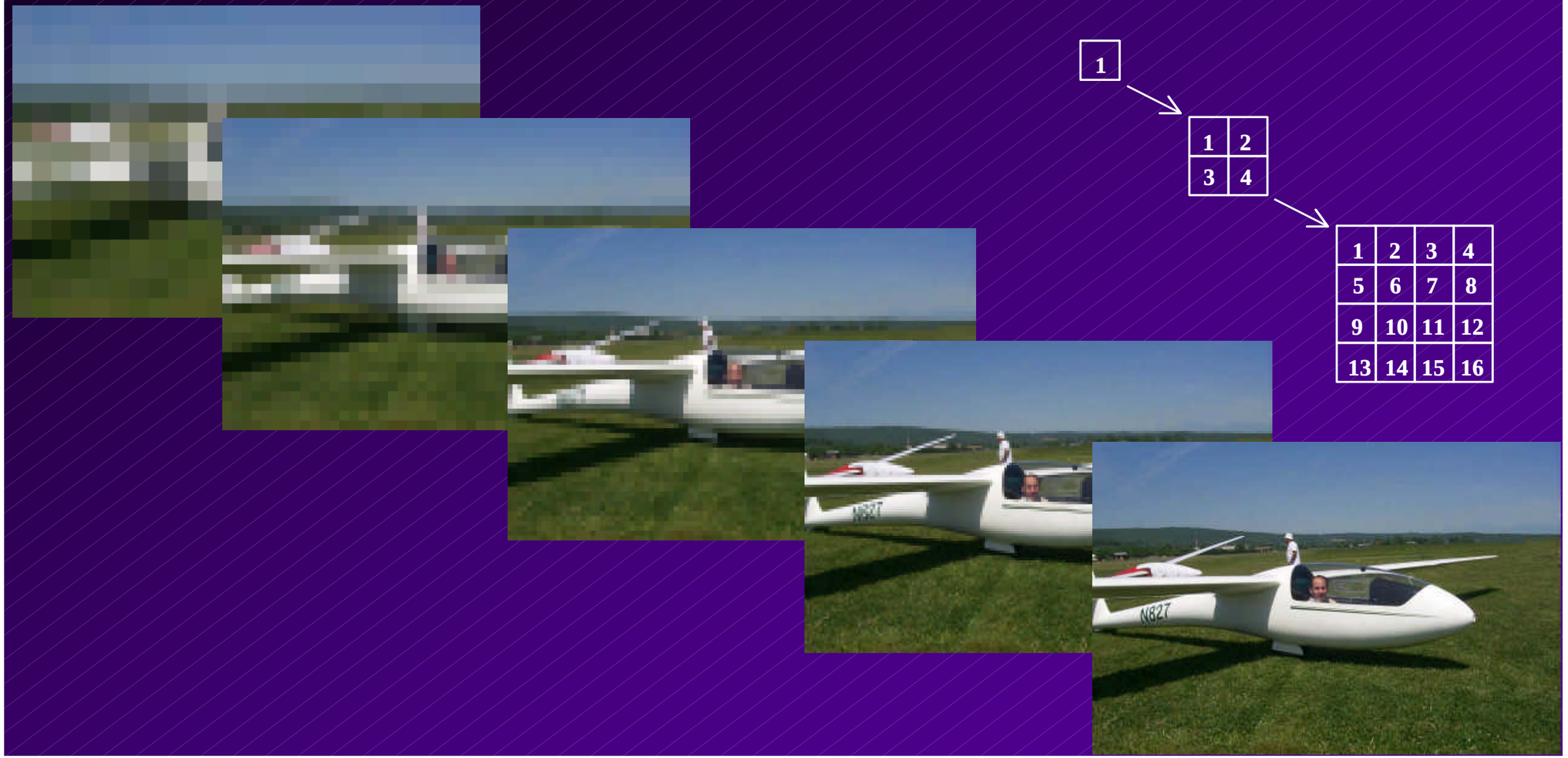
Image Resolution: 320 x 240 Pixels

Image Resolution: 640 x 480 Pixels

Image Resolution: 1280 x 960 Pixels*
NEW·YORK
MNW 691

Spatial Sampling & File Size

- Doubling the linear image sampling rate renders more image detail, but quadruples the file size.



RGB Color Images

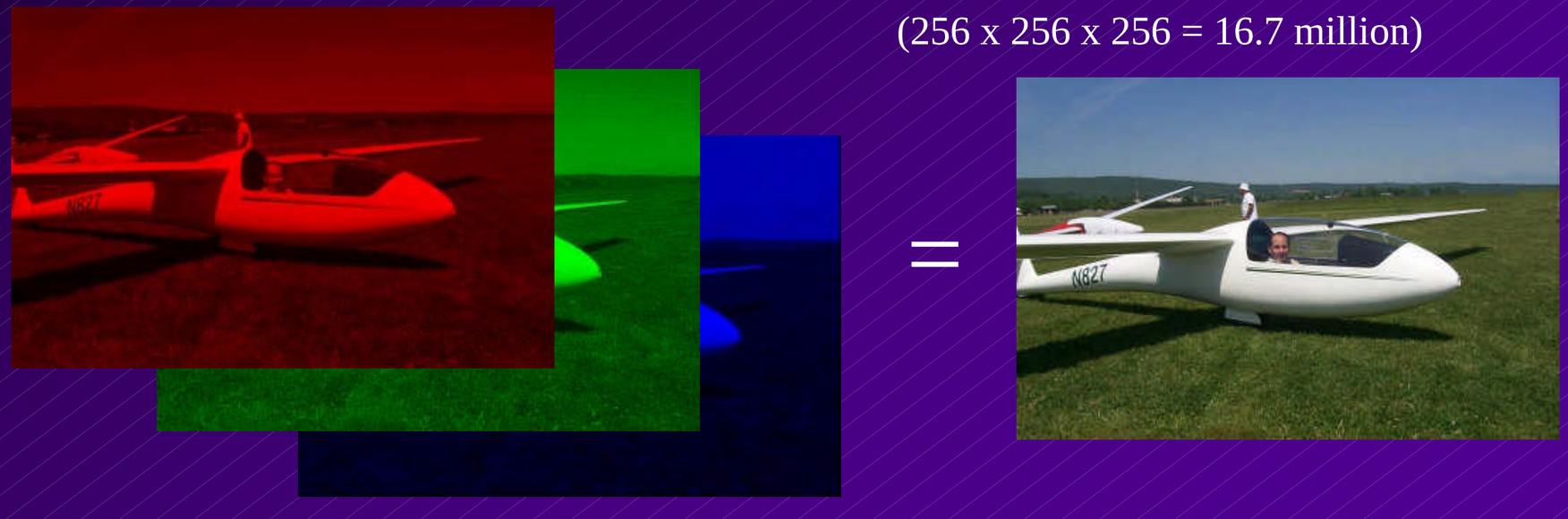
- The most straightforward way to capture a color image is to capture three images; one to record how much red is at each point, another for the green, and a third for the blue.

- Each one of the color images (‘planes’) is like a grayscale image, but is displayed in R, G, or B

Color images: 24-bit RGB

- Color images also need to be coded
- The bit depth in a color image determines the number of colors that can be assigned to a given pixel.
- One common format is the *24-bit* RGB image, with three 8-bit planes; Red, Green, and Blue; 16.7M colors

(256 x 256 x 256 = 16.7 million)



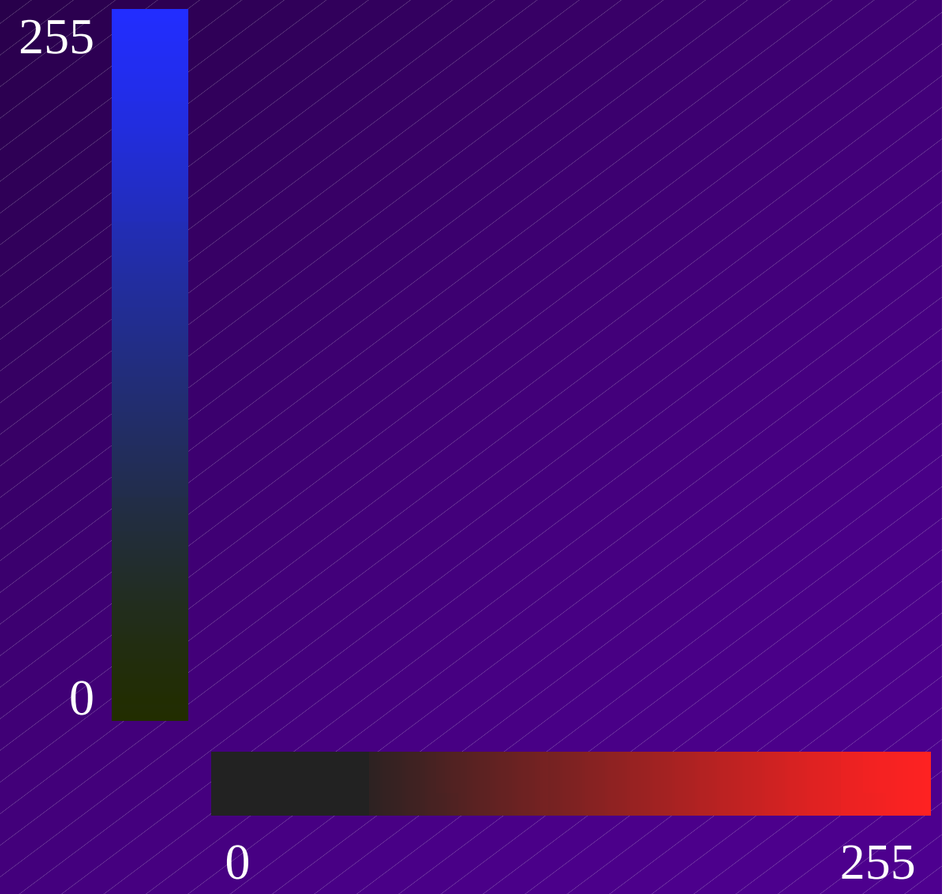
RGB Color Images: 24-bit color

- Every pixel in each of the three 8-bit color planes can have 256 different values (0-255)

- If we start with just the blue image plane, we can make 256 different “colors of blue”

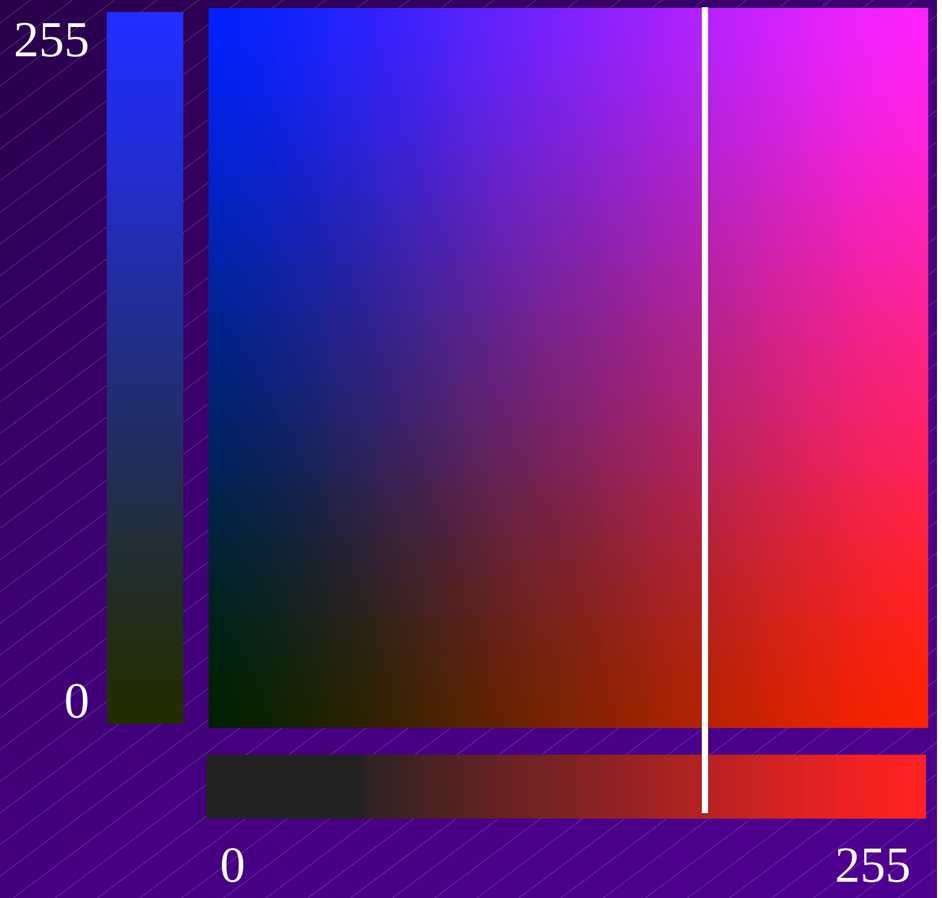
RGB Color Images: 24-bit color

- Every pixel in each of the three 8-bit color planes can have 256 different values (0-255)
- If we start with just the blue image plane, we can make 256 different “colors of blue”
- If we add red (which alone gives us 256 different reds):



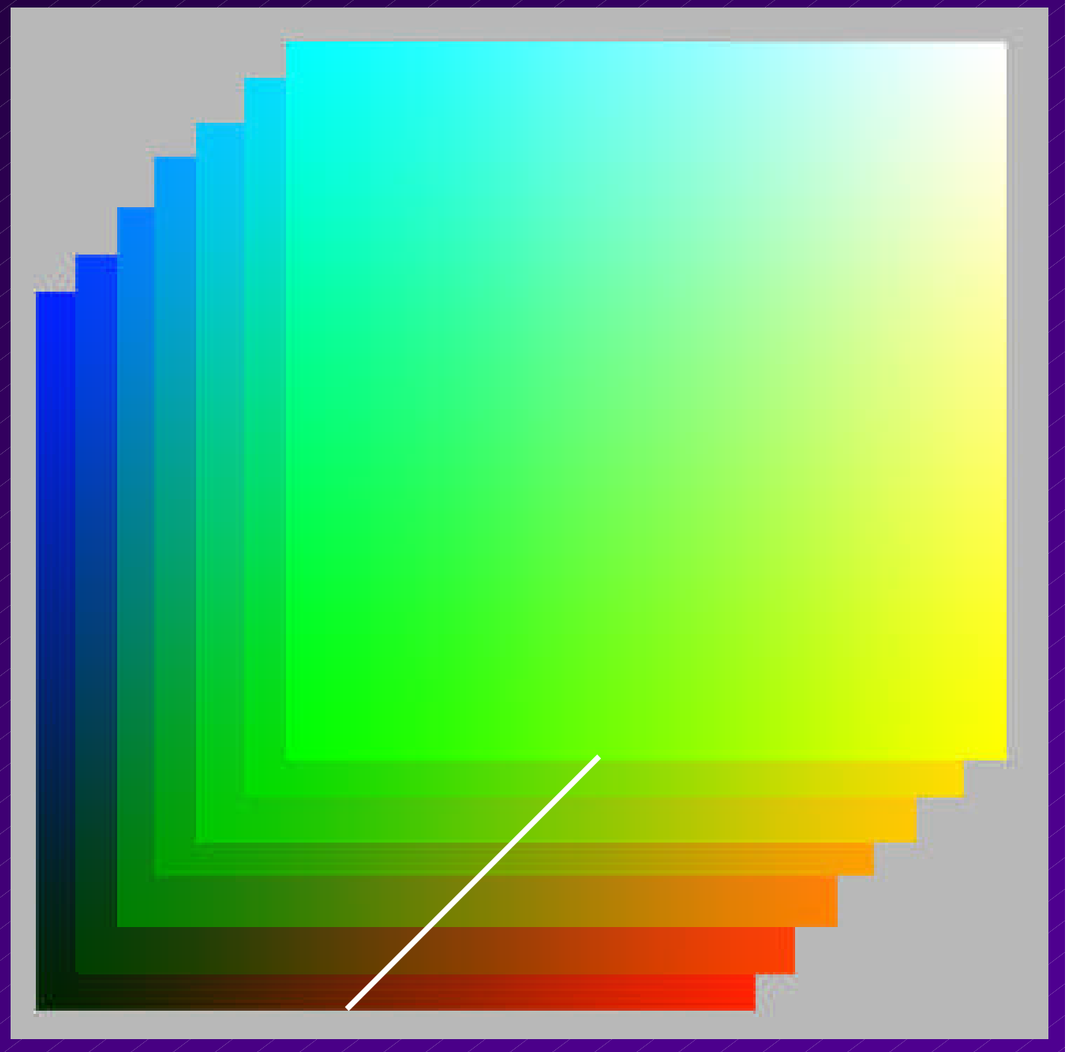
RGB Color Images: 24-bit color

- Every pixel in each of the three 8-bit color planes can have 256 different values (0-255)
- If we start with just the blue image plane, we can make 256 different “colors of blue”
- If we add red (which alone gives us 256 different reds):
- We can make 256 x 256 = 65,536 combination colors because for every one of the 256 reds, we can have 256 blues.



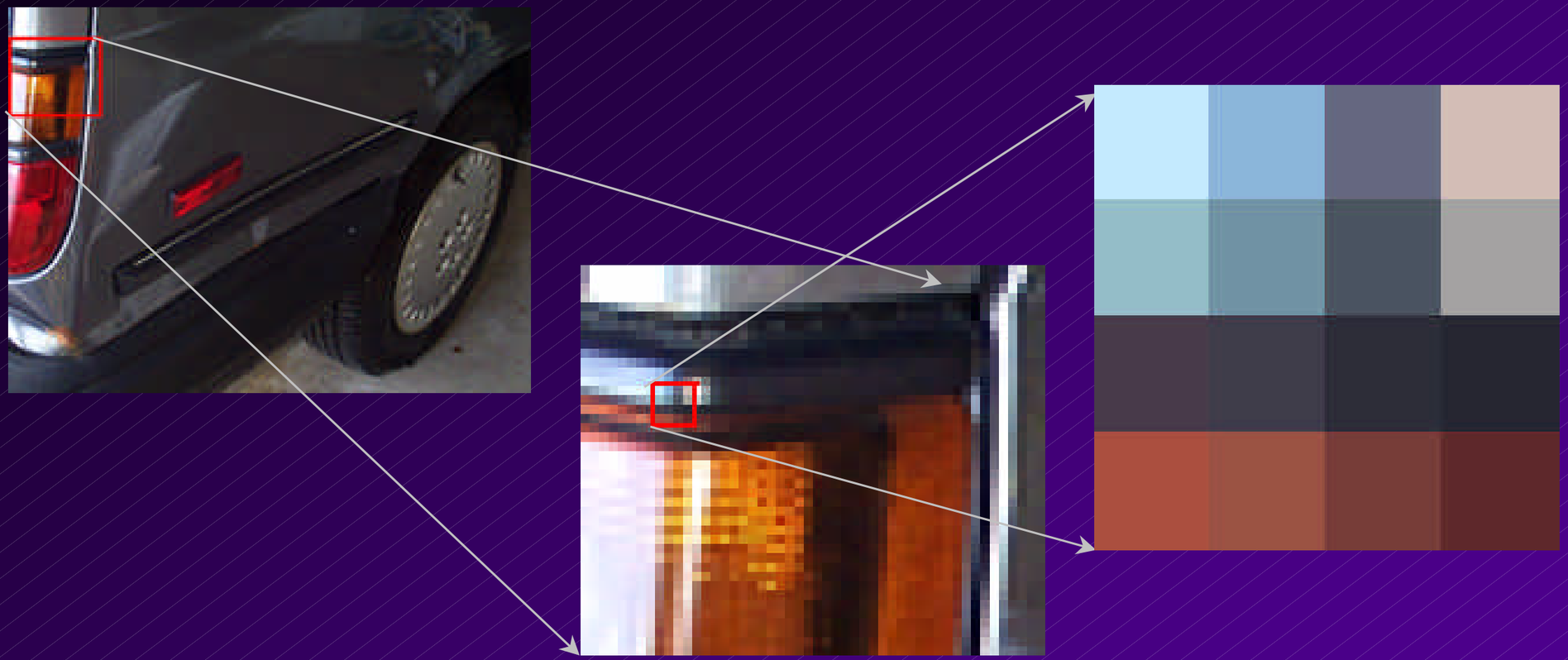
RGB Color Images: 24-bit color

- When we have all three colors together, there are 256 possible values of green for each one of the 65,536 combinations of red and blue:



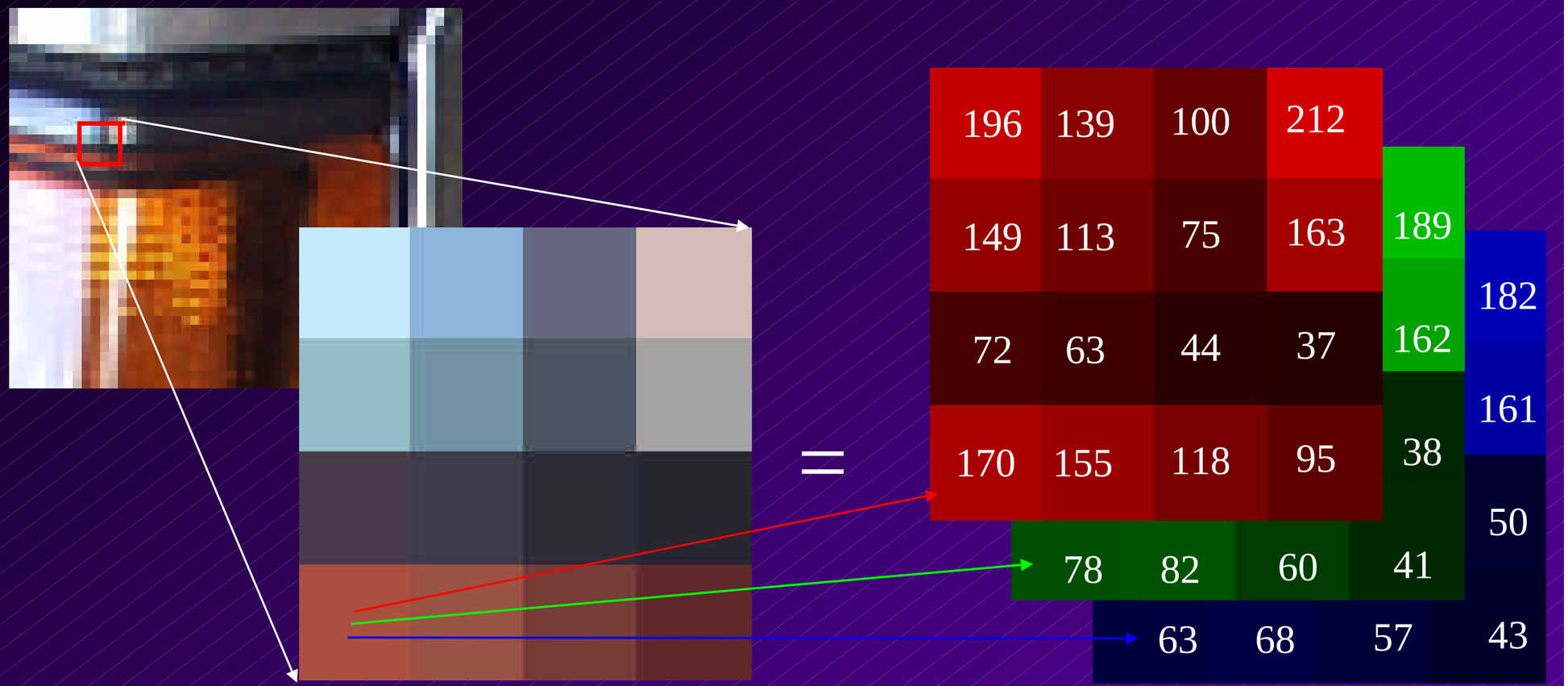
- 256 x 256 x 256 = 16,777,216 (“> 16.7 million colors”)

RGB Color Images: 24-bit color



- The numbers stored for each pixel in a color image contain the color of that pixel

Color Image = Red + Green + Blue



- In a 24-bit image, each pixel has R, G, & B values
- When viewed on a color display, the three images are combined to make the color image.

Indexed Color Images

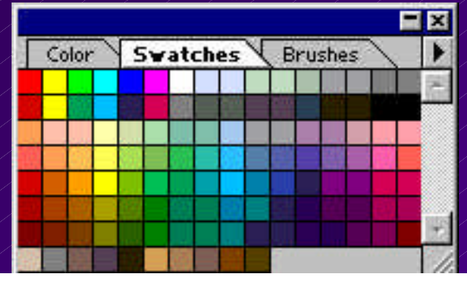
- A small subset of the 16 million colors can often be used instead of the full 24 bits -- 256 colors is often sufficient if the colors are chosen carefully

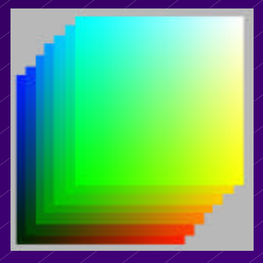
- Indexed color images take advantage of this fact to use less memory or work with displays that can’t show 24-bit images

Indexed Color images

24 bit

8-bit “system”





8-bit“adaptive”

Color images: Index Color

- A more compact code can be created for color images by making a *look-up-table* of colors for use in an image. *Indexed color* images store a fixed number of colors limited by the bit-depth:

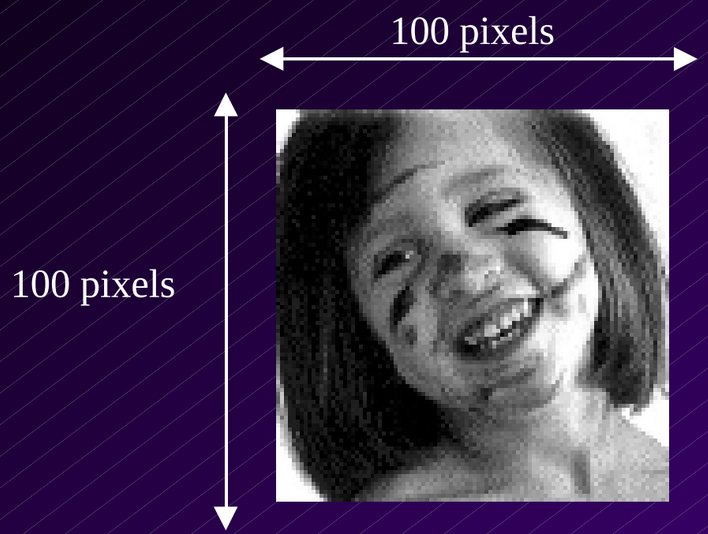
3 bits/pixel : 8 colors

4 bits/pixel : 16 colors

5 bits/pixel:64 colors

8 bits/pixel : 256 colors

File Size Calculation



Bit depth = 8 bits per pixel (256 gray levels)

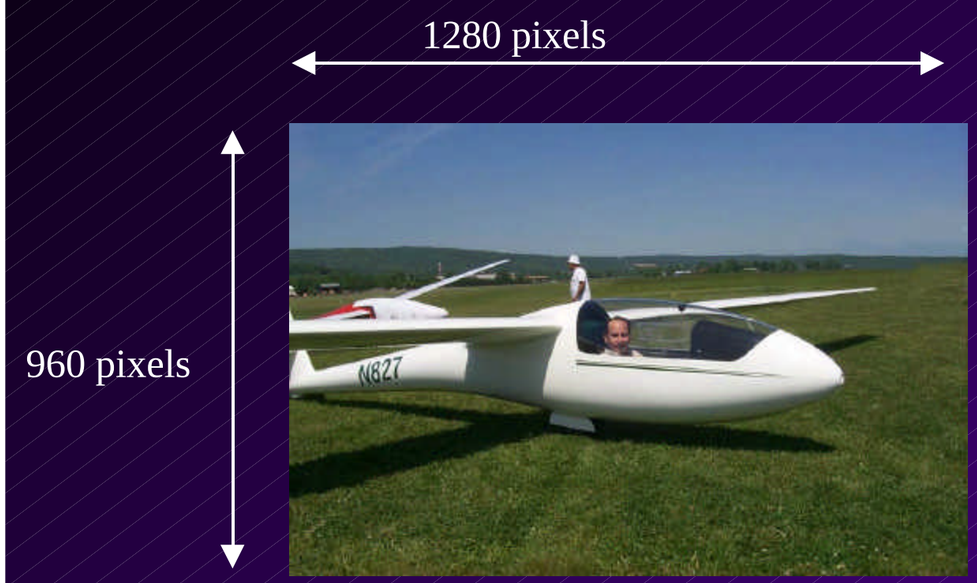
- How much memory is necessary to store an image that is 100 x 100 pixels with 8 bits/pixel?

File size (in bits) = Height x Width x Bit Depth

100 x 100 x 8 bits/pixel = 80,000 bits/image

80,000 bits or 10,000 bytes

File Size Calculation



- How much memory is necessary to store an image that is 1280 x 960 pixels with 24 bits/pixel?

Bit depth = 24 bits per pixel (RGB color)

File size (in bits) = Height x Width x Bit Depth

960 x 1280 x 24 bits/pixel = 29,491,200 bits/image

29,491,200 bits = 3,686,400 bytes = 3.5 MB

Raw image = 3,686 KB JPEG - Adobe PhotoShop “10” 324 KB

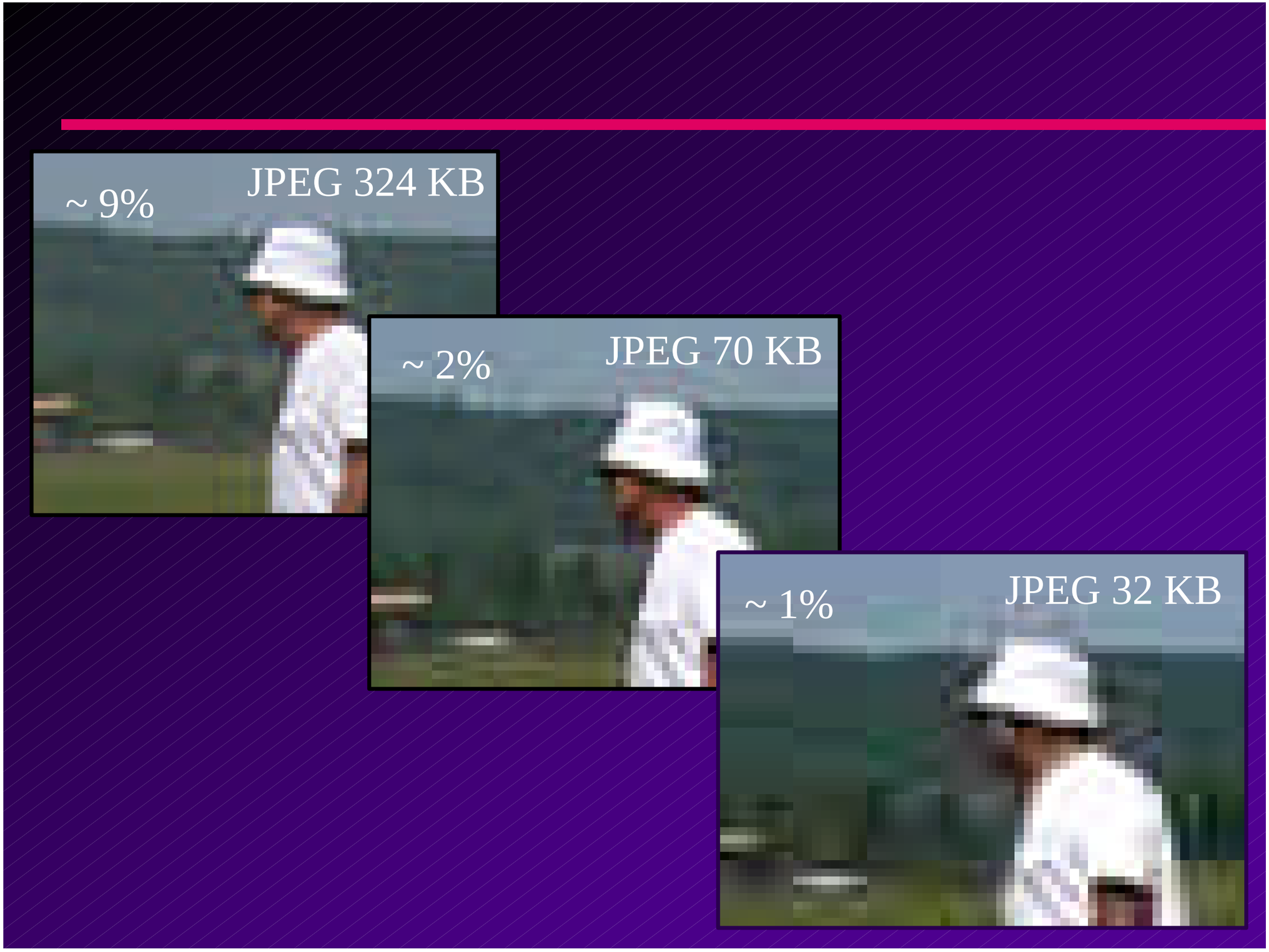
compressed/raw ~ 9%

Raw image = 3,686 KB JPEG - Adobe PhotoShop “5” 70 KB

compressed/raw ~ 2%

Raw image = 3,686 KB JPEG - Adobe PhotoShop “0” 32 KB

compressed/raw ~ 1%


~ 9%
JPEG 324 KB
~ 2%
JPEG 70 KB
~ 1%
JPEG 32 KB