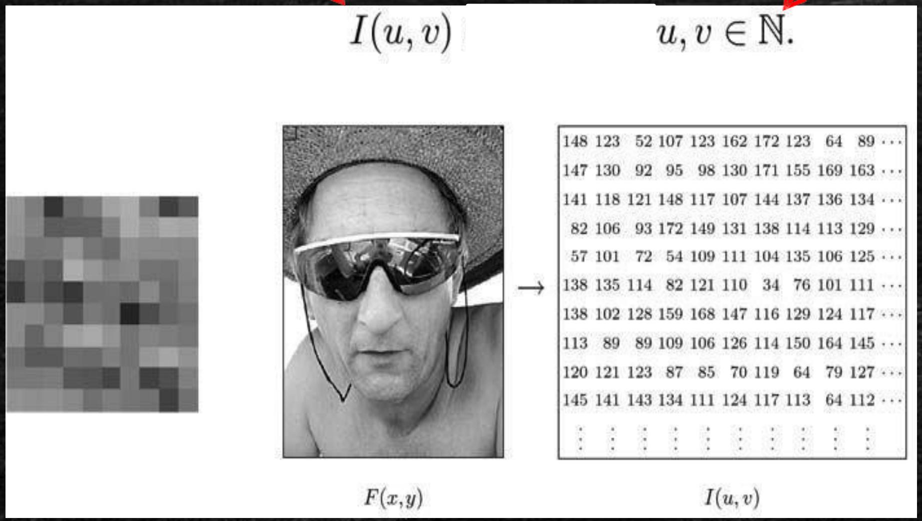
Images

Representing Images

- 2-dimensional matrix of Intensity (gray or color) values

Set of Intensity values

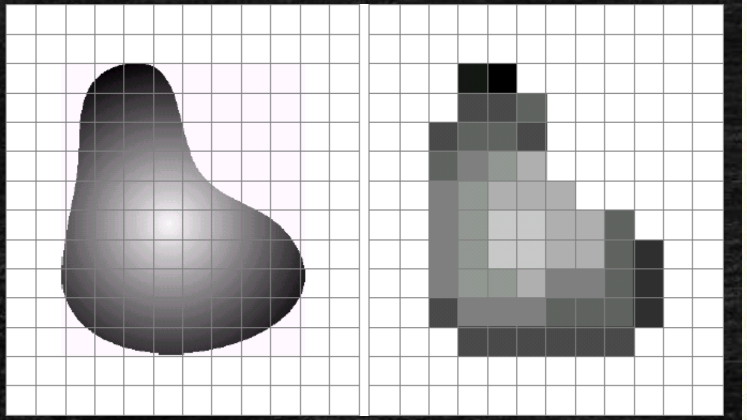
Image coordinates are integers



$\mathbb{N}$ Is the set of natural numbers. These are whole, non-negative numbers. Therefore N = {0, 1, 2, 3, …}

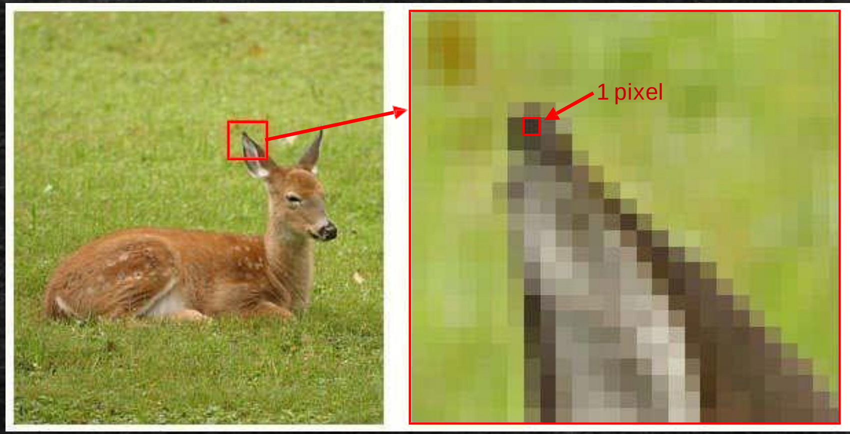
Digitization

- **Remember:** *digitization* causes a digital image to become an *approximation* of a real scene



Real image

Digital Image
(an approximation)



Real image

Digital Image
(an approximation)

Common Image Formats

- Common image formats include:
  - 1 values per point/pixel (B&W or Grayscale)
  - 3 values per point/pixel (Red, Green, and Blue)
  - 4 values per point/pixel (Red, Green, Blue, + “Alpha” or Opacity)

Grayscale

RGB

RGBA

Example of Digital Images

a) Natural landscape
b) Synthetically generated scene
c) Poster graphic
d) Computer screenshot
e) Black and white illustration
f) Barcode
g) Fingerprint
h) X-ray
i) Microscope slide
j) Satellite Image
k) Radar image
l) Astronomical object

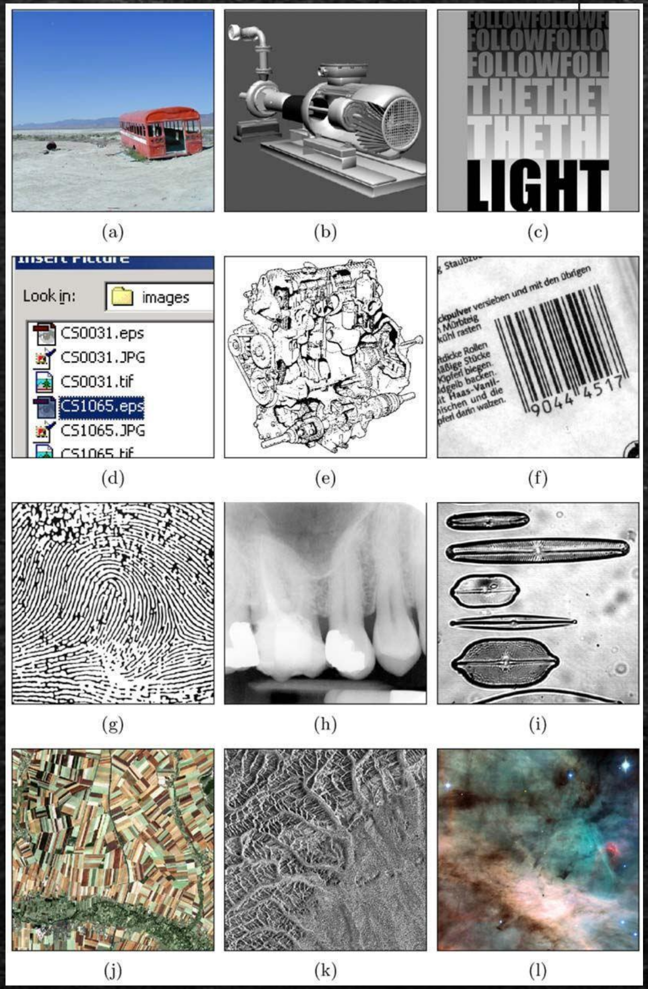


Image File Formats

- Hundreds of image file formats. Examples
  - Tagged Image File Format (TIFF)
  - Graphics Interchange Format (GIF)
  - Portable Network Graphics (PNG)
  - JPEG, BMP, Portable Bitmap Format (PBM), etc
- Image pixel values can be
  - Grayscale: 0 – 255 range
  - Binary: 0 or 1
  - Color: RGB colors in 0-255 range (or other color model)
  - Application specific (e.g. floating point values in astronomy)

How many Bits Per Image Element?

Grayscale (Intensity Images):

Chan.	*Bits/Pix.*	*Range*	*Use*
1	1	$0 \ldots 1$	Binary image: document, illustration, fax
1	8	$0 \ldots 255$	Universal: photo, scan, print
1	12	$0 \ldots 4095$	High quality: photo, scan, print
1	14	$0 \ldots 16383$	Professional: photo, scan, print
1	16	$0 \ldots 65535$	Highest quality: medicine, astronomy

Color Images:

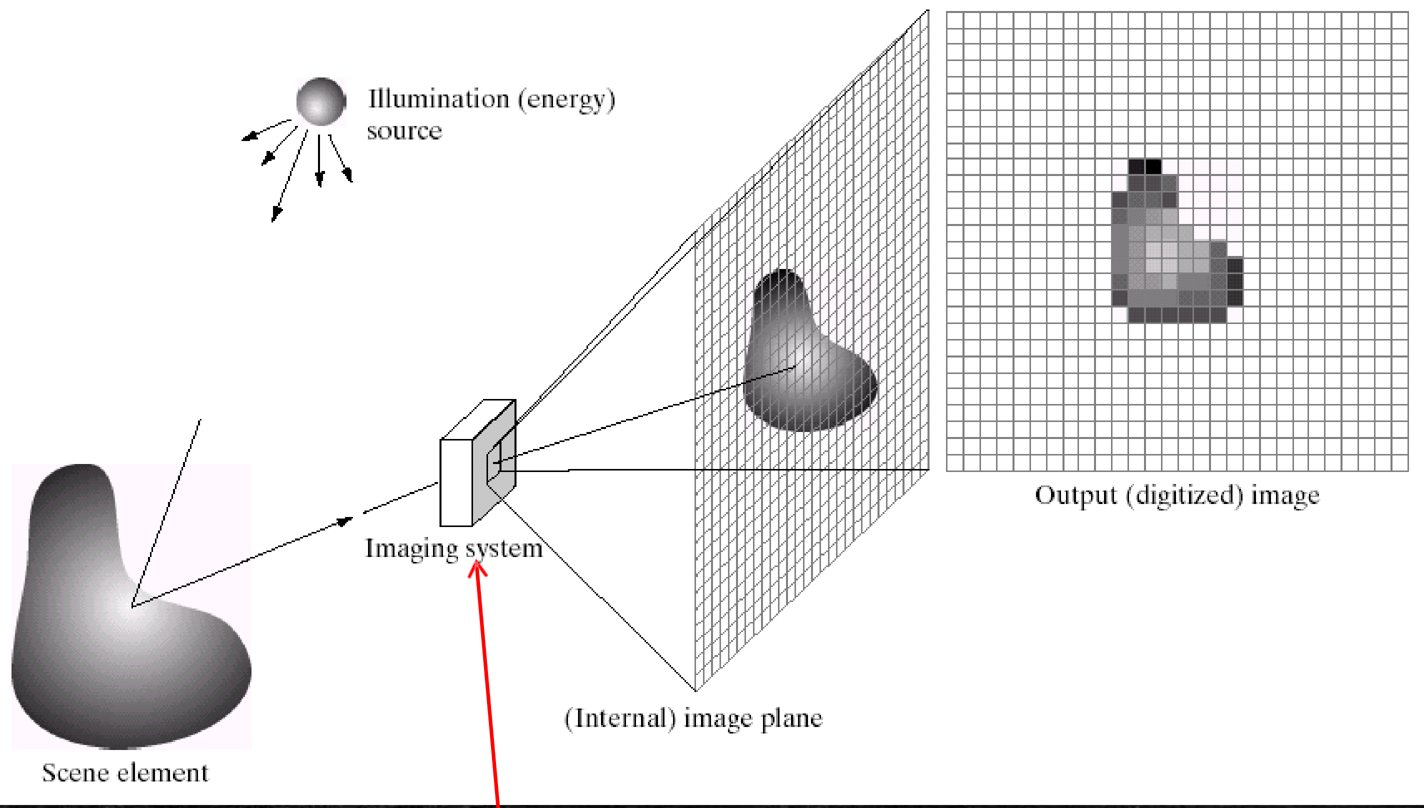
Chan.	*Bits/Pix.*	*Range*	*Use*
3	24	$[0 \ldots 255]^3$	RGB, universal: photo, scan, print
3	36	$[0 \ldots 4095]^3$	RGB, high quality: photo, scan, print
3	42	$[0 \ldots 16383]^3$	RGB, professional: photo, scan, print
4	32	$[0 \ldots 255]^4$	CMYK, digital prepress

Special Images:

Chan.	*Bits/Pix.*	*Range*	*Use*
1	16	$-32768 \ldots 32767$	Whole numbers pos./neg., increased range
1	32	$\pm 3.4 \cdot 10^{38}$	Floating point: medicine, astronomy
1	64	$\pm 1.8 \cdot 10^{308}$	Floating point: internal processing

Image Acquisition and Imaging Devices

Imaging System



Example: a camera
Converts light to image

Image Acquisition

- Images typically generated by ***illuminating a scene*** and absorbing energy reflected by scene objects

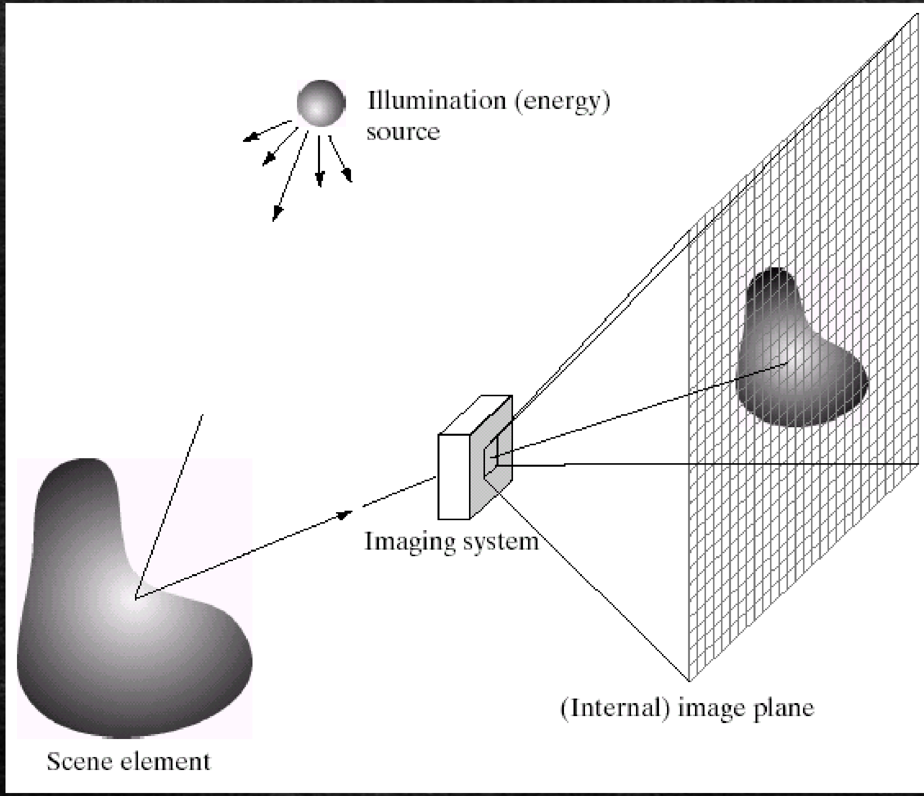
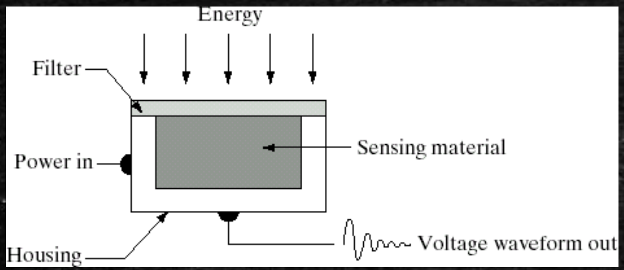
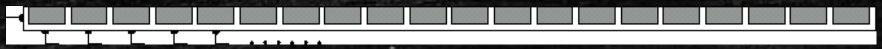


Image Sensing

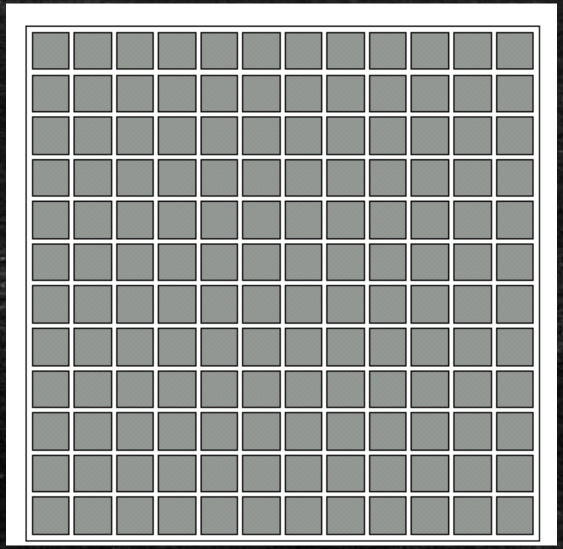
- Incoming energy (e.g. light) lands on a sensor material responsive to that type of energy, generating a voltage
- Collections of sensors are arranged to capture images



Imaging Sensor



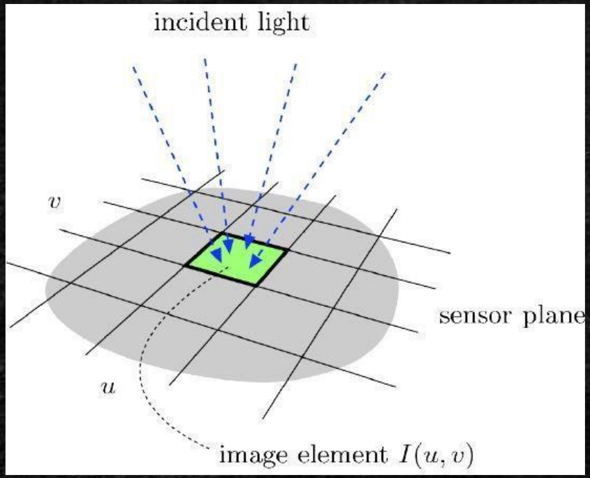
Line of Image Sensors



Array of Image Sensors

Spatial Sampling

- Cannot record image values for all (x,y)
- Sample/record image valuesat discrete (x,y)
- Sensors arranged in grid to sample image



$F(x,y)$ $\longrightarrow$

148	123	52	107	123	162	172	123	64	89	⋯
147	130	92	95	98	130	171	155	169	163	⋯
141	118	121	148	117	107	144	137	136	134	⋯
82	106	93	172	149	131	138	114	113	129	⋯
57	101	72	54	109	111	104	135	106	125	⋯
138	135	114	82	121	110	34	76	101	111	⋯
138	102	128	159	168	147	116	129	124	117	⋯
113	89	89	109	106	126	114	150	164	145	⋯
120	121	123	87	85	70	119	64	79	127	⋯
145	141	143	134	111	124	117	113	64	112	⋯
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	

$I(u,v)$

Image (Spatial) Sampling

- A digital sensor can only measure a limited number of **samples** at a **discrete** set of energy levels
- **Sampling** can be thought of as:

  Continuous signal x comb function

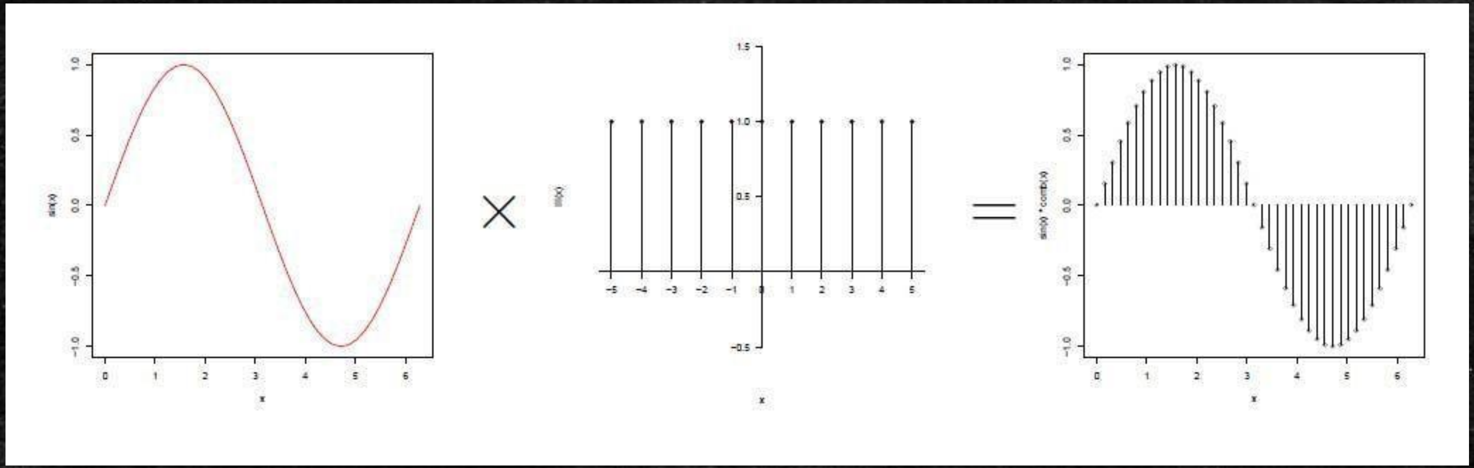


Image Quantization

- ***Quantization:*** process of converting continuous **analog** signal into its digital representation
- Discretize image $I(u,v)$ values
- Limit values image can take

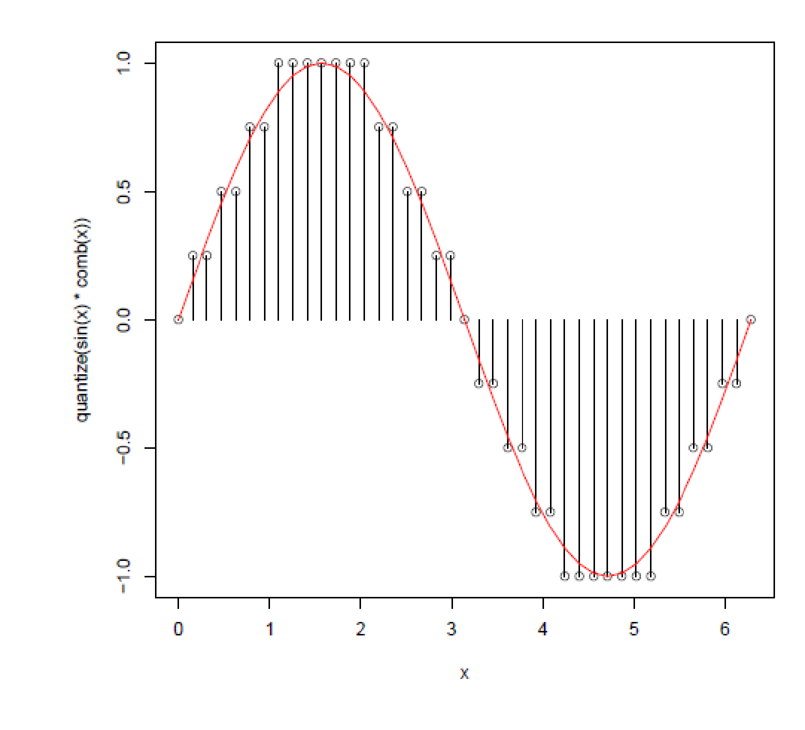
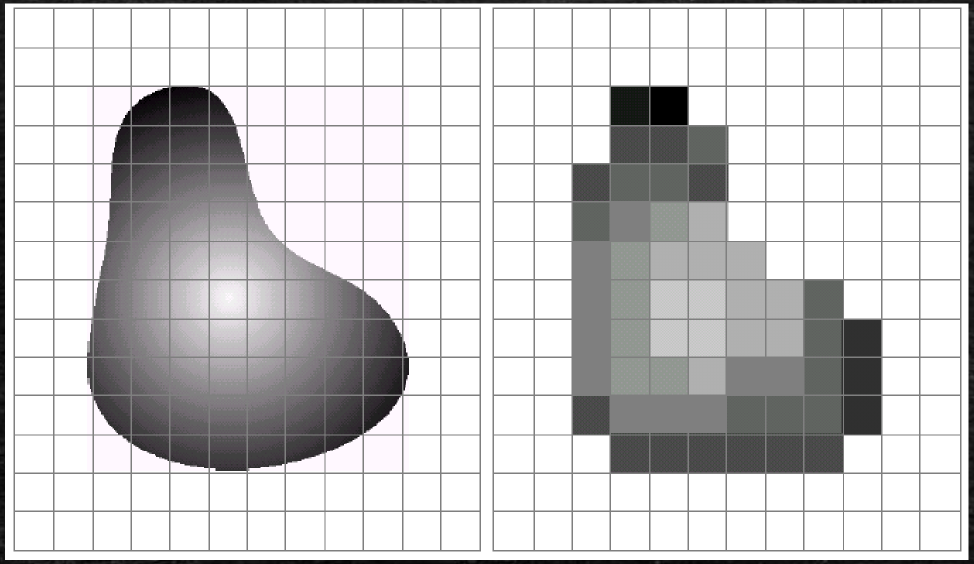


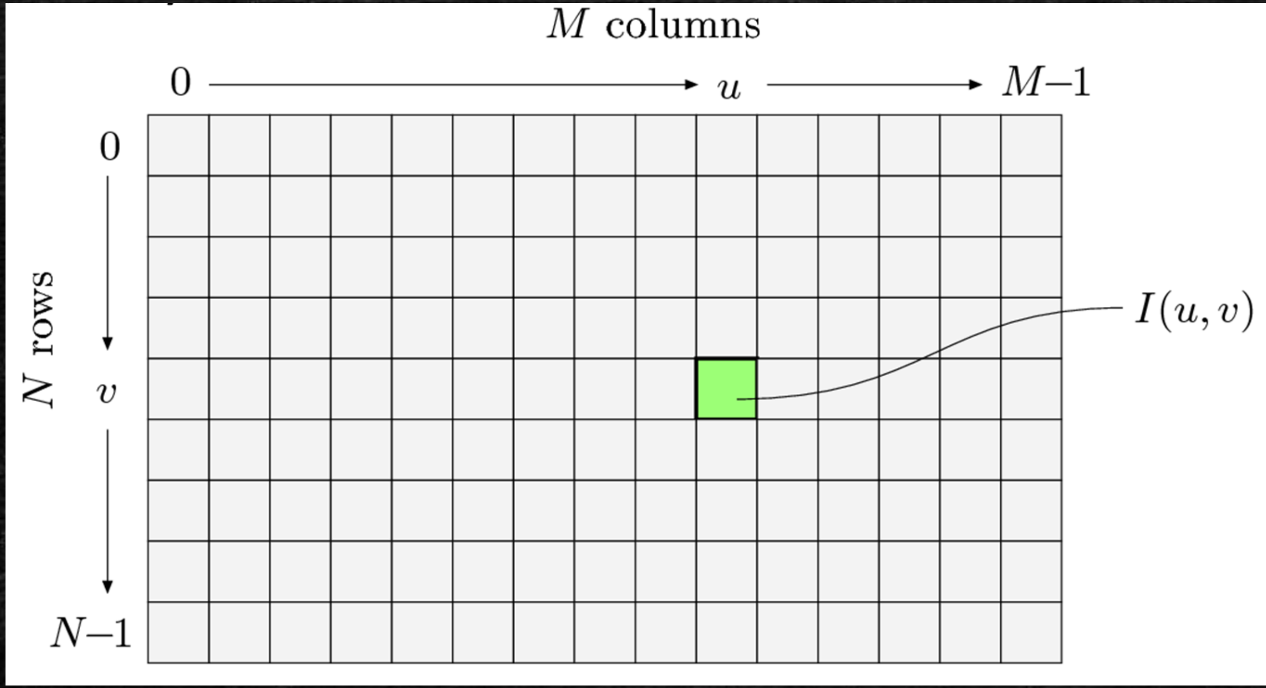
Image Sampling And Quantization

- Sampling and quantization generates approximation of a real world scene



Representing Images

- Image data structure is 2D array of pixel values
- Pixel values are gray levels in range 0-255 or RGB colors
- A numeric datatype (bit, byte, int, float, double, etc.)

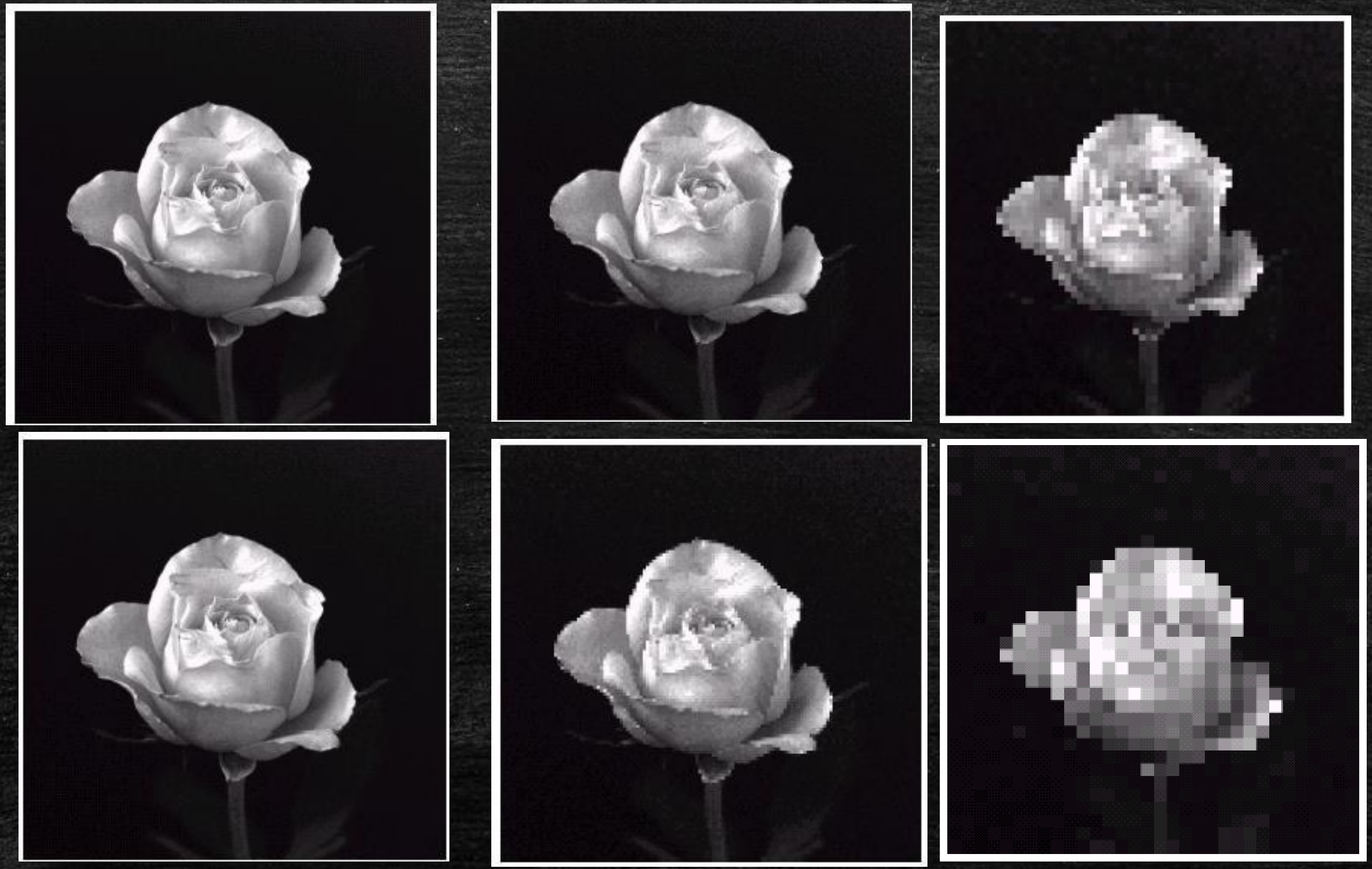


Spatial Resolution

- *The* ***spatial resolution*** of an image is determined by how fine/coarse sampling was carried out
- **Spatial resolution:** smallest discernable image detail
  - Vision specialists talk about image resolution
  - Graphic designers talk about *dots per inch* (DPI)

Spatial Resolution

Spatial Resolution: Stretched Images

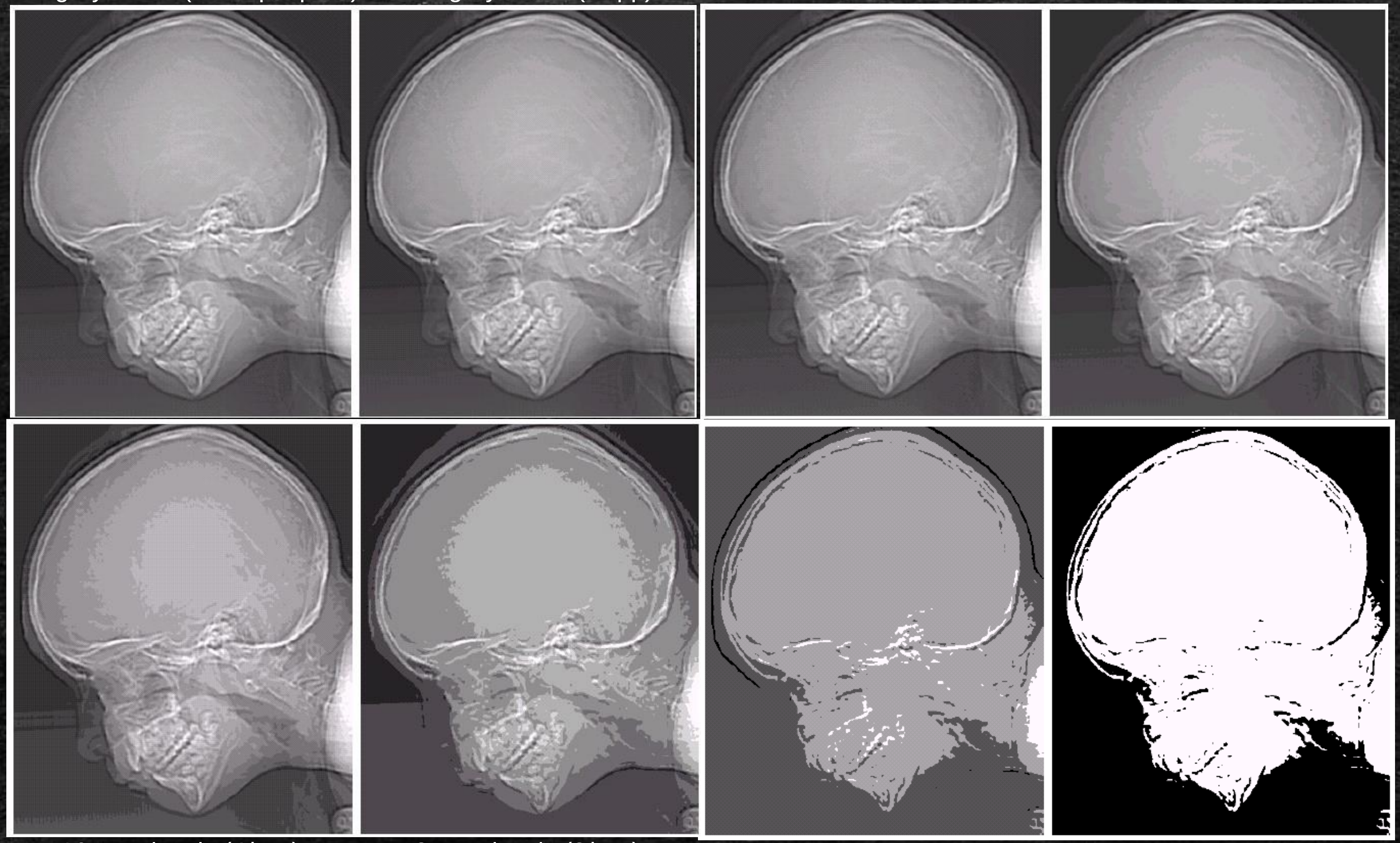


Intensity Level Resolution

- ***Intensity level resolution:*** number of intensity levels used to represent the image
  - The more intensity levels used, the finer the level of detail discernable in an image
  - Intensity level resolution usually given in terms of number of bits used to store each intensity level

Number of Bits	**Number of Intensity Levels**	**Examples**
1	2	0, 1
2	4	00, 01, 10, 11
4	16	0000, 0101, 1111
8	256	00110011, 01010101
16	65,536	1010101010101010

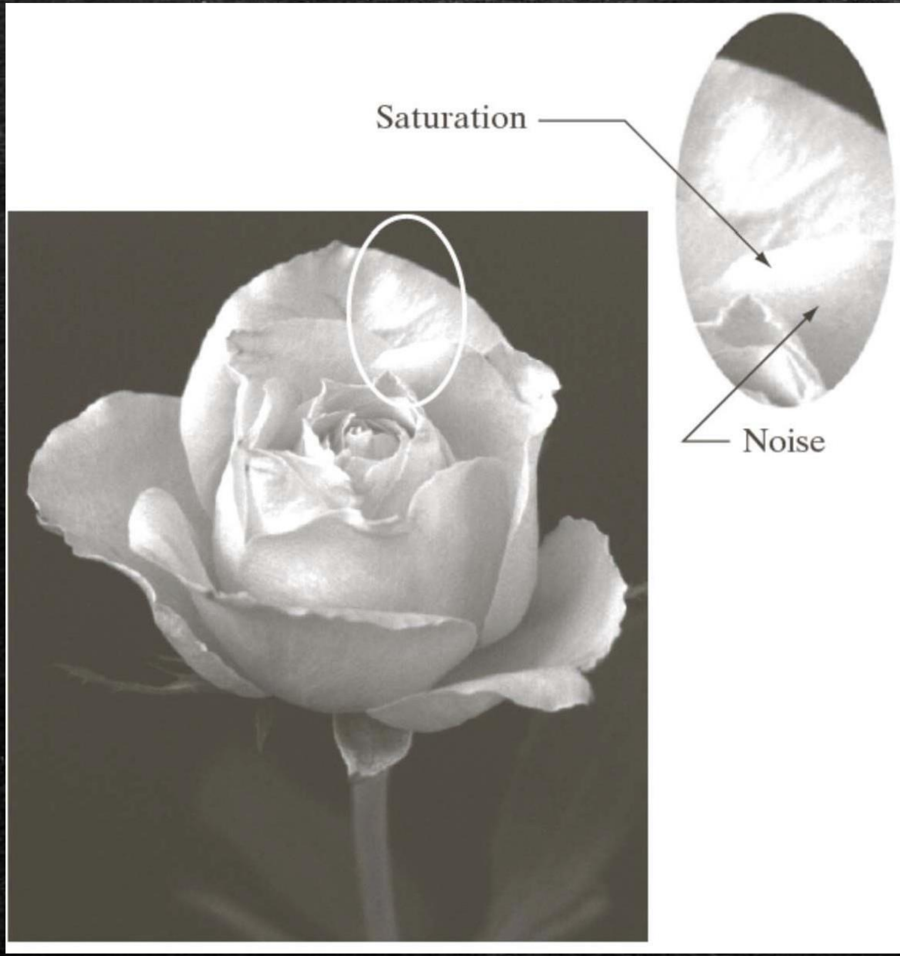
Intensity Level Resolution



256 grey levels (8 bits per pixel) 128 grey levels (7 bpp) 64 grey levels (6 bpp) 32 grey levels (5 bpp)

16 grey levels (4 bpp) 8 grey levels (3 bpp) 4 grey levels (2 bpp) 2 grey levels (1 bpp)

Saturation & Noise



Saturation: highest intensity value above which color is washed out

Noise: grainy texture pattern

Resolution: How Much Is Enough?

- The big question with resolution is always *how much is enough*?
  - Depends on what is in the image *(details)* and what you would like to do with it *(applications)*
  - Key questions:
    - Does image look aesthetically pleasing?
    - Can you see what you need to see in image?

Resolution: How Much Is Enough?

- **Example:** Picture on right okay for counting number of cars, but not for reading the number plate

Intensity Level Resolution

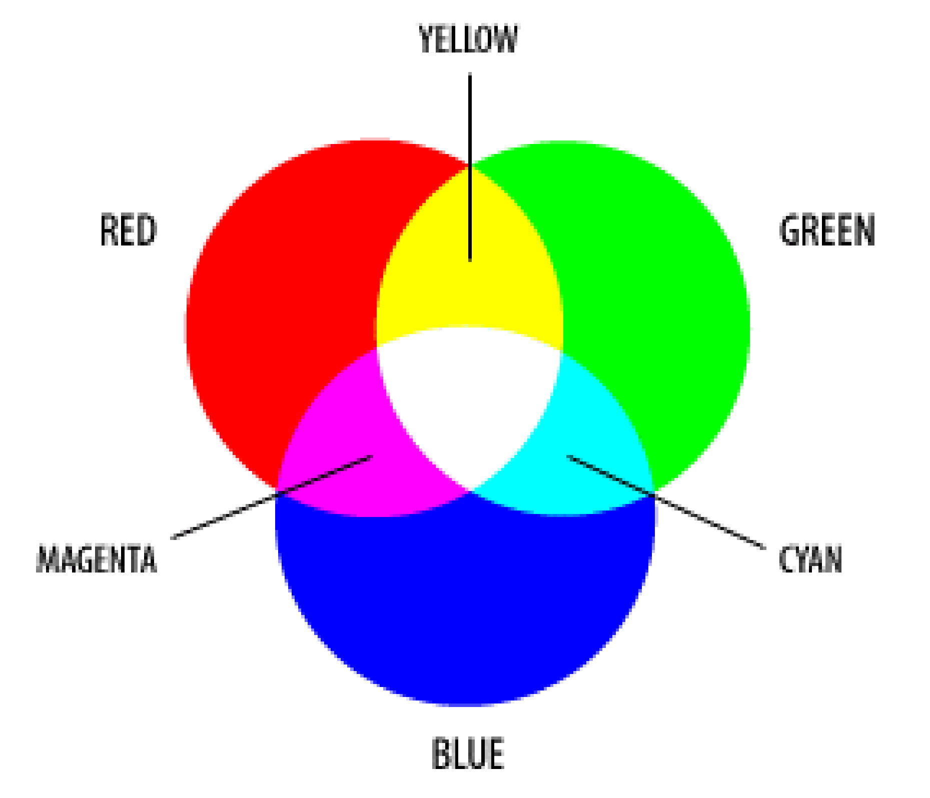
Low Detail

Medium Detail

High Detail

Colour Spaces

RGB (Red, Green, Blue)



Red, green, and blue are the primary stimuli for human colour perception and are the primary additive colours. The secondary colours produced by the addition of the common parts are Cyan, Magenta and Yellow.

49	55	56	57	52	53
58	60	60	58	55	57
58	58	54	53	55	56
83	78	72	69	68	69
88	91	91	84	83	82
69	76	83	78	76	75
61	69	73	78	76	76

Red

64	76	82	79	78	78
93	93	91	91	86	86
88	82	88	90	88	89
125	119	113	108	111	110
137	136	132	128	126	120
105	108	114	114	118	113
96	103	112	108	111	107

Green

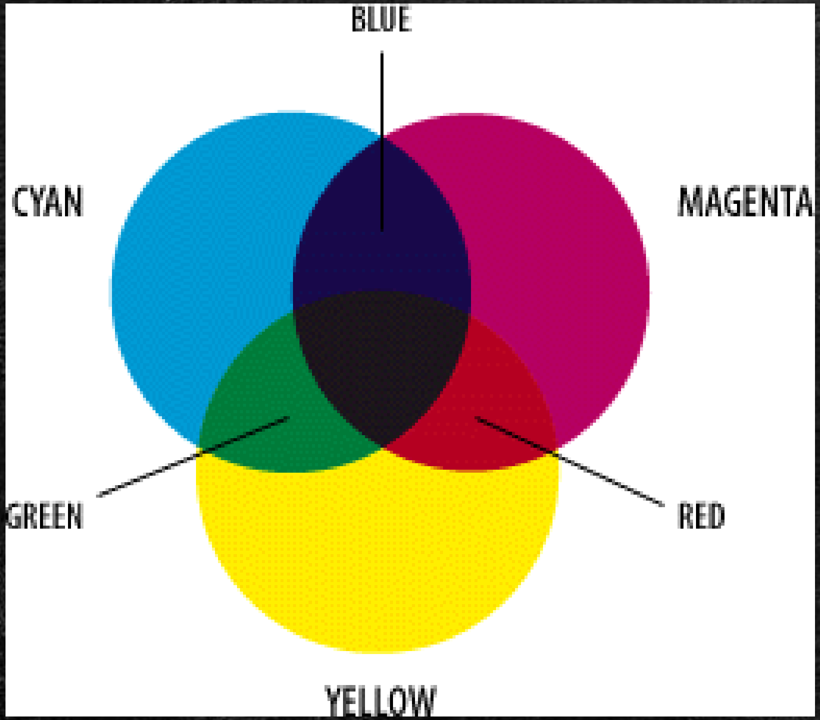
66	80	77	80	87	77
81	93	96	99	86	85
83	83	91	94	92	88
135	128	126	112	107	106
141	129	129	117	115	101
95	99	109	108	112	109
84	93	107	101	105	102

Blue

RGB

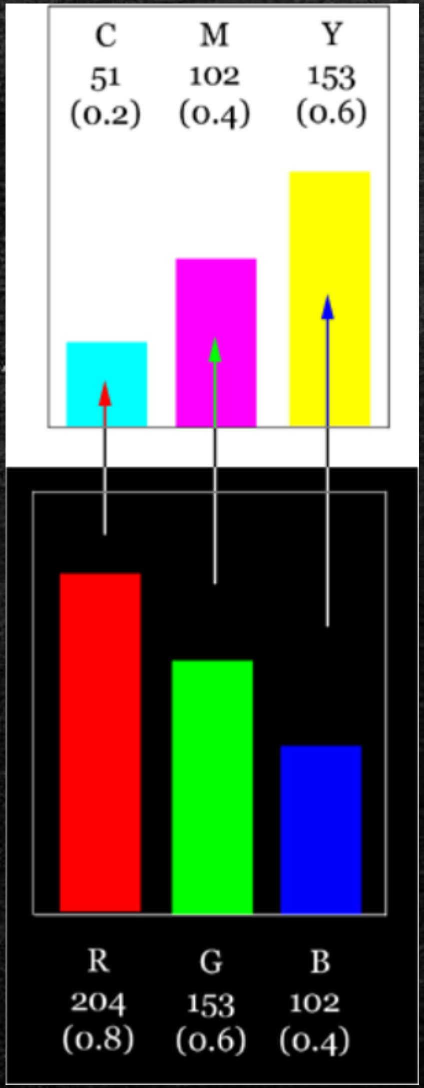
- **Additive Colours** are created by mixing spectral light in varying combinations.
- The most common examples of this are television screens and computer monitors, which produce coloured pixels by firing red, green, and blue electron guns at phosphors on the television or monitor screen.
- The total of maximum red, maximum green and maximum blue yields a white colour , whereas the absence of the three colours produces an absence of light i.e. black.
- Note that this colour space combines luminance with colour.
- The importance of RGB as a colour model is that it relates very closely to the way we perceive colour with the r g b receptors in our retinas.
- RGB is the basic colour model used in television or any other medium that projects the colour. It is the basic colour model on computers and is used for Web graphics, but it cannot be used for print production.

CMY (Cyan , Magenta, Yellow)



In CMY space, the same RGB colours are considered as subtractive mixtures of varying quantities of cyan (C), magenta (M) and yellow (Y) colourants. The resulting cubic space is identical to RGB space, apart from the fact that the origin of the C, M and Y axes is at the point representing white instead of black

CMY



- *To convert from CMY to RGB*

  C = 255 - R (or 1 - r)
  M = 255 - G (or 1 - g)
  Y = 255 - B (or 1 - b)

- *Note that C,M and Y behave as ideal subtractive colourants complementary to the particular R,G and B additive primaries.*

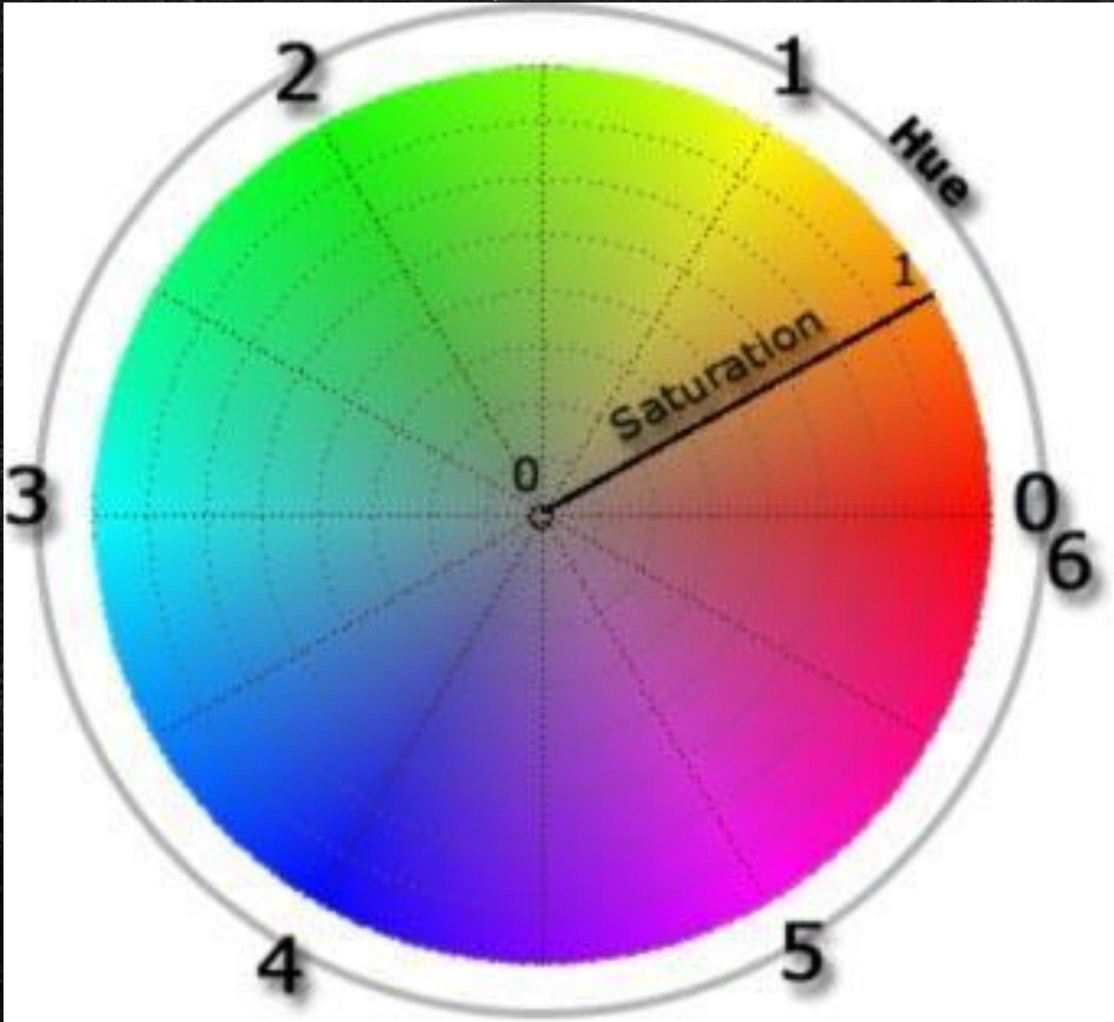
CMY

- **Subtractive Colours** are seen when pigments in an object absorb certain wavelengths of white light while reflecting the rest.
- Any coloured object, whether natural or man-made, absorbs some wavelengths of light and reflects or transmits others; the wavelengths left in the reflected/transmitted light make up the colour we see.
- It is used mostly for print.

HSL (Hue, Saturation, Luminance)

HSL stands for Hue, Saturation and Luminance (or brightness).
The HSL colour space defines colours more naturally: Hue specifies the base colour, the other two values then let you specify the saturation of that colour and how bright the colour should be.

HSL

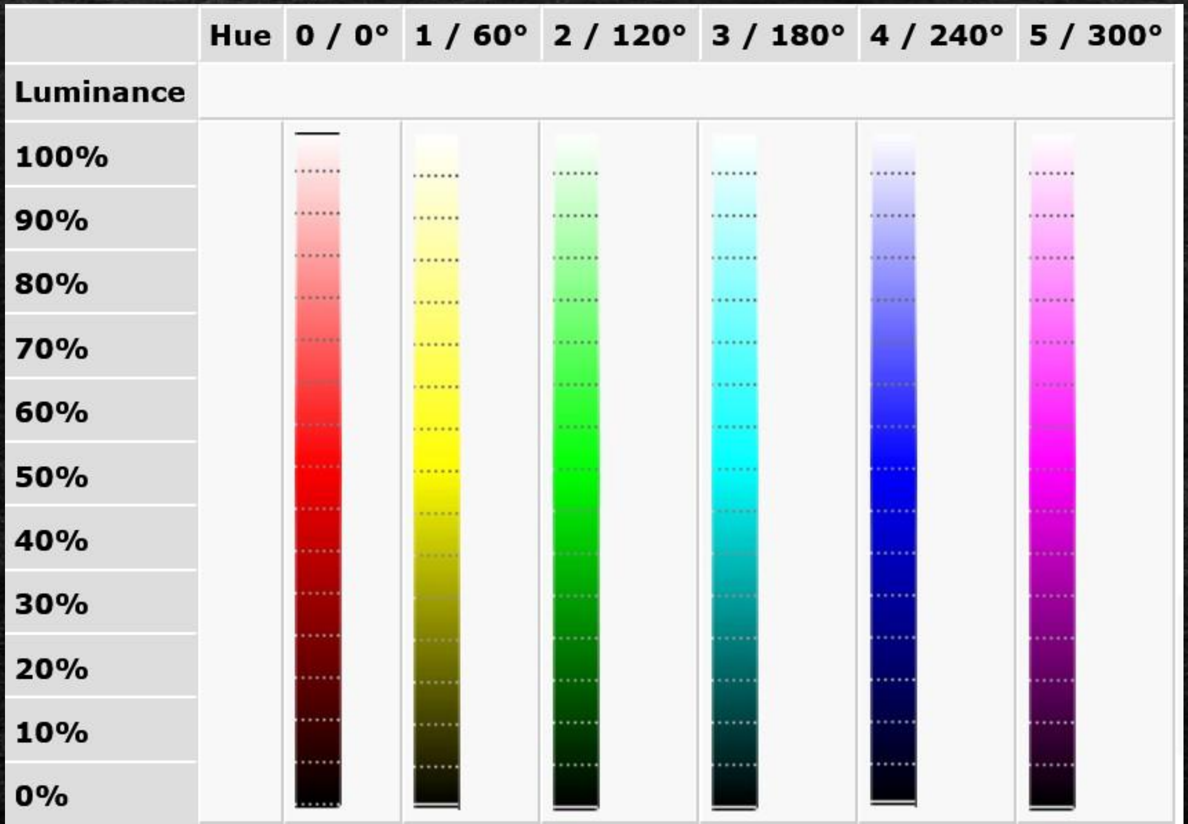


Hue specifies the colour
Hue is normally specified as degrees ranging from 0° to 360° or as numbers from 0 to 6
Saturation is a measure between 0 (less saturated i.e. grey) and 100% (more saturated i.e. colourful)

Hue	Hue (degree)	Color
0	0°	red
1	60°	yellow
2	120°	green
3	180°	cyan
4	240°	blue
5	300°	magenta
6	360°	red

HSL

- The third parameter (luminance) lets you specify how "bright" the color should be.
- 0% means, the brightness is 0, and the colour is is black.
- 100% means maximum brightness, and the colour is white.



HSL

- Unlike RGB, HSL separates *luminance*, from colour
- When doing histogram equalization of a colour image, it would be ideal to apply it on the intensity component only.
- In computer vision we often want to separate colour components from intensity for various reasons, such as robustness to lighting changes, or removing shadows.

YUV

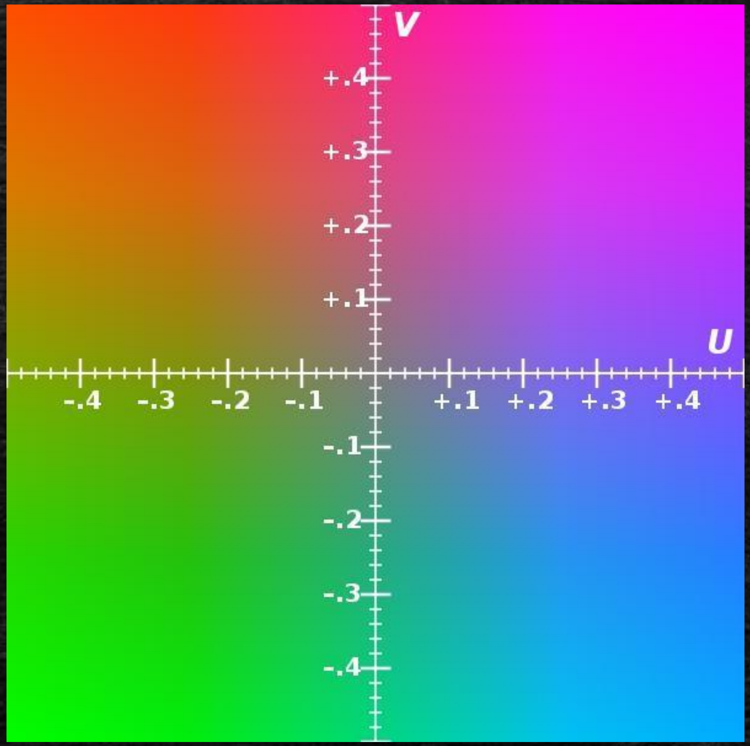
YUV *is a colour space typically used as part of a colour image pipeline. It encodes a* [*colour image*] *or video taking* [*human perception*] *into account, allowing reduced* [*bandwidth*] *for* [*chrominance*] *components. This enables transmission errors or* [*compression artefacts*] *to be more efficiently masked by the human perception than using a "direct" RGB-representation.*

YUV (Chrominance, Luminance)

- YUV is built up out of 3 components:
  Y – luminance (brightness)
  U and V – chrominance (color)

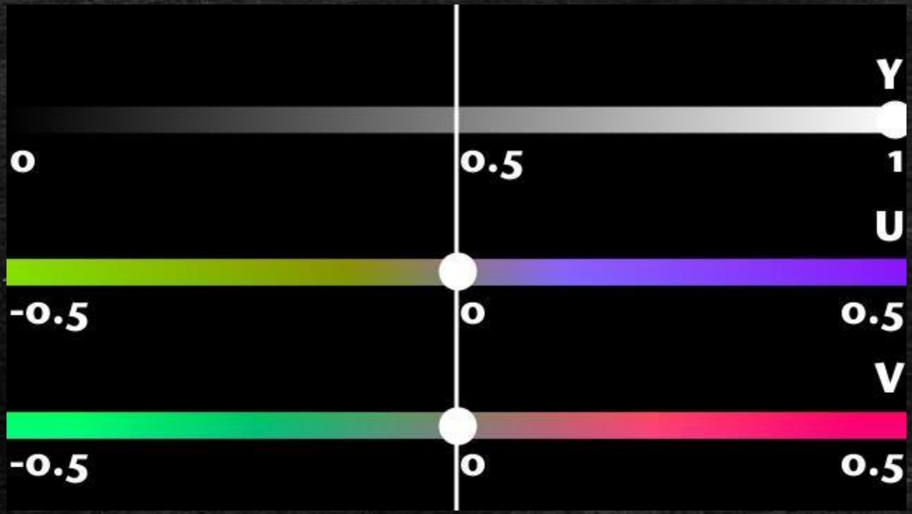
- The Y component indicates how bright the picture is.
  Y = 0 – black
  Y = 0.5 – grey
  Y = 1 – white

- U and V are a matrix of colors as seen on the right:



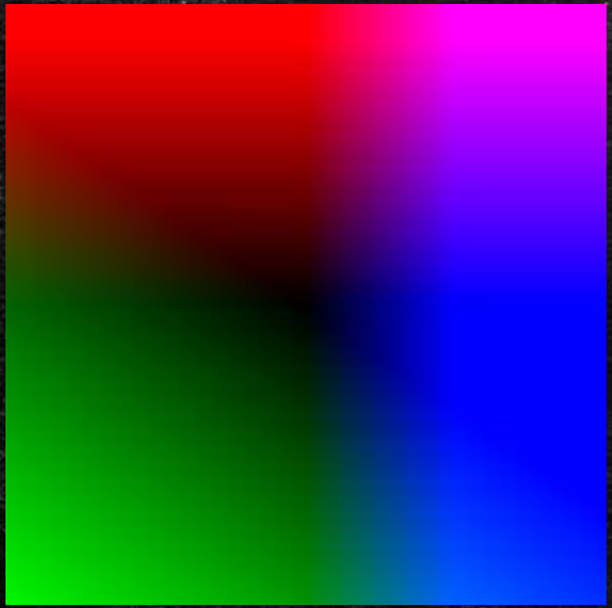
YUV

- When working in grey-scale, U and V are in the middle (U = 0 and V = 0)
- Y would provide you with all the shades between black and white (Y = 0 to Y = 1)

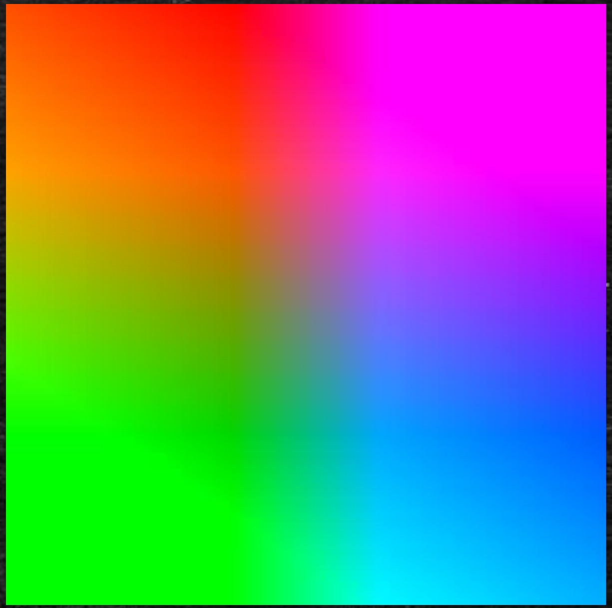


YUV

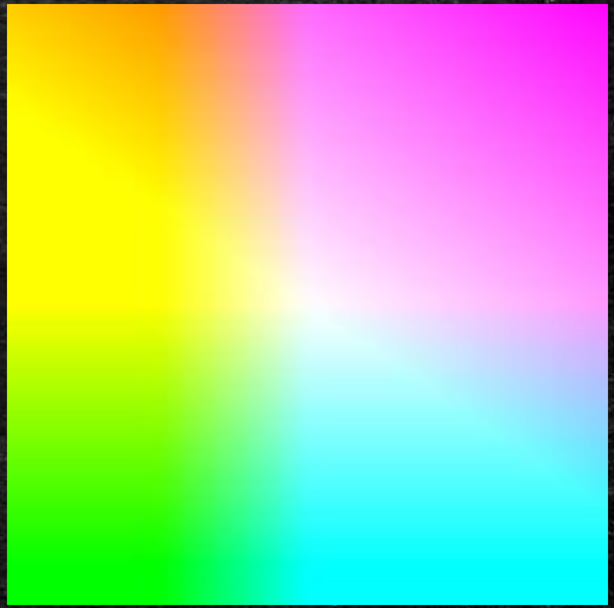
- The U-V matrix colors change depending on how bright the Y value is. The higher the Y value, the brighter the colors



Y = 0



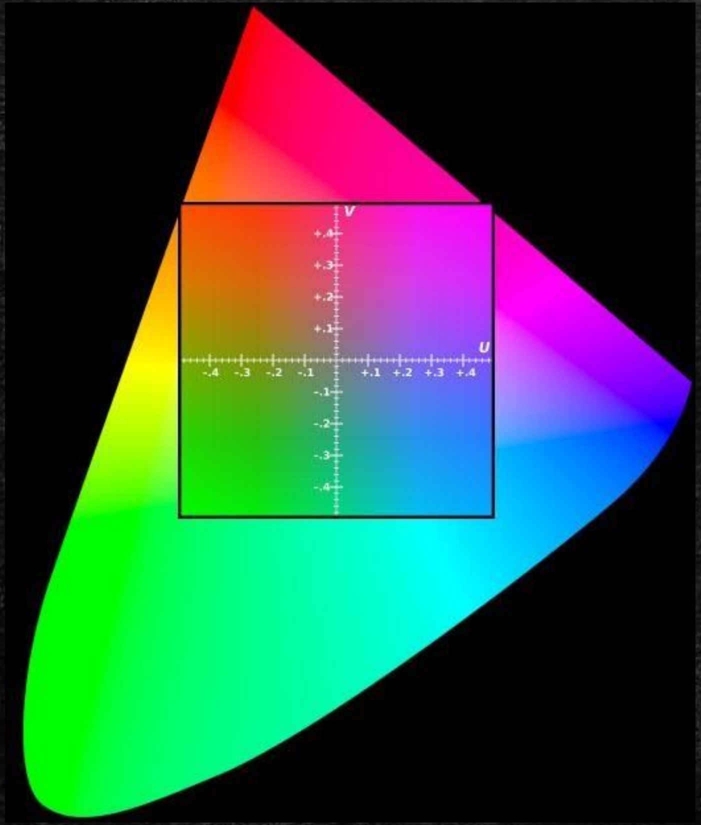
Y = 0.5



Y = 1

YUV

- YUV color space **cannot** represent all colours available in the RGB color space.



An (NTSC) approximation to convert RGB to YUV gives:

$Y = 0.299 \times R + 0.587 \times G + 0.114 \times B$
$U = -0.147 \times R - 0.289 \times G + 0.436 \times B$
$V = 0.615 \times R - 0.515 \times G - 0.100 \times B$

This is in line with the sensitivity of the human eye for these different colours

YUV

- image/video codecs use YUV in favour of compression while retaining good image quality.
- Like HSL , YUV separates the colour channels from the luminosity hence it makes it an ideal format when manipulation is required on the luminosity.
- Chrominance (UV) is kept to a minimum
- Luminance (Y) is kept as high as possible

Some Applications of CV based on Colour

Skin Detection

Some Applications of CV based on Colour

Red Eye Detection and Removal

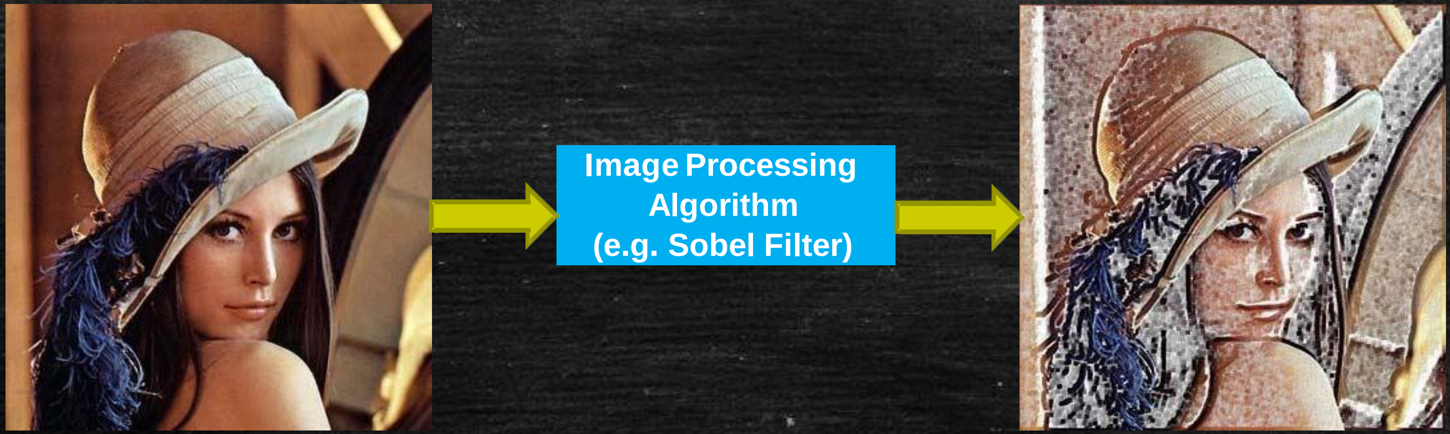
A pair of red eyes.

The corrected eyes.

Image Processing

Image Processing: Definition

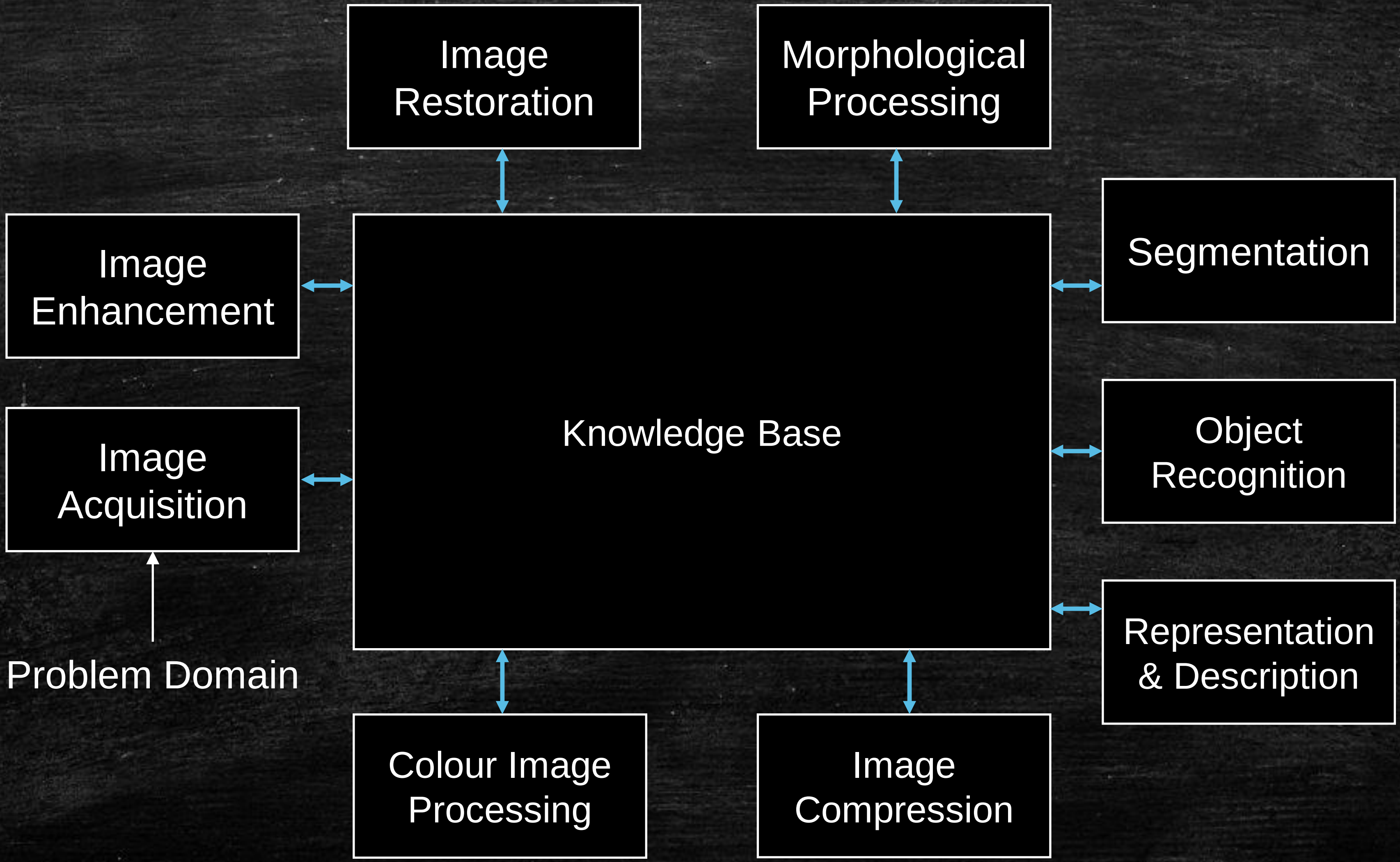
- Algorithms that alter an input image to create new image
- Input is image, output is image



Original Image *Processed Image*

- Improves an image for human interpretation in ways including:
  - Image display and printing
  - Image editting
  - Image enhancement
  - Image compression
- Some Examples follow....

Key Stages in Digital Image Processing

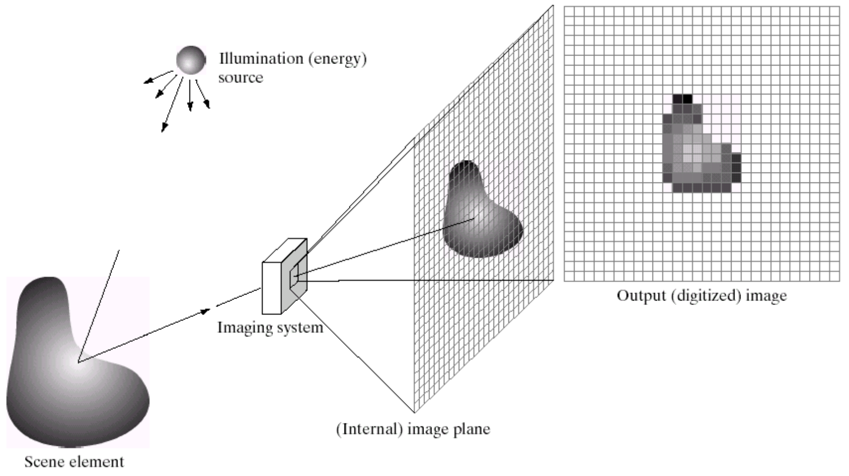


Key Stages in Digital Image Processing: Image Aquisition

Image Restoration

Morphological Processing

Image Enhancement



Segmentation

Image Acquisition

Object Recognition

Example: Take a picture

Problem Domain

Representation & Description

Colour Image Processing

Image Compression

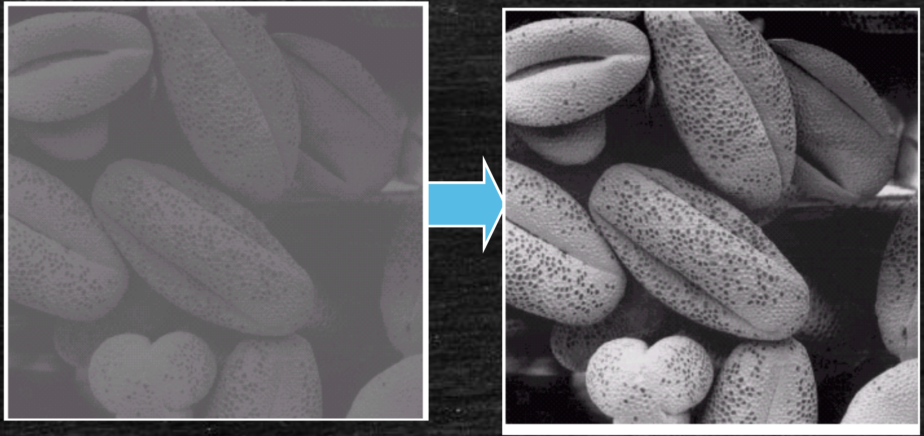
Images taken from Gonzalez & Woods, Digital Image Processing (2002)

Key Stages in Digital Image Processing: Image Enhancement

Image Restoration

Morphological Processing

Image Enhancement



Segmentation

Image Acquisition

Object Recognition

Example: Change contrast

Problem Domain

Representation & Description

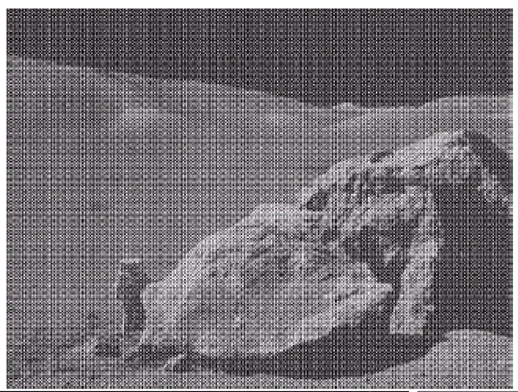
Colour Image Processing

Image Compression

Key Stages in Digital Image Processing: Image Restoration

Image Restoration

Morphological Processing



Example: Remove Noise

Image Enhancement

Segmentation

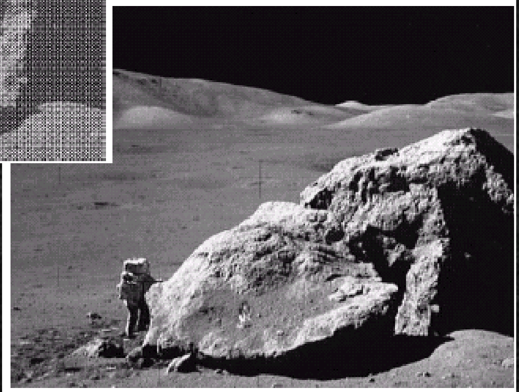


Image Acquisition

Object Recognition

Problem Domain

Representation & Description

Colour Image Processing

Image Compression

Key Stages in Digital Image Processing: Morphological Processing

Extract attributes useful for describing image

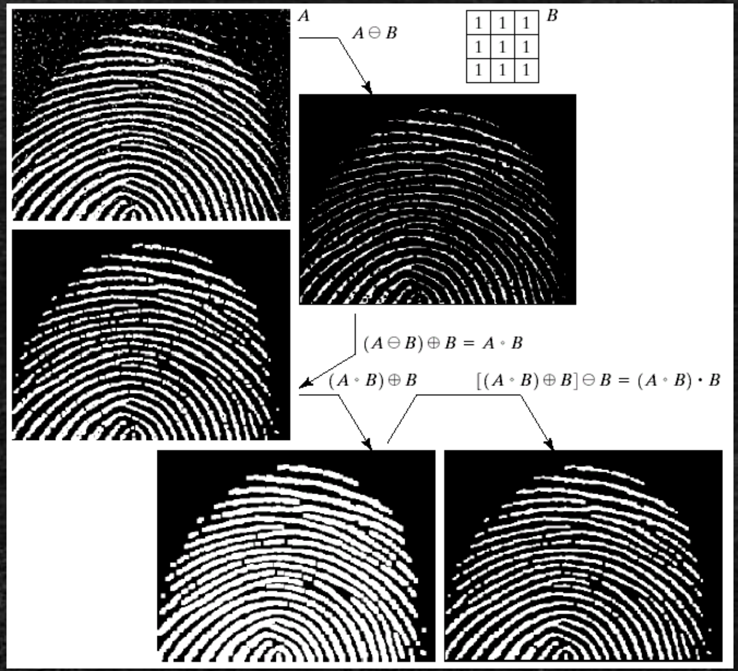
Image Restoration

Morphological Processing

Image Enhancement

Image Acquisition

Problem Domain



Segmentation

Object Recognition

Representation & Description

Colour Image Processing

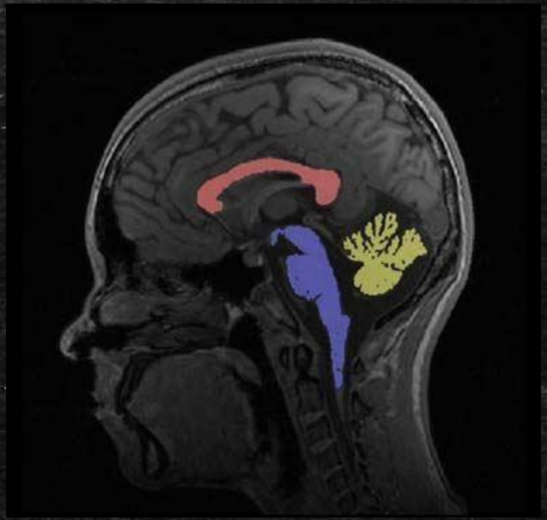
Image Compression

Key Stages in Digital Image Processing: Segmentation

Image Restoration

Morphological Processing

Image Enhancement



Segmentation

Divide image into constituent parts

Image Acquisition

Object Recognition

Problem Domain

Representation & Description

Colour Image Processing

Image Compression

Key Stages in Digital Image Processing: Object Recognition

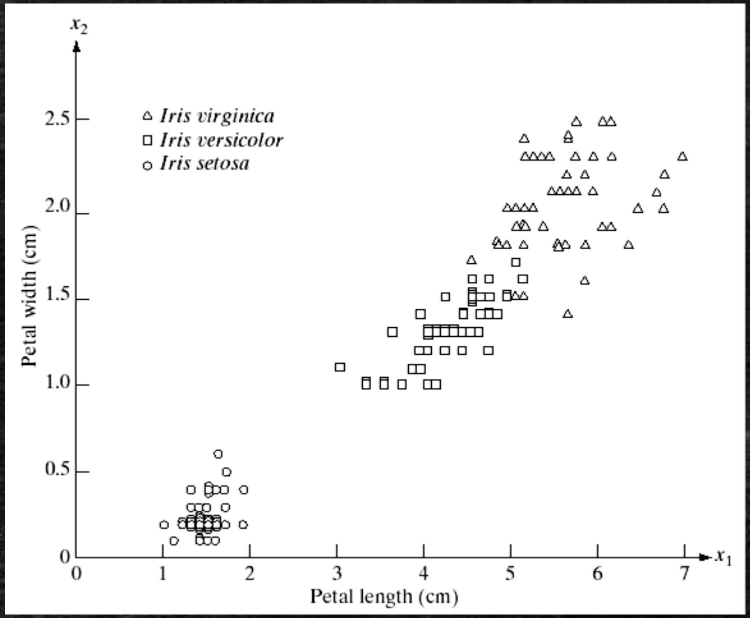
Image Restoration

Morphological Processing

Image Enhancement

Image Acquisition

Problem Domain



Segmentation

Object Recognition

Finds & Labels objects in scene (e.g. motorbike)

Representation & Description

Colour Image Processing

Image Compression

Key Stages in Digital Image Processing: Representation & Description

Image Restoration

Morphological Processing

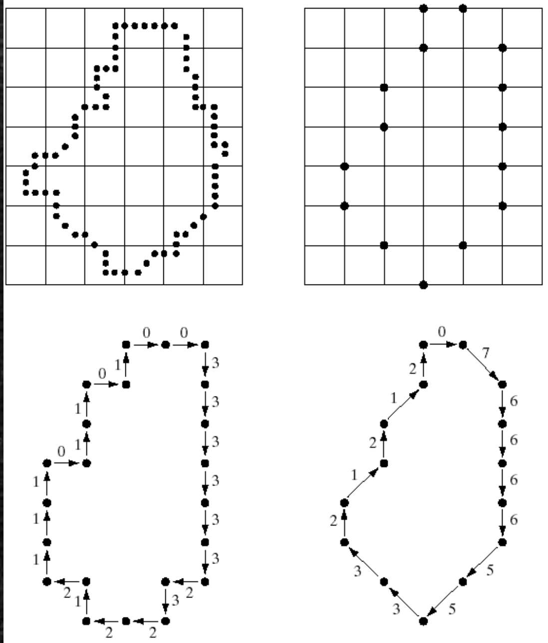
Image Enhancement

Segmentation

Image Acquisition

Object Recognition

Problem Domain



Representation & Description

Image regions transformed suitable for computer processing

Colour Image Processing

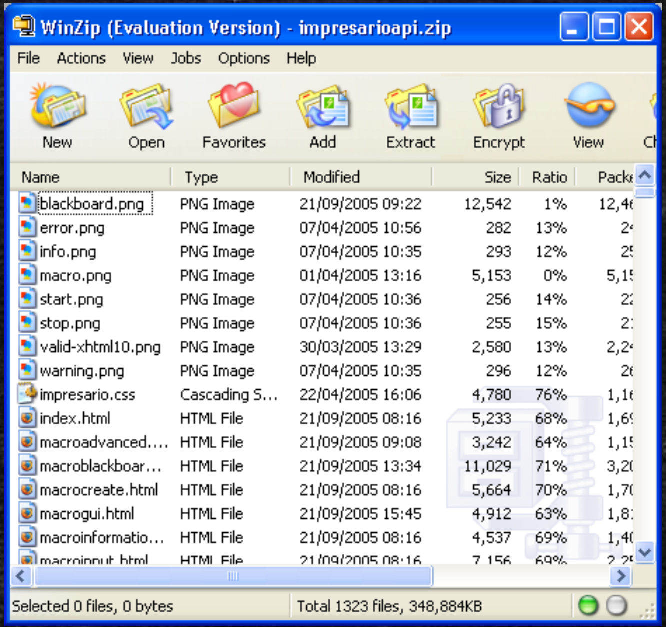
Image Compression

Key Stages in Digital Image Processing: Image Compression

Image Restoration

Morphological Processing

Image Enhancement



Segmentation

Image Acquisition

Object Recognition

Problem Domain

Representation & Description

Colour Image Processing

Image Compression

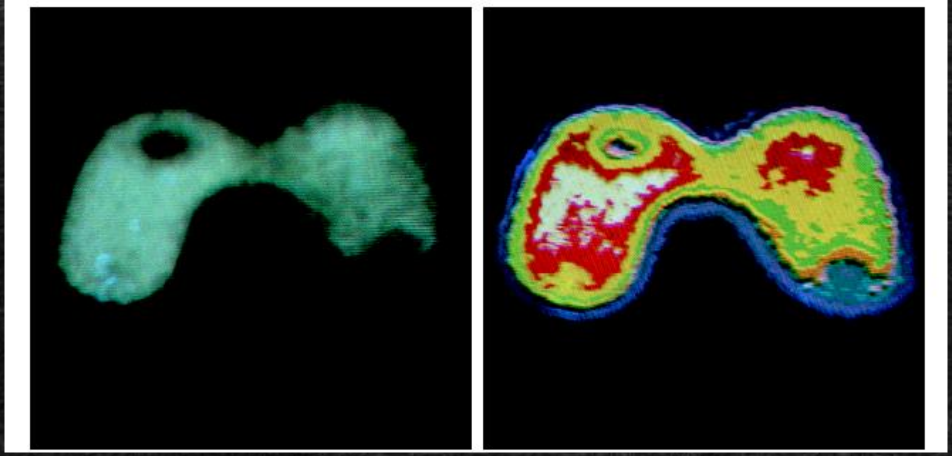
Reduce image size (e.g. JPEG)

Key Stages in Digital Image Processing: Colour Image Processing

Image Restoration

Morphological Processing

Image Enhancement



Segmentation

Image Acquisition

Object Recognition

Problem Domain

Representation & Description

Consider color images (color models, etc)

Colour Image Processing

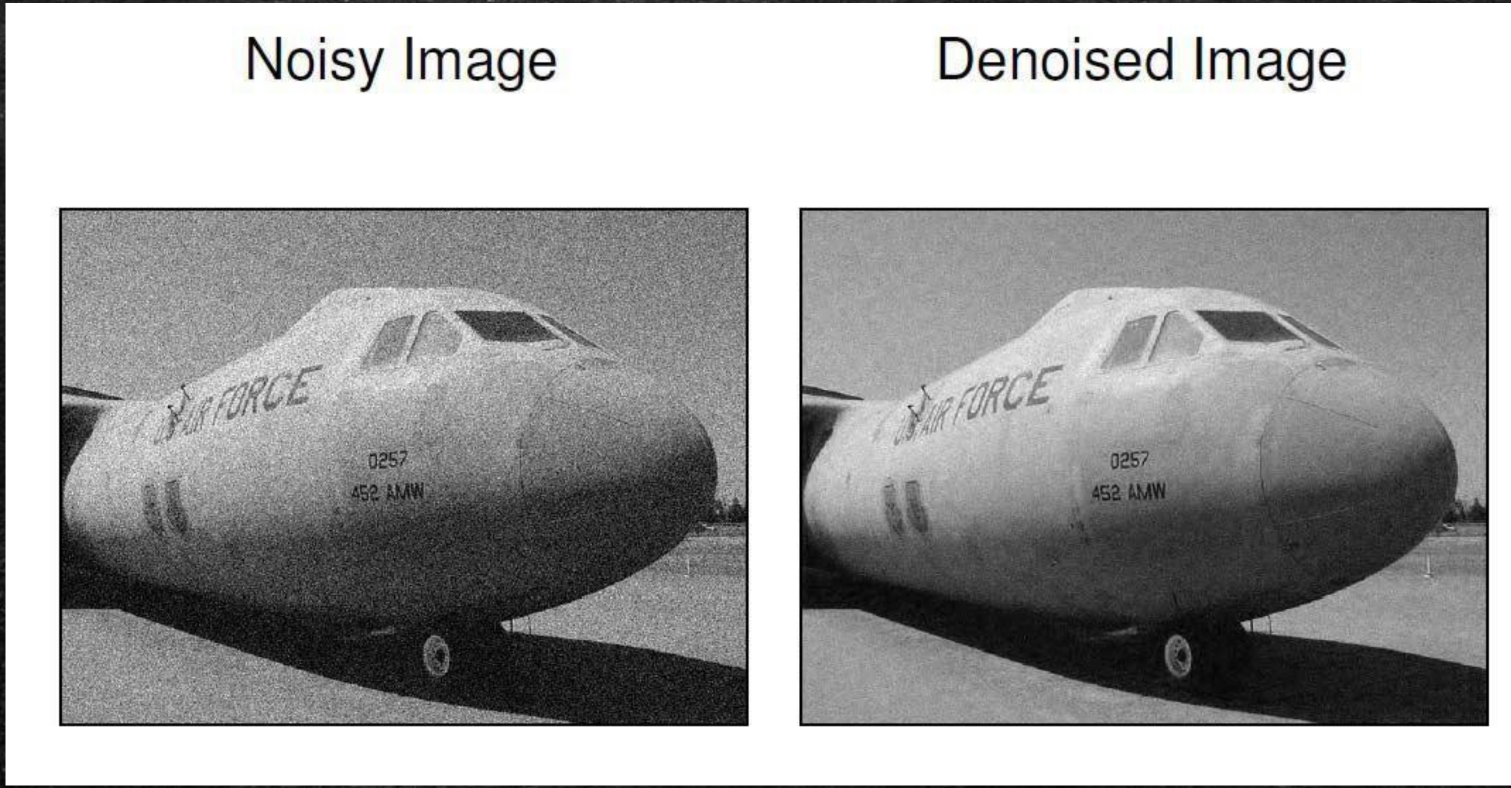
Image Compression

Loading and Displaying Colour Images

Color images are loaded as a $h * w * 3$ array. Following code loads an image called "fruits.jpg". Then it zeros-out the red plane and the green plane. As a result the image becomes blue.

```
close all
im = imread ('fruits.jpg');
imshow( im)
pause
im ( : , : , 1 ) = 0;
im ( : , : , 2 ) = 0;
figure
imshow( im)
```

Image Processing: Noise Removal



Think of noise as white specks on a picture (random or non-random)

Image Processing: Contrast Adjustment

Low Contrast   Original Contrast   High Contrast

Image Processing: Blurring

Original Image Blurred Image

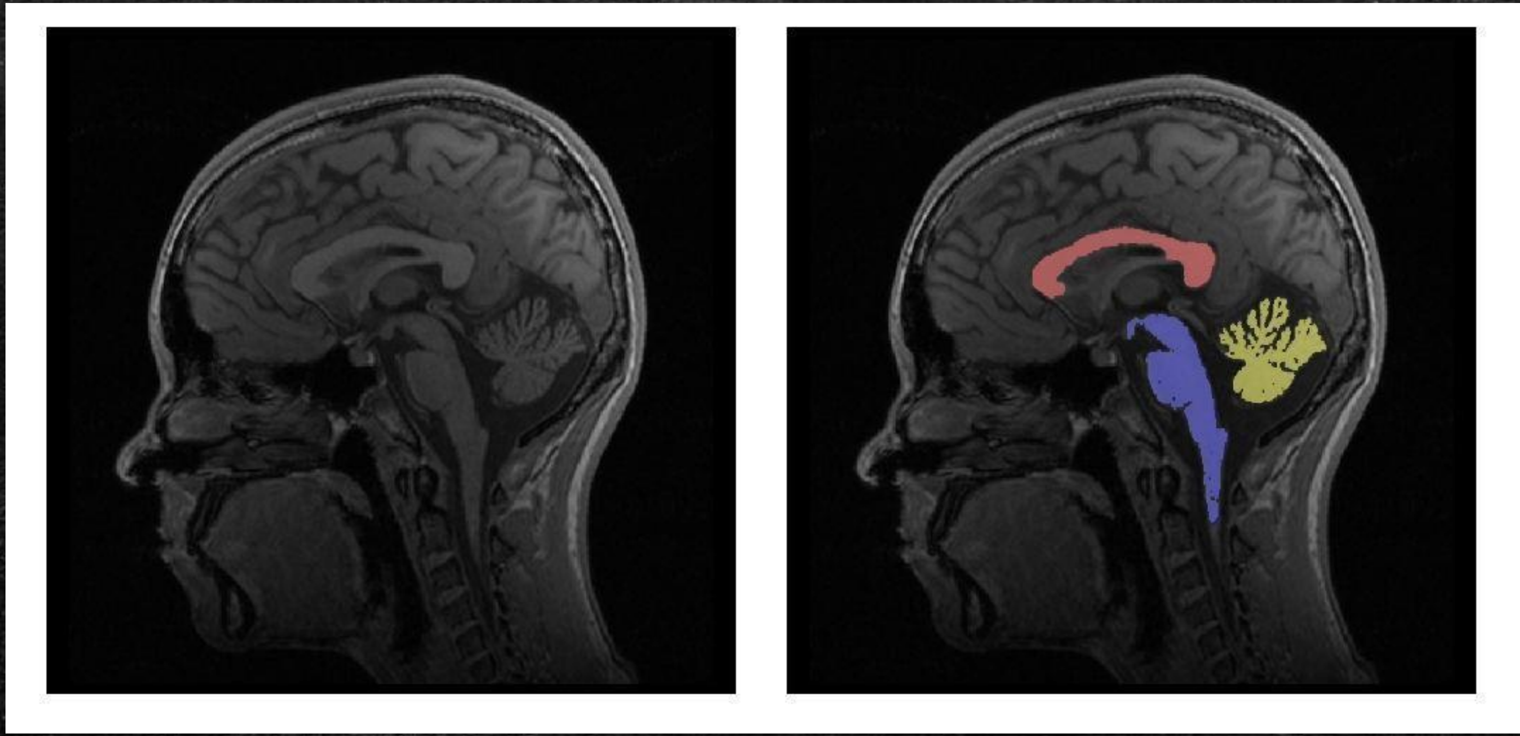
Image Processing: Compression

Original, 2.1MB

JPEG Compression, 308KB (15%)

Image Analysis: Edge Detection

Segmentation



Relating IP, IA, CV