Flow Measurement
via
Digital image analysis

- Working principle of a digital camera
- Digital image processing
- LDV, LDA (continued)

Fu-Ling Yang
May, 09, 2012

Working principle of a digital camera

-- array of optical sensors that converts light intensity into electrical signals

Scientific Digital Camera

- Digital image sensor
  - Converts an optical image (light intensity) into an electric signal by CCD or CMOS sensors.
    - CCD: Charge-Coupled Device
    - CMOS: Complementary-Metal-Oxide-Semiconductor detector
- Digital camera
  - Built-in digital image processing chip to covert the raw data from the image sensor into a color-corrected image in a standard image file format.

Common types: CMOS camera, CCD camera, EMCCD (Electron multiplying CCD) camera; ICCD (Image intensified CCD) camera.

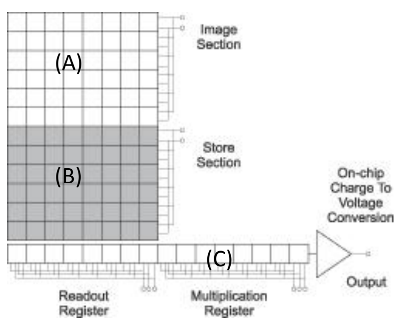
CCD sensor

Charge-coupled device has three major components:

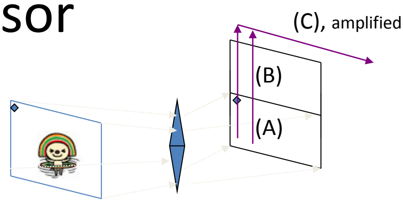
(A) A **silicon diode photo-sensor** (a Pixel) that receives photons of various intensity. If the incident photons have sufficient energy to agitate an electron motion away from the silicon layer, which generates a charge.

(B) The charge moves to a down-stream **charge storage region**, generating an analogous signal.

(C) The quantity of the accumulated charge, the sum of 0 or 1 (depending on the incident light intensity), is then amplified and transmitted through a clock signal.



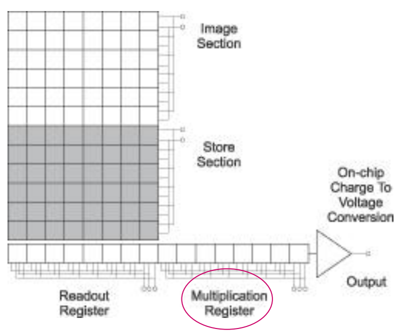
CCD sensor



- An image is projected by a lense on the diode photo-sensor, causing each pixel to accumulate an electric charge proportional to the light intensity at that location.
- Once the array has been exposed to the image, a control circuit causes each pixel to transfer its charges to its neighbor. The last capacitor in the storage section sends its charge into a charge amplifier, which converts the charge into a voltage.
- By repeating this process (scanning across the photo-sensor), the control signal converts the charge information of the entire pixel array (in (A)) to a sequence of voltages (output from (C)), which it samples, digitizes and stores in some memory format.
- The algorithms for voltage conversion results in digital images/videos of different formats. These stored images can then be transferred to a printer, digital storage device or video display.

EMCCD

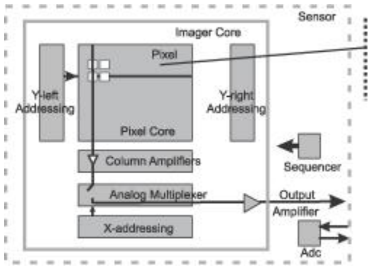
In a CCD, there is typically only one amplifier at the corner of the entire array; the stored charge is sequentially transferred through the parallel registers to a linear serial register, then to an output node adjacent to the read-out amplifier.



EMCCDS use similar structures to CCD's, but prior to being readout at the output node the charge is shifted through an additional register— the multiplication register— in which the charge is amplified.

A signal can therefore be amplified above the readout noise of the amplifier and hence an EMCCD can have a higher sensitivity than a CCD.

CMOS sensor

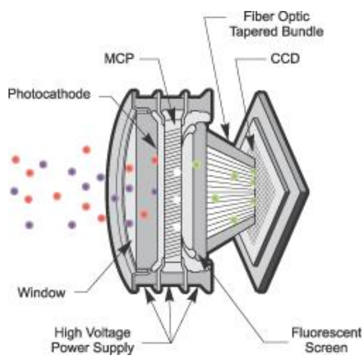


- Each column of photo-sensors has an amplifier associated with it (image can be transmitted above the noise).

- A row of pixels can be readout in parallel with the row selected by an addressing register (Y-addressing) or an individual pixel can be selected by column multiplexer (X-addressing).

- A CMOS device is essentially a parallel readout device and therefore can achieve higher readout speeds. However, compensating for the variations in the current state of the art CMOS devices is difficult.

ICCD sensor



- The photocathode is similar to the photo-sensitive region of a Photo-Multiplier Tubes (PMTs) widely used in confocal microscopes and spectrometers: the incident photons strike out electrons with the impact energy..

- The liberated electrons are then accelerated toward an electron multiplier composed of a series of angled tubes known as the Micro-Channel Plate (MCP).

- Under the accelerating potential of a high voltage, the incident electrons gain sufficient energy to knock off additional electrons and hence amplifies the original signal.

- This signal is then detected/amplified/transmitted using the techniques developed in other CCD sensors.

Remarks on the performance

- One weakness of a CCD stems from the fact that the CCD is essentially a serial readout device and low noise performance is only achieved at the expense of slow readout speeds. Thus a short exposure time is the major bottle-neck for the CCD technique.

- CMOS cameras can achieve high frame rates with moderate sensitivity. The precisions, however, is comparatively low due to the non-linear modulation of signal at transmission.

- Due to the acceleration section, Intensified CCD Cameras can achieve ultra short exposure times. But the MCPs make the unit expensive, heavy, and bulky.

- Since the electron transmission is sensitive to temperature, overheating may degrade the performance of a high-speed CCD image acquisition facility. Extra expenses is paid for integrated cooling system.

Color image sensor

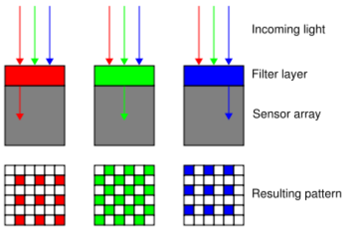
Differed by the means of the color separation mechanisms:

- Bayer sensor: use Bayer filter to distribute R,G,B light to a specific pixels. The image is reconstructed using a specific algorithm.

- Foveon X3 sensor: layered sensor at one pixel that responds to R,G,B light differently.

- 3CCD: the light is first separated by a dichroic prism, which generates separate R,G, B incident rays one separate sensors at one pixel. (Best quality, but most expensive)

Bayer filter (GRGB)



- A color filter array atop of a square grid of CCD sensors that redistribute R,G,B lights in a specific arrangements:

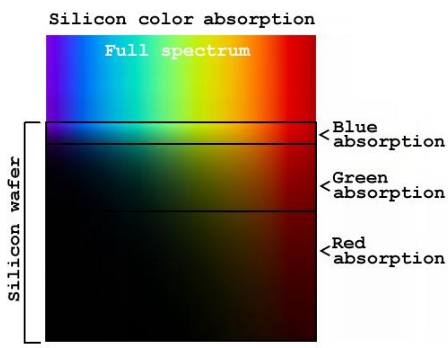
  To mimic human eyes that are more sensitive to green lights, the filtered image is 50% green, 25% red, and 25% blue, resulting a 'Bayer pattern image'.

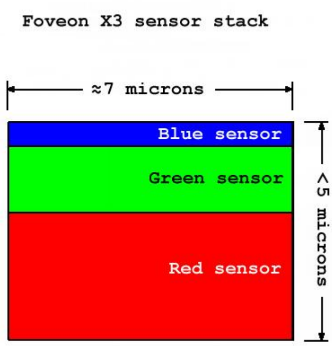
- Each pixel of the Bayer pattern image only carries information of 'ONE' color. Thus, some color data is missing and has to be reconstructed with some mathematical algorithms (artificial compensation).

- To interpolate a complete image from a partial raw data, Demosaicing (in contrary to a Mosaic pattern) algorithm can be the mean of nearest neighborhood, linear interpolation, …, etc.

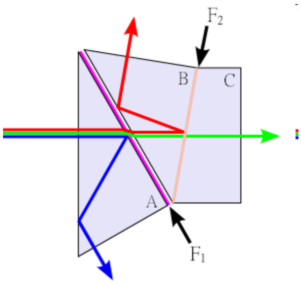
Foveon vertical color filter

- Atop each pixel, there are 3 stacked active sensors with different silicon layer deposition. The R,G,B components are filtered in sequence and the 'filtered' composition of R,G,B.
- The transmission rate of each color element is based on the wavelength-dependent absorption of light of the silicon.





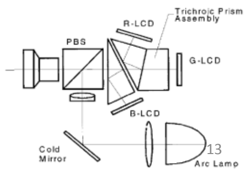
Trichroic prism assembly (TPA)



- After a light ray enters the first prism (A), the blue component is reflected by a coating (F1) at the AB interface (long wave-length bandwidth, high frequency).
- The transmitted light ray (of lower frequency) enters a second prism (B) and is split again at the BC interface, with a coating (F2) designated for the red component.

- The remaining green component of the beam travels through prism C.
- Each of R,G, B rays undergoes a total internal reflection
- The three color components are then

Note: TPA can be used in reverse to combine red, green and blue beams into a color image and is used in many projector devices.

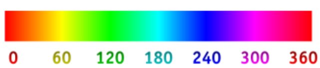


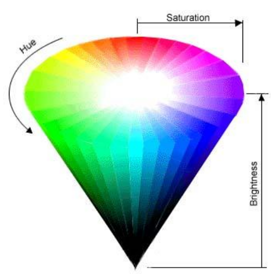
Color image formats

In addition to RGB decomposition, HSL (Hue, Saturation, Lightness) and HSV (-,-, value) presentations are also popular.

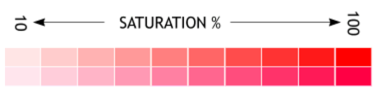
A true color can also be composed by the combination of

-- hue (色調 彩度)





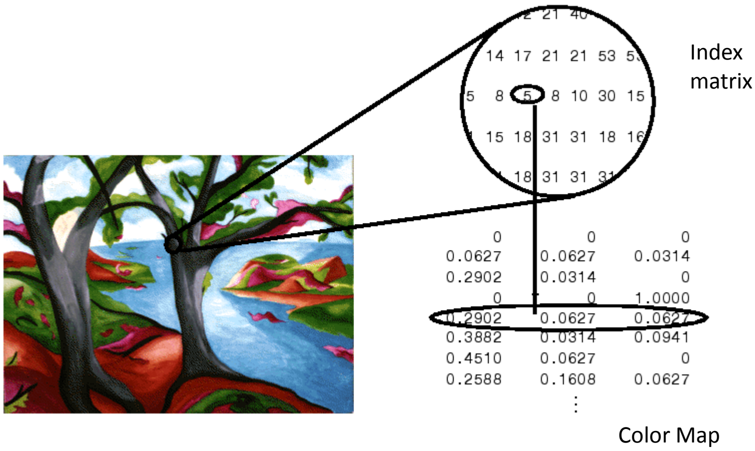
-- the saturation (飽和度)



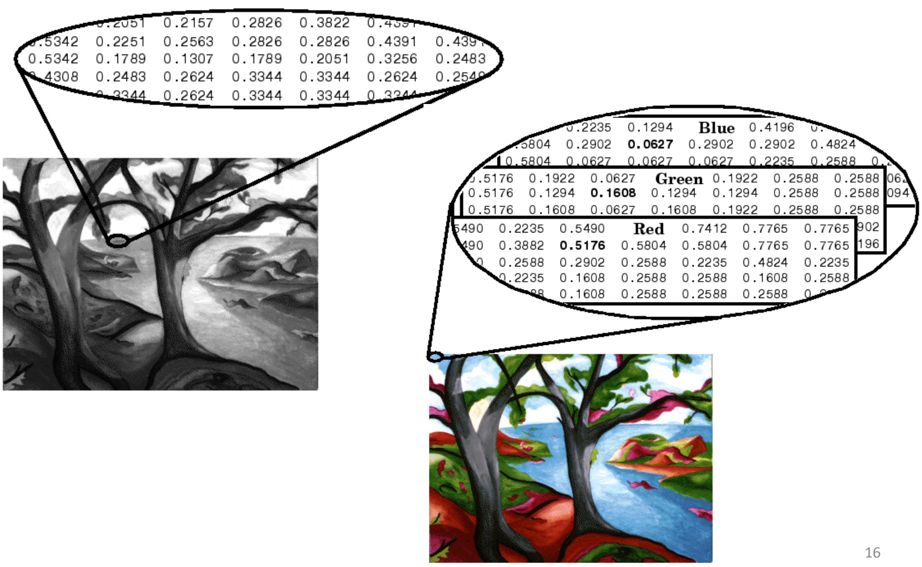
-- the lightness/value (亮度)

Note that HSL/V formats are not the raw data from a digital camera.

Structure of Indexed image
(8-/16- bit)



Intensity / True color (RGB) image



Digital image formats

Digital image formats

Two main categories:

- Raster image (JPEG, GIF, PNG, TIFF, BMP, RAW)
  - 2-dimensional
- Vector image (SVG—scalable Vector Graphics)
  - Also known as geometric modeling, object-oriented graphics that utilizes geometrical primitives (point, line, triangle, curve, polygon) to represent an image in computer graphics.
  - Thus, the chosen mathematical formulae determine where to put the constructing elements to generate an image wrt. a specific screen/image resolution.
  - The resulting image in general preserves good quality than a 2D 'flat' raster image when the images are enlarged.

Image compression

- Lossless compression:
  - reduce file size with no loss in image quality
- Lossy compression
  - Discard information that cannot be recognized by human eyes; different lossy compression algorithms utilize various levels of quality compression to reduce the image file size
  - May cause deterioration (compression artifacts)
- Dither image
  - Apply an internationally-accepted noise to the data to reduce the artificial error induced by quantization.

Raster image types

- Each image is stored as a m-by-n matrix. Each entry (pixel) stores the information of the image (in relation to light intensity) with one of the following formats:
  - **Binary**: 0/1 (black/white) image
  - **8-bit/16-bit:** 1-256 / 1- 2^{16}
  - **Grayscale:** 0-1
  - **Color:** One popular format is **RGB color** formats. Each pixel has three digits (8-/16-bits) that store the intensity of each color (R,G,B) element. Other format includes HSL, HSV.

Raster image format (1)

- JPEG (Joint Photographic Experts Groups)
  - **Lossy** format (8-bit palette; 256 colors in total); degradation after repetitive editing/saving
- BMP (Windows bitmap)
  - Lossless compression (developed for MS system, thus widely accepted)
- GIF (Graphics Interchange Format)
  - Lossless compression; 8-bit palette
  - Suitable to store images of simple color combination; widely used in animation
- Raw
  - Lossless compression but differ among devices, thus not accessible to normal software

Raster image format (2)

- TIFF (Tagged Image File Format)
  - Lossy or Lossless format (8-/16- bits per color)
  - use Lempel-Ziv-Welch (LZW) compression algorithm
  - Record device-specified color spaces, not supported by normal software
- PNG (Portable Network Graphics)
  - Lossless format is excellent for editing
  - Lossy format used when distributing images (as popular as JPEG)
  - Supports true color (16million colors) images; while the similar GIF only accepts 256 colors.
- Other formats:
  - EPS (Encapsulated PostScript); PDF; Windows Metafile

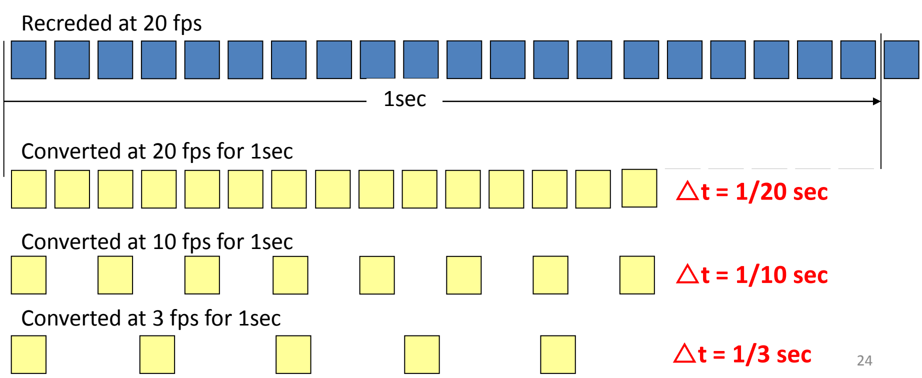
Digital video format

A digital video is generated by threading a series of digital images in time.
Differed by the algorithms for audio and video data compression:

- **MPEG (Movie Professional Expert Group)**
  - MPEG-1, MPEG-2 (for broadcast quality television), MPEG-4
- **AVI (Audio Video Interleave)**
  - especially designed for MS environment, developed upon the Resource Interchange File Format (RIFF).
  - An event's audio/video data is divided into big chunks, which is coded/decoded by a specific CODEC.
- **QuickTime**
  - Multimedia framework developed by Apple.

Analysis of a digital video…

- Convert the AVI, MPEG, ..etc videos into a sequence of digital images (grayscale or true color). Analyze the consecutive images to obtain the temporal variations of the phenomena.
  - Be careful about the conversion rate:



Fundamentals of image analysis

- Conversion between image formats
- Define the targeted image components
  - Object motion can be tracked by determining its center of mass in the consecutive image frames.
  - The liquid motion can be extrapolated by the displacement of the seeding particles between two frames
  - The volume fraction of a solid-liquid mixtures can be estimated by the total area of the solid objects…
- Develop an effective algorithm for the task by manipulating the images.
  - Software: MATLAB, ImageJ, …
- By analyzing the sequential images, the transient phenomena can be extrapolated. Thus, the knowledge of time duration between consecutive images is essential for a real transient phenomenon.

Digital image acquisition

- Employ a digital camera to record a transient or a dynamic phenomena. The obtained digital video is transformed into a sequence of digital images with a designated rate.
- Both the filming and export frame rate combine to determine the actual time duration between the two consecutive images, which information is important in extrapolating the transient information.
- The camera filming rate determines how fast the information of light intensity is exported from a CCD sensor. Thus, a strong light source is required if a high frame rate is required.

Digital image processing and analysis

Generally contains three major categories:

(1) Image conversion

change of formats, compression (to reduce the amount memory needed to store a digital image);

(2) Image enhancement and restoration

image defects caused by digitization process or by the system setup may be ‘corrected’ or ‘enhanced’ in this step to obtain an image appropriate for later measurement

(3) Measurement extraction

further manipulation may be needed to achieve the desired measurement

Image processing

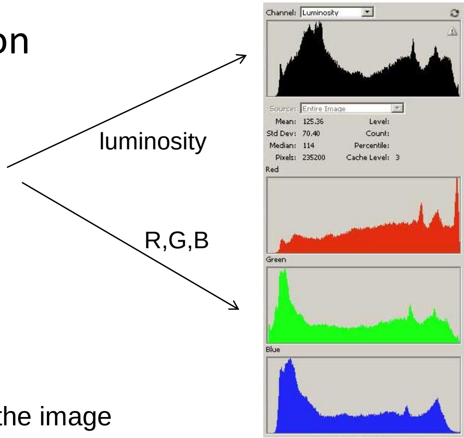
- A digital image is a commonly stored as a M x N x 3 (or x1) matrix of numbers, which value correspond to the light intensity (EM wave) of the object.
- Image processing / editing is performing mathematical operations on these numbers in order to achieve desired format or to extrapolate information from the arrays of numbers:
  - For example, the following algorithm follows NTSC standard to calculate the luminance of a RGB color image stored as P(M,N,3):

$$I(m,n)=\underset{R}{0.2989\,P(m,n,1)} + \underset{G}{\underline{0.587}\,P(m,n,2)} + \underset{B}{0.114\,P(m,n,3)}$$

Manipulation / Conversion

- Histogram: a graphical representation of tonal (grayscale or luminosity) distribution of a digital image

  By changing the histogram, image enhancement and manipulation can be achieved.

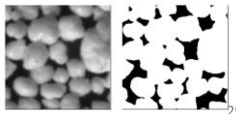


- Brightness mean luminosity of the image
- Threshold change a grayscale image into a BW (binary) image:

For each pixel

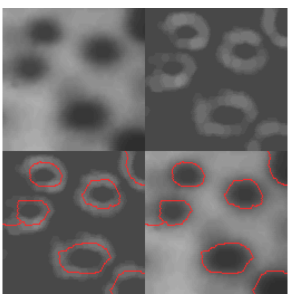
I >threshold, make it 1 (Black pixel)

I <threshold, make it 0 (white pixel)



Manipulation / data extraction

- Edge detection: on **a grayscale image** (image luminosity stored in an array of numbers between 0 and1), the image light **gradient** can be calculate (use the difference between the neighboring pixels) . Use the **local maximum** of gradient to present the edge.



Different algorithms can be developed to outline the image.

This class of approach requires

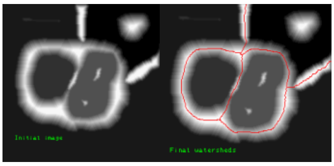
(1) Gradient calculation (mathematical description, difference equation)

(2) Criterion for the characteristic gradient of an edge

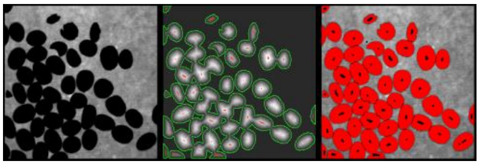
Manipulation / data extraction

- **Sharpen / Blur** algorithms using the gradient field of the original grayscale image
- **Watershed algorithm (Segmentation method)** to split the whole image into small segments

Fill from the minimum (local or global) up to a specific level to reveal the boarder lines (which can be used to regenerated the watershed).

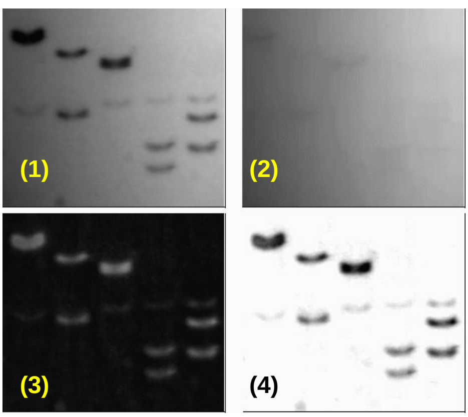


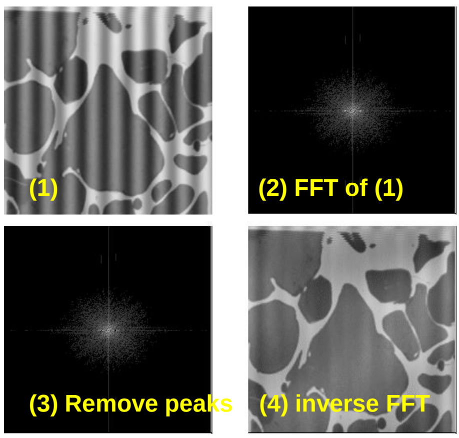
Due to the shape homogeneity, 'equal distance' between the 'boarder' shapes can be applied as one extra algorithm to separate the objects



example

Enhance contrast: by subtracting an 'averaged' background (2), which is generated by dilating / smearing out the original image

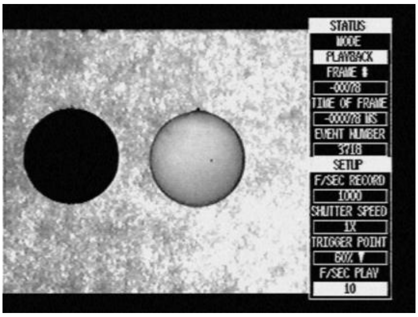




<u>Removal the regular unwanted pattern</u> with the help of FFT (fast Fourier Transformation) that reveals the underlying relation (distribution) of the number array.

Example 1.

- We'd like to analyze the motion of the two spheres from the recorded video.



- Objective: Locate the sphere position in each frame.
- Goal: By the displacement between two consecutive images, the sphere velocity can be estimated <u>with a given</u> <u>elapsed time.</u>

Conert an AVI video into an image sequence at a specific frame rate.

Preparation: generate image sequence, obtain **Δt**

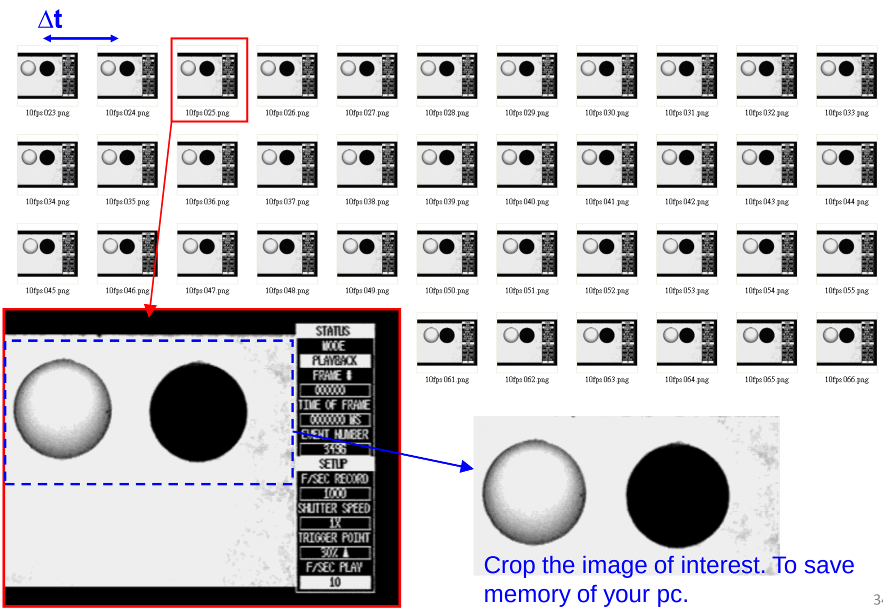


Image processing:

- Convert the image type: RGB to Grayscale (intensity) to manipulate the image

Apply ‘**threshold**’ to convert grayscale image into BW image:

For each pixel
I >threshold, make it 1 (Black pixel)
I <threshold, make it 0 (white pixel)

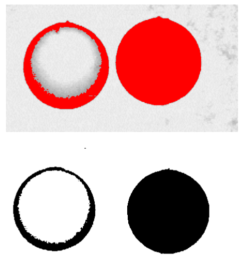
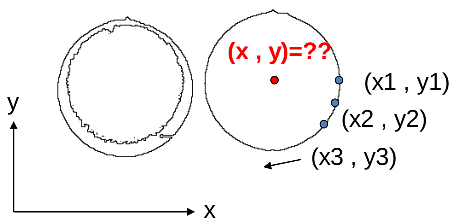


Image Processing (3)

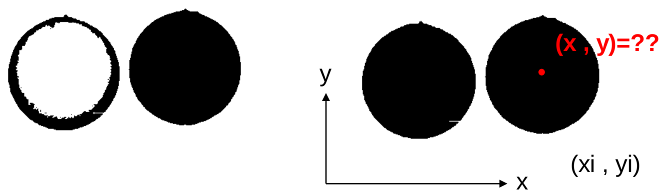
- Method1:
  – Use the pixels with 0-1-1 or 1-1-0 neighbors to locate the **edge** of the black pixels.
  – Then use the circumferential pixels to locate the sphere center.



Large deviation due to the thin line that defines the edge of the sphere

Image Processing (4)

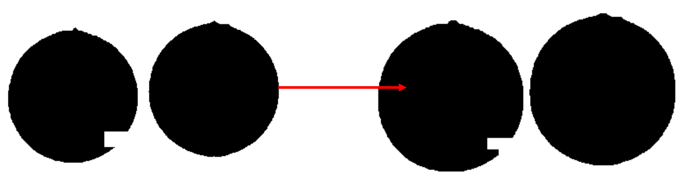
- Method2:
  - **Fill holes**: now all the black pixels present the solid sphere while white pixel stands for the ambient liquid.
  - Average out the coordinates (xi, yi) of all the black pixels would give an averaged position for the sphere.



Due to the greater number of 'solid' pixels, the estimated sphere center is less sensitive to the BW image manipulations.

Image Processing (5)

- Small defect due to unevenness of the light intensity
  - Dilate / Erode
  - Dilate-then-erode the images (Close)
  - Erode-then-dilate the image (Open)

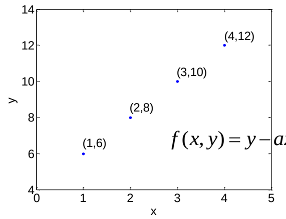


Dilate 3 times (radially) to diminish the influences of the defect on the averaged coordinate of the sphere (more black/less white pixels).

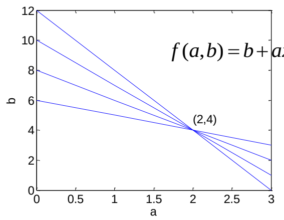
Image Processing (6) : Hough transformation (2nd method)

Concept: shape-detection in parameter space

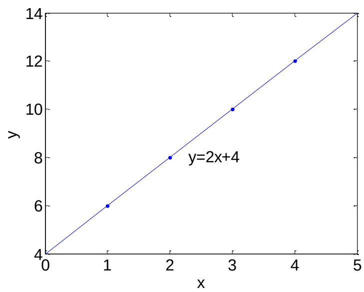
Line:



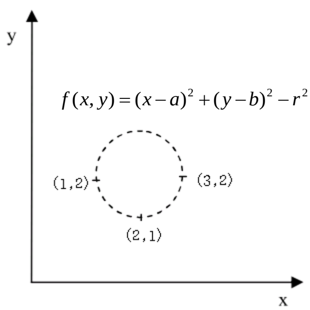
$$f(x,y)=y-ax-b=0$$



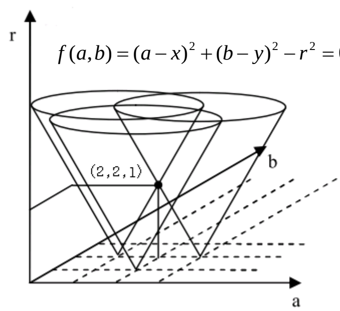
$$f(a,b)=b+ax-y=0$$



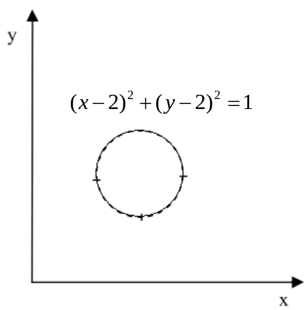
Circle



$$f(x,y)=(x-a)^2+(y-b)^2-r^2=0$$

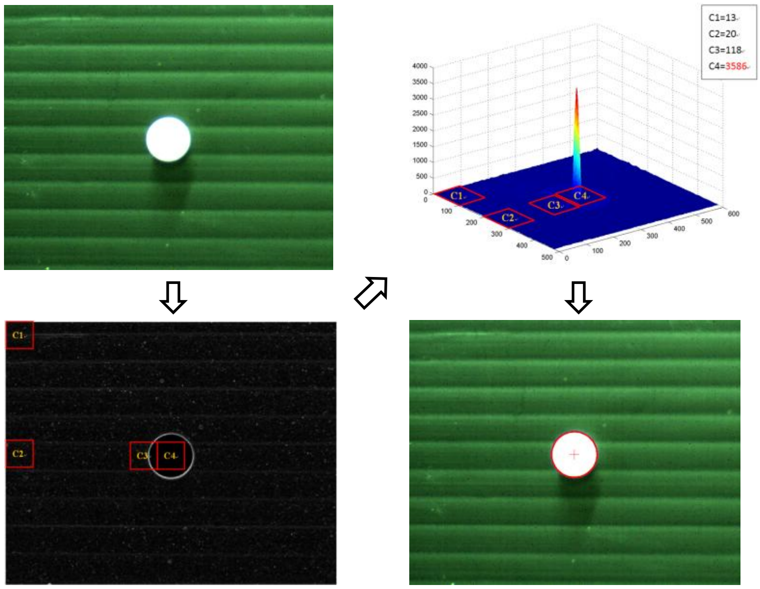


$$f(a,b)=(a-x)^2+(b-y)^2-r^2=0$$

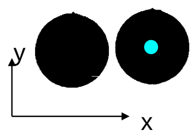


$$(x-2)^2+(y-2)^2=1$$

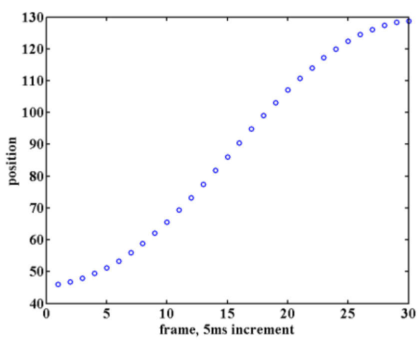
Circle, used to locate a sphere center

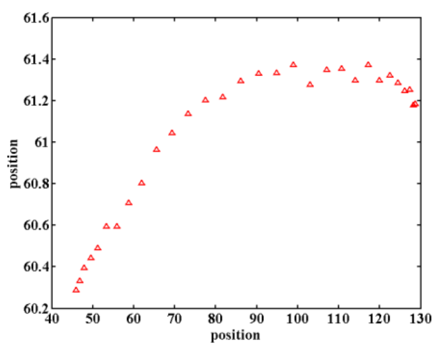


Post-analysis (1)

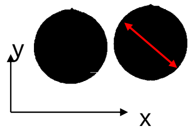


- After we obtain the sphere position in the consecutive images, post-analysis can be followed to obtain the varying velocity and the acceleration components





Post-Analysis (2)



- Velocity: differentiation of the digitized position-time data

To express df/dx : use

$$\frac{df}{dx} \equiv \lim_{h\to 0}\frac{f(x+h)-f(x)}{h} = \lim_{h\to 0}\frac{1}{h}\left[f(x)+f'(x)h+\tfrac{1}{2!}f''(x)h^2+\cdots-f(x)\right]$$
$$= f'(x)+\lim_{h\to 0}\left[\tfrac{1}{2!}f''(x)h+\tfrac{1}{3!}f^{(3)}(x)h^2+\cdots\right]$$

Thus if $\boxed{f'_{app}(x) \equiv \dfrac{f(x+h)-f(x)}{h}}$ **Forward-difference scheme**

$$\underline{\text{Error}} = f'_{app}(x) - f'(x) = \frac{f(x+h)-f(x)}{h} - f'(x) = \underline{o(h)} + H.O.T.$$

Post-analysis (3): Finite difference equation

To express first order differential equation

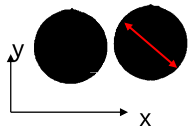
- **Forward difference**
- **Backward difference**

$$\begin{cases} \dfrac{df}{dx} = \dfrac{f(x+h)-f(x)}{h} + o(h) \\ \dfrac{df}{dx} = \dfrac{f(x)-f(x-h)}{h} + o(h) \end{cases}$$

- **Central difference** $\quad \dfrac{df}{dx} = \dfrac{1}{2}\left[\dfrac{df}{dx_F} + \dfrac{df}{dx_B}\right] + o(h^2)$ **More accurate**

$$\frac{df}{dx_C} \equiv \frac{1}{2}\left[\frac{df}{dx_F}+\frac{df}{dx_B}\right] = \lim_{h\to 0}\frac{1}{2}\left[\frac{f(x+h)-f(x)}{h}+\frac{f(x)-f(x-h)}{h}\right]$$
$$= \lim_{h\to 0}\frac{1}{2h}\left[f(x+h)-f(x-h)\right]$$
$$= \lim_{h\to 0}\frac{1}{2h}\left[\begin{array}{l}\left(f(x)+f'(x)h+\tfrac{1}{2!}f''(x)h^2+\cdots\right)- \\ \left(f(x)-f'(x)h+\tfrac{1}{2!}f''(x)h^2-\cdots\right)\end{array}\right]$$

Post-analysis (4)



- Velocity, if taking forward difference scheme:

$$u_i = \frac{(x_{i+N} - x_i)}{N}$$ **(pixel/frame)** N: Number of elapsed frames

To obtain the actual velocity, need conversion between the digital and the actual sizes and time duration:

(1) Length: measure how many pixels correspond to a sphere diameter (**flow characteristic length scale that is easy to measure**)

(2) Time: by know the actual time duration between two processed images (N of elapsed frames / recording frame rate)

$$u_i = \frac{(x_{i+1} - x_i)}{\Delta t} \times \left[\frac{5mm}{72\,pixels} \times \frac{?\,ms}{n\ \ frames} \right]$$ **(mm/s)**

Error estimation

- Since the time difference between each frame is precisely set (via digital camera and the conversion software), the main source of error in determining the sphere velocity comes from the image analysis!
  - This is why we need to be careful about image manipulations.

- Repeat several measurements, then an averaged velocity profile will be obtained. The standard deviation of the residues of each measurement from the mean value can be calculated.

- If a line is fitted to the empirical data, then a fitting deviation can be calculated likewise.

Error Estimation

- Assume we want to measure a quantity $F(x, y, z)$ that requires two actual measurements on x, y, and z from the experiment.
- The errors in measuring x, y, and z will contribute to errors in $F(x, y, z)$. A general error estimation can be obtained as the following.
- Denote the uncertainty in the measured x, y, and z by $\delta x, \delta y$, and δz .
- The overall uncertainty in F is thus:

$$dF^2 = \left(\left.\frac{\partial F}{\partial x}\right|_{y,z} \delta x \right)^2 + \left(\left.\frac{\partial F}{\partial y}\right|_{x,z} \delta y \right)^2 + \left(\left.\frac{\partial F}{\partial z}\right|_{x,y} \delta z \right)^2$$

- Normalizing the uncertainty with the measured value of F gives the generalized error estimation

$$\boxed{\hat{E} = \sqrt{\frac{(dF)^2}{F^2}}}$$

Error Estimation example

Assume that we want to calculate the coefficient of restitution for a collision between two solid spheres, defined as

$$e = -\frac{U_2 - U_1}{V_2 - V_1}$$

where V_1, V_2, U_1, and U_2 are the velocities of each sphere before and after the collision measured in the experiment.

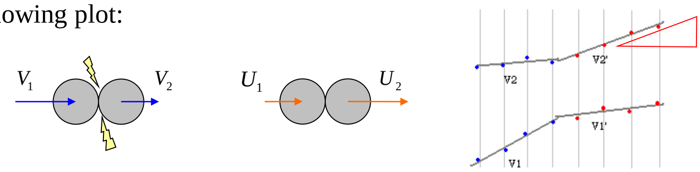
From the previous formula, the error would be $\hat{E} = \sqrt{\frac{(de)^2}{e^2}}$

$$de^2 = \left(\frac{\partial e}{\partial U_2} \delta U_2 \right)^2 + \left(\frac{\partial e}{\partial U_1} \delta U_1 \right)^2 + \left(\frac{\partial e}{\partial V_2} \delta V_2 \right)^2 + \left(\frac{\partial e}{\partial V_1} \delta V_1 \right)^2$$

$$\hat{E} = \sqrt{\frac{de^2}{e^2}} = \sqrt{\frac{(\delta U_2)^2 + (\delta U_1)^2}{(U_2 - U_1)^2} + \frac{(\delta V_2)^2 + (\delta V_1)^2}{(V_2 - V_1)^2}} \longleftarrow \text{??}$$

Error Estimation example

For example, we can track the sphere motion in time, which gives the following plot:



We can fit a line through 4 position data points and use the slope for the sphere velocity.

Then the error in estimating one sphere velocity may be calculated using the standard deviation of line-fitting.

$$\delta U_2 = \sqrt{\frac{\sum_{i=1}^{N} \left(y_{2i\_\text{predicted}} - y_{2i\_\text{measured}}\right)^2 / \Delta t}{N}}$$