

Short Notes

- **Filtering**

- Filtering in spatial domain : $g(x,y) = T [f(x,y)] \rightarrow$ direct manipulation
- Filtering in frequency/wavelet domain :
 - image transformed \rightarrow frequency/wavelet domain \rightarrow process \rightarrow back-transformed \rightarrow image domain
 - $F(x,y) = FFT [f(x,y)]$
 - $g(x,y) = FFT^{-1} \{ T [F(x,y)] \}$

1. Assign an arbitrary constant value (0 by default)
2. Mirror reflecting the array cross the borders
3. Replicate the closest value

Avantages:

- Not sensible to extreme values or outliers
 - The value is selected among existing values
 - More edge preserving than the Gaussian filter
- $$G\sigma^*I - G\sigma^*(G\sigma^*I) = (G\sigma - G\sigma^*G\sigma)^*I = (G\sigma - G2\sigma)^*I$$

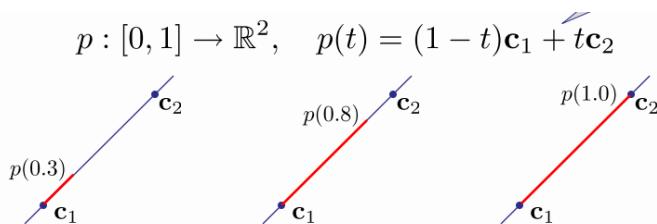
Small Questions:

1. Vessel segmentation, design a method, result cell detection, possible disadvantages
2. Quantitatively and qualitatively check the validation of cell detection
3. How registration is needed in medical application? SSD SAD NCC formula, briefly describe, pros&cons!
 - Why above similarity measurement cannot be used for multi-model?
 - How to use Shannon entropy to register this model?
4. How does B-spline being used in deformable registration
 - Define the forward, backward wrapping. Which is better?
 - Why multi-resolution is good? 2 reasons
5. How to separate the stained histology into RGB
6. non-corresponding...
7. Marked...
8. Two methods
9. Three ways

Deformable Registration

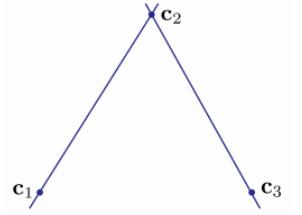
- Dense deformable registration
 - optimize → displacement @ **image Location**
 - “parameter-less”
 - assume image domain as “Continuous”
 - Formulation → calculus variations
- Grid-based deformable registration
 - Parameterize
 - place Grid of control points (on image)
 - optimize → displacement of **Control points**
 - EX: Free-form deformation (FFD)
- Registration Energy : $E(u) = S(u)$
 - $u \rightarrow$ displacement field
 - $S(u) \rightarrow$ image similarity measure
 - $S(u) = \text{sum}(I_1(x) - I_2(u(x)))^2$ (SSD)
 - $I_1 \rightarrow$ Fixed image (reference)
 - $I_2 \rightarrow$ Moving image (template)
- **Why do we need Regularization?**
 - Min. the similarity measure is not enough
 - Min. different s.m. → Aperture problem
 - Can be used to include pre-knowledge
 - EX: tissue properties
 - Practical reason :
 - with no regularization → will have high number of local minima in the energy function $E(u) \rightarrow$ Bad for optimization
- Apply Regularization
 - penalty term → intensity-based
 - $E(u) = S(u) + \alpha R(u) \rightarrow \text{min.}!$ minimize the energy function
 - ex: Diffusion regularization (1st order regu.)
 - sum over (derivative of x,y,z) 2

- **What is registration in general?**
 - bring two or more images into spatial alignment
 - establish a common geometric reference frame
- **What do we need registration for in the medical context?**
 - we want to use as much information as possible to get the maximum value of anatomical images
 - **FOR** : to minimize and compensate the shifting or moving difference/errors on patient or organs
- **How is an image deformation typically represented?**
 - we cannot represent arbitrary deformation by simply using transformation matrix
 - it basically describe the motion for every individual pixel between 2 images
 - Displacement Field
 - function defined for every pixel in image domain
 - displacement in d dimension
 - transformation function $T(x) = x + u(x)$
 - $u(x) \rightarrow$ difference between x and transformation $T(x)$
- **Why do we need regularization?**
- Parametric Curves & B-Splines
 - How to represent a smooth, curved shape mathematically?
 - implicit functions $\rightarrow x^2 + y^2 = 1$
 - difficult to directly obtain points
 - polynomials $\rightarrow f(x) = ax^2 + bx + c$
 - complexity shape & no closed curves & oscillations...
 - **Parametric curves** \rightarrow Explicit representation for points on shape
 - Mapping : scalar parameter domain P \rightarrow vector-valued target domain S
 - Goal: specify Control points \rightarrow define shape of curve
 - Polynomials \rightarrow parametric representation & interpolate points



- ex: line connecting 2 control points $c_1, c_2 \rightarrow$ convex combination!

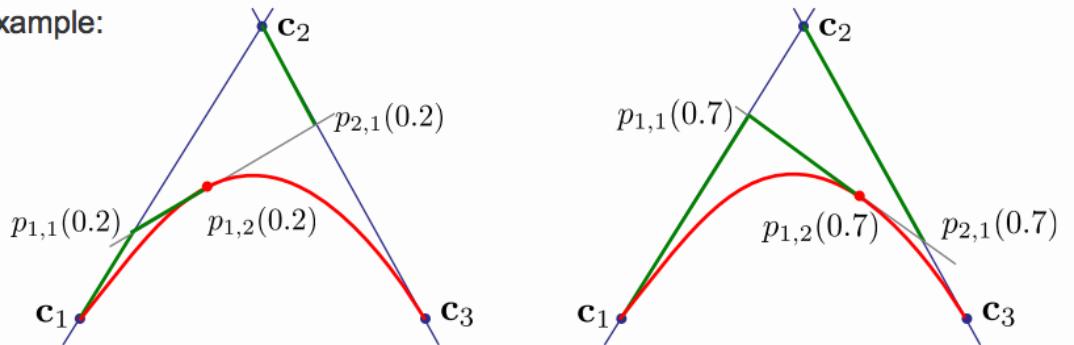
- How to extend to more than 2 points?
 - ex: 3 control points c_1, c_2, c_3
 - interpolation : pairs of points \rightarrow line segmentation
 - $p_{1,1}(t) = (1-t)c_1 + tc_2$
 - $p_{2,1}(t) = (1-t)c_2 + tc_3$
 - interpolation : line segment \rightarrow quadratic polynomial
 - $p_{1,2}(t) = (1-t)p_{1,1}(t) + t p_{2,1}(t)$
 $= (1-t)^2 c_1 + 2t(1-t) c_2 + t^2 c_3$



- **Bezier Curves**

$$\begin{aligned} p_{1,1}(t) &= (1-t)c_1 + tc_2 \\ p_{2,1}(t) &= (1-t)c_2 + tc_3 \\ p_{1,2}(t) &= (1-t)p_{1,1}(t) + tp_{2,1}(t) \end{aligned}$$

Example:

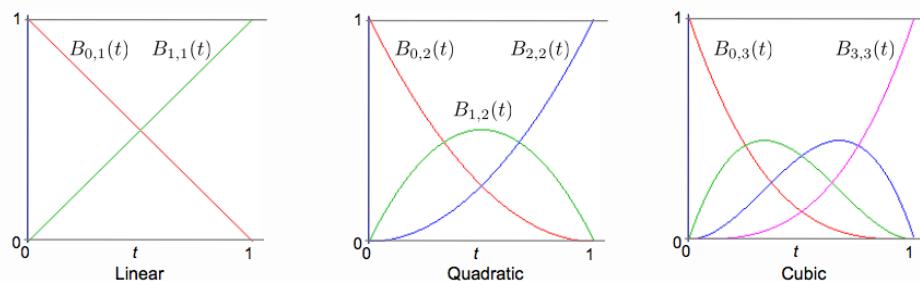


- Properties:

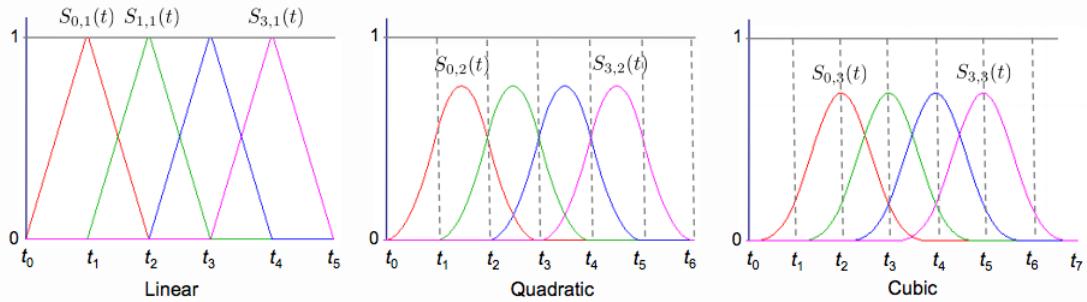
- no need to pass through all control points
- convex hull
- degree = numbers of control points
 - 2 \rightarrow linear $d=1$
 - 3 \rightarrow quadratic $d=2$
- complex shape (many control points) \rightarrow high Bezier Curve
- no locally
- *** Attach Bezier curve with fixed degree!!!

- alternative view:

- at each location along the curve, every control point has a non-zero weight
- every control point has influence on the whole curve
- Bernstein basis polynomials for $d = 1, 2, 3$



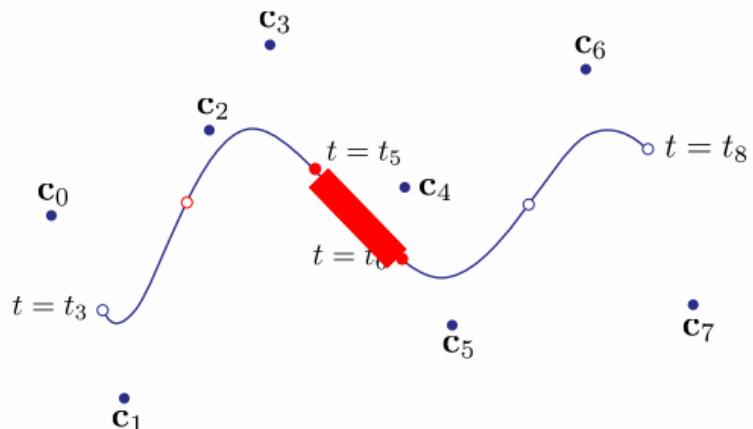
- Bezier curve → **B-Spline Curves**
 - Bezier curve
 - no local control
 - all control points are taking into account
 - degree related to control points
 - B-Spline Curves
 - compute curve from : control points nearest to a given location on the curve
 - free choice of degree



- note : B-Spline have local support! i.e. each of them is zero outside a certain parameter range

B-Spline Curve Example

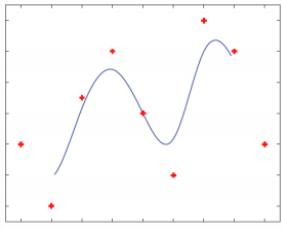
$n = 12$ $d = 3$
Control points:
 $n - d - 1 = 8$



- properties
 - no need pass through all control points
 - convex hull of control points
 - degree $d \rightarrow d-1$ continuous derivatives
 - degree not depend on control points
 - typically use → cubic B-spline ($d=3$)
 - parameter range divided into sequence of knots
 - typically use → uniform knot sequence
 - locality → changing control point → only region effect

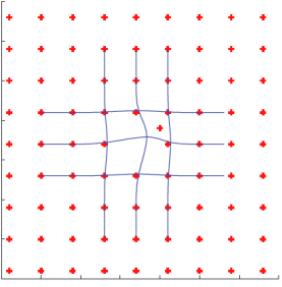
■ Curves → 2D patches

- parameterization of a curve
 - one parameter (t) varying as the curve is traced out!



$$s(t) = \sum_{i=0}^d S_i^d(u) \mathbf{c}_{q-d+i}$$

- parameterization of the deformation of a 2D patch
 - 2 parameters (t_1, t_2) varying throughout the patch!



$$s(t_1, t_2) = \sum_{i=0}^d \sum_{j=0}^d S_i^d(u_1) S_j^d(u_2) \mathbf{c}_{q_1-d+i, q_2-d+j}$$

- $c_{ij} \rightarrow$ 2D grid control point
- $u_1, u_2, q_1, q_2 \rightarrow$ analogy to u & q

■ Deformation of 3D volumes

- parameterization
 - 3 parameters (t_1, t_2, t_3) varying throughout the space

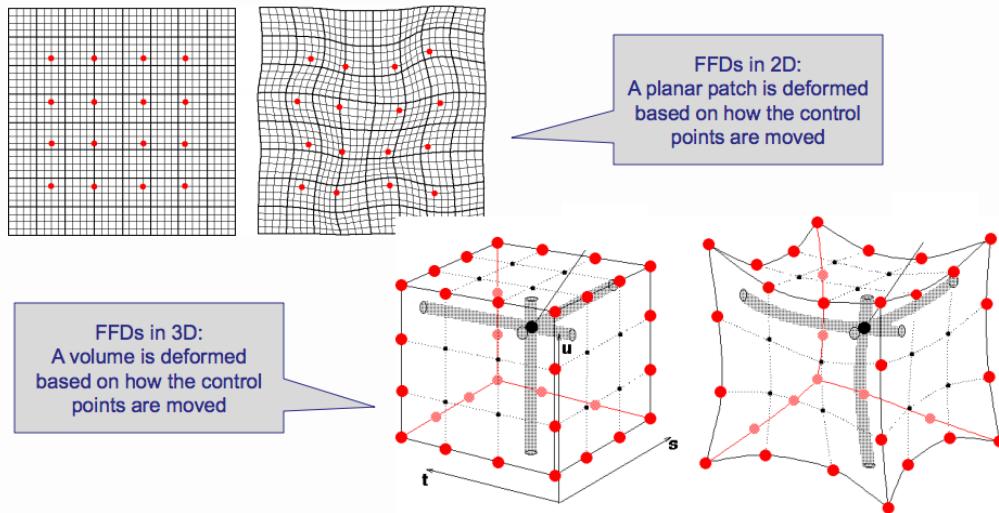
$$s(t_1, t_2, t_3) = \sum_{i=0}^d \sum_{j=0}^d \sum_{k=0}^d S_i^d(u_1) S_j^d(u_2) S_k^d(u_3) \mathbf{c}_{q_1-d+i, q_2-d+j, q_3-d+k}$$

- $c_{ijk} \rightarrow$ 3D grid (regularly spaced) control point
- $u_1, u_2, u_3, q_1, q_2, q_3 \rightarrow$ analogy to u & q

■ Deformation of 2D patch & 3D volume

- moving control points → deform the space between
- Free-form Deformation FFD

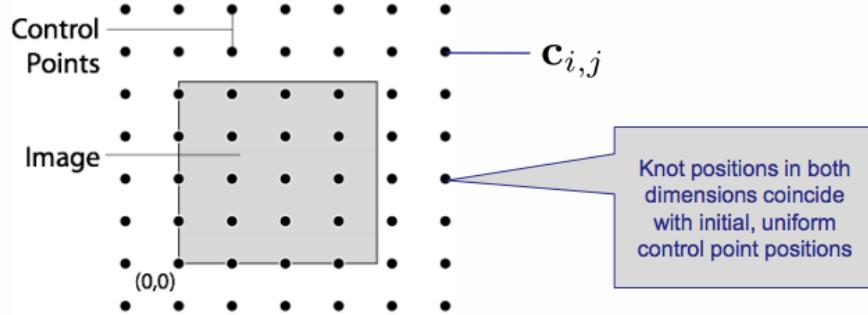
- Free-form Deformation FFD
 - IDEA : deform an object by deforming the space surrounding it!



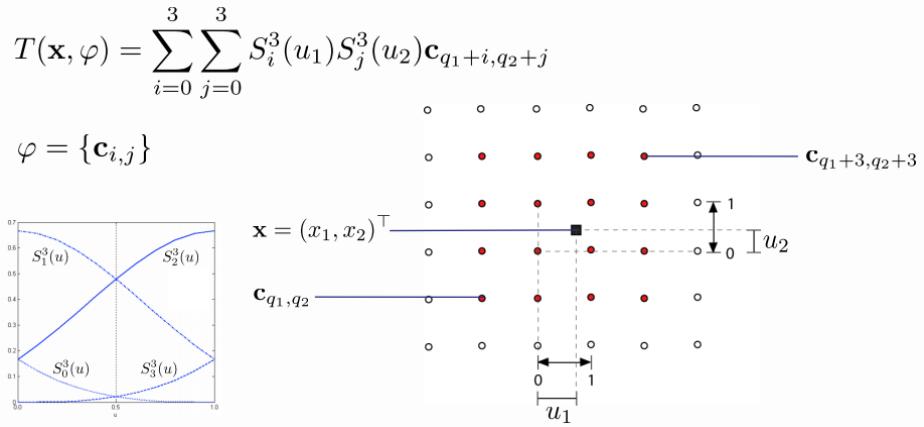
- 1st → place object within grid of control points
- 2nd → approximate deformation between control points using **B-Spline**
- 3rd → Based on **Uniform cubic B-spline**

- **What is parametric curve?**
 - by using a mathematical way to represent a curve
 - explicit representation for points on shape
 - mapping the scalar parameter domain P to vector-value target domain S
- **What are the main advantages of B-Spline over Bezier curves?**
 - free to choice control points
 - no need to consider all control points
 - changing control points only affect locally
- **What is the difference between a parametric curve in 2D and a B-Spline transformation on a 2D patch?**
 - one parameter
 - two parameters
- **What are free-form deformations FFD?**

- Deformable registration using FFDs
 - embed image domain (template) in regular grid of control points \mathbf{c}



- move control points to smoothly deform the image & compute displacement for every pixel location (dense deformation field $u \rightarrow B\text{-spline}$) \rightarrow transformation function T



- wrap the moving image using the dense displacement field u
- repeat until the moving matches the reference image as good as possible

- **optimization strategy**

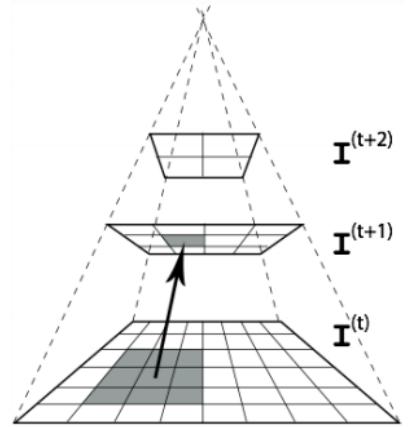
- GENERAL formulation for registration Energy!
 - $E(u) = S(u) + \alpha R(u) \rightarrow \min$
- Control point \rightarrow registration Energy! (not directly on displacement field u)
 - $E(\phi) = S(\phi) + \alpha R(\phi) \rightarrow \min$

- **Multi-Resolution Approach**

- natural deformation occur on different level, local global
- computation is really costly on full resolution images, especially 3D

- **Registration on different resolution!**

- **resampling**
 - gaussian resolution pyramid
 - initialize control point grid
- **registration**
 - start with lowest resolution image
 - perform registration
- **subdivision**
 - subdivide obtained control point grid
 - register again on next finer image
- Why good?
 - reduce computation cost!
 - ???



Segmentation

- segmentation in medical imaging
 - classify image into segment (anatomical/pathological interest)
 - retrieve quantitative measures → size, pathology(tumor,aneurysm...)
 - analyze shape, appearance, dynamic behavior of structure of interest

Pixel-based	Region-based	Edge-based
<ul style="list-style-type: none">• Histogram• Thresholding• Influence of noise<ul style="list-style-type: none">◦ filtering◦ morphological operation• EM-clustering• K-mean clustering	<ul style="list-style-type: none">• Region growing• Region splitting & merging<ul style="list-style-type: none">◦ quadtree◦ adjacency graph• <u>Active counters</u>	<ul style="list-style-type: none">• Basic edge detection• Canny filter• Laplacian Zero-crossing• <u>Active counter</u>

What is **Pixel-based approach?**

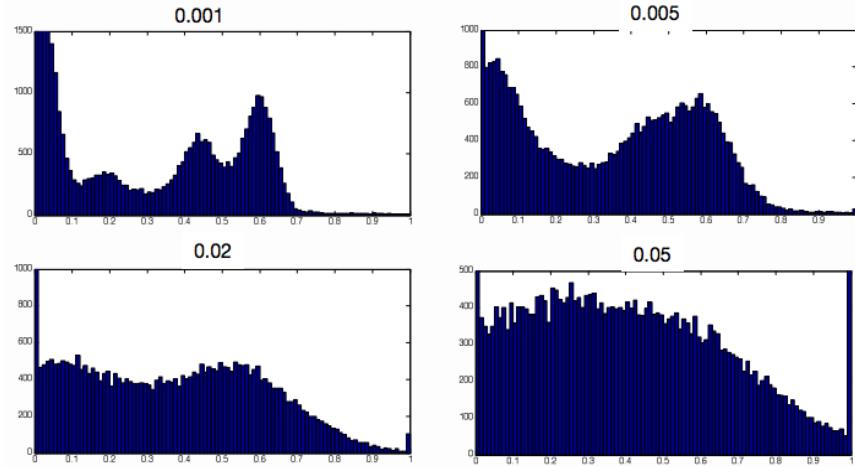
- Only consider the intensities of the pixels
- **How to formally represent Intensity distribution?**
 - Plotting the Histogram
- Definition :
 - In x-axis → define density segments (e.g. between 1-10, 10-20...)
 - In y-axis → represent How many **special intensity Pixels** are within segment range
- **How do we segment the image using the Histogram?**
 - Regroup pixels → similar intensities
 - Define thresholds → separate the classes/mode
 - Define the i-th homogeneous region by :

$$\Omega_i = \{x \in R^2 \setminus T_i < I(x) \leq T_{i+1}\}$$

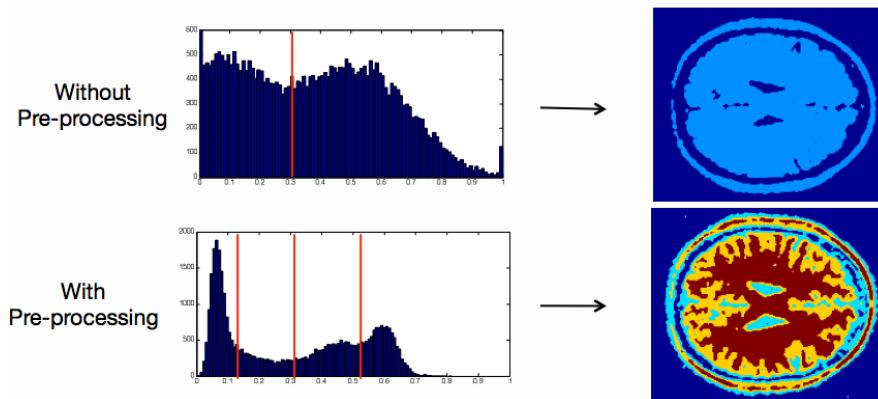
- Could we Automatically select the Threshold?
 - **EM Clustering** → solution for Gaussian distribution
 - Expectation-Maximization Algorithm
 - Alternating Alg.
 - **E-step** : calculate expectation value → each sample belong to each classes for the current parameter configuration
 - **M-step** : update the parameter configuration → Maximize the expectation values
 - **K-mean** → solution for Non-Gaussian distribution
 - Clustering alg.
 - **Iterative** alg.
 - set number for intended classes to N
 - start from initial mean values:
 - $\{m_i\}_{1 \leq i \leq N}$
 - **Assignment-step** : assign the intensities → cluster with closest mean :
 - $$\Omega_i = \{x \in R^2 \mid |I(x) - m_i| \leq |I(x) - m_j|\}$$
 - **Update-step** : calculate the new mean intensity (if new clusters required!)

$$m_i = \frac{1}{|\Omega_i|} \sum_{x \in \Omega_i} I(x)$$
 - Limitations :
 - **Number of classes N** should be known in advance
 - Or obtained by :
 - Minimizing → **with in cluster scatter**
 - Maximizing → **between cluster separation**
 - implicitly assume that the distribution → **equal standard deviation**
 - **sensible to noise**

- How robust is this method in respect to noise?
 - How histogram change when noise increase!
 - corrupt the image with Gaussian noise with 0-mean & increasing variance
 - mode cannot be differentiated → **Impossible segmenting image**

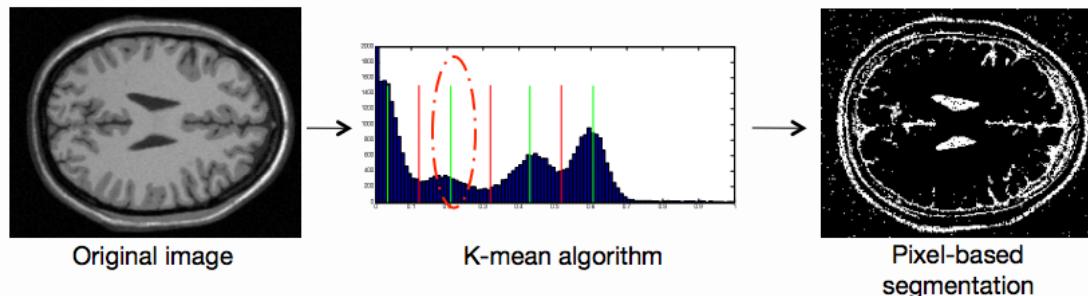


- How do we compress the image/histogram?
 - Smooth → attenuate noise (Gaussian, median, anisotropic diffusion filters)
 - Result → after spatial Gaussian smoothing & k-mean

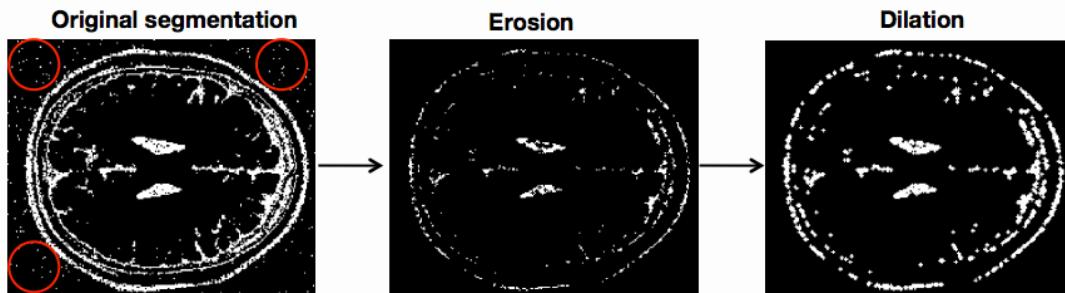


■ Post-processing

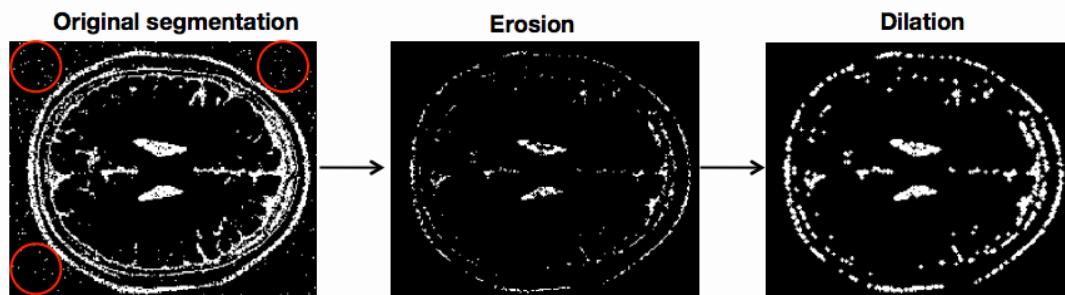
- not enough to only see the intensity
- EX: How to segment the **cerebral vesicles**?



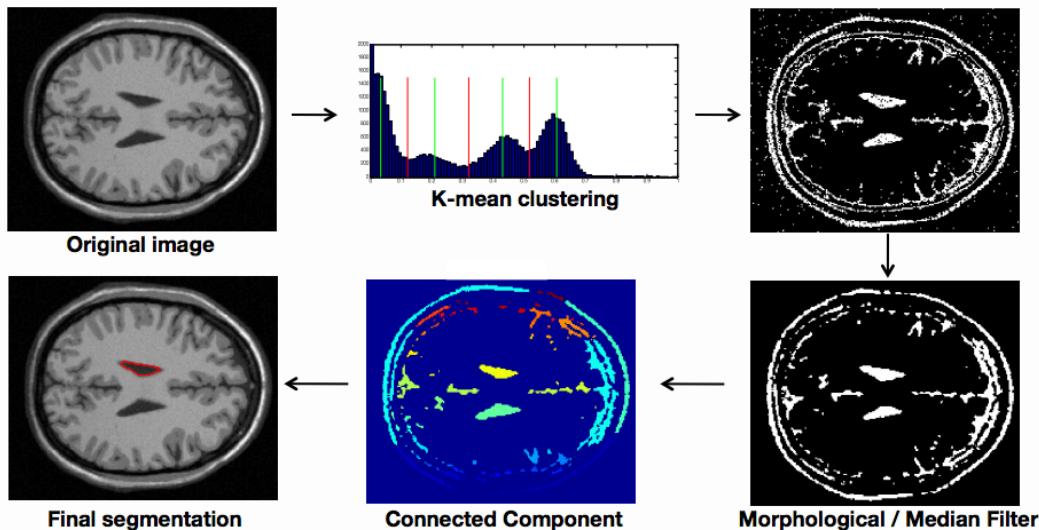
- **Step 1a : Morphology / Median filter** → filter the pixels which should be included and which should not be included
 - **Remove the False-positive pixels** (outside object!)
 - 1st : apply erosion operation (remove)
 - 2nd : apply dilation operation (restore)
- ⇒ Morphological Opening method



- **Remove the False-negative pixels** (outside object!)
 - 1st : apply dilation operation (fill the holes)
 - 2nd : apply erosion operation (restore)
- ⇒ Morphological Closing method

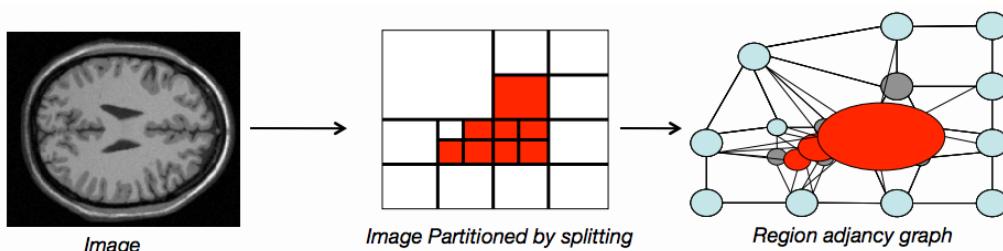


- Remove both false-negative & false-positive pixels
 - 1st : apply Close - Open
 - 2nd : apply Open - Close
- ⇒ Close - Open NOT EQUAL Open - Close
- Step 1b : Median Filtering
 - Step 2 : Connected components → deal with falsely selected objects
 - Definition:
 - 2 pixels connected → exist a path between them consisting entirely of pixel with the same label
 - subset of pixels is a connected component → exist a path between every pair of pixels within this subset
 - Method:
 - Label the connected components in image
 - Manually select the connected component correspond to one of the vesicle!



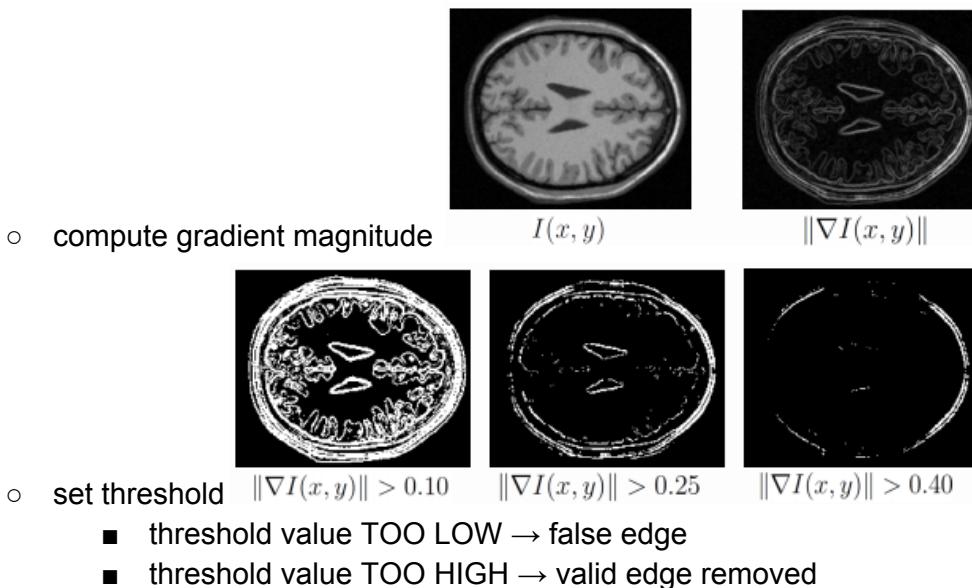
What is Region-Based approach?

- Incorporated topological/spatial information → post process the pixel
- How to incorporate this information directly?
 - Bottom up approach : **Region growing**
 - grow a region from a seed point
 - **Principle:**
 - start with seed point
 - add more points from borders to the growing region
 - add points fulfil a homogeneity criterion
 - stop
 - efficient approach:
 - First In First Out **FIFO**
 - get seed point
 - add seed point to queue of pixels to visit
 - queue is not empty
 - pick & remove first pixel
 - homogeneous → label as belonging to the growing region (visit) & Add unvisited neighbor pixels to the queue
 - Top down approach : **Region splitting**
 - break image into set of disjoint region
 - Step 1 : **SPLIT**
 - iteratively **split** the image until every subregion is homogeneous
 - **Method:**
 - image saved as **quadtree**
 - **root** → **image itself**
 - **leafs** → **subregion of the image**
 - **How to build a quadtree?**
 - check the regions → homogeneity or not?
 - If R → Not homogeneous
 - a. Split into 4 new subregions
 - b. consider the first leaf
 - If R → Homogeneous
 - a. consider the last created leaf that was not already studied
 - b. QUIT if such leaf does not exist!
 - Problem : over-segment!
 - **Merging** → build the **Region Adjacency Graph (RAG)**
 - For each node in RAG → check adjacent region



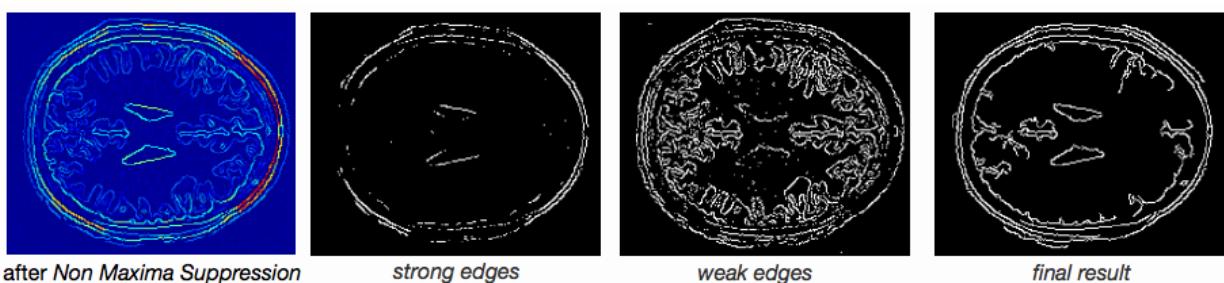
What is Edge-based approach?

- edge correspond to →
 - local maximum → 1st derivative magnitude (gradient)
 - zero-crossing → 2nd derivative (Laplacian)
- Edge detection from **Gradient**:



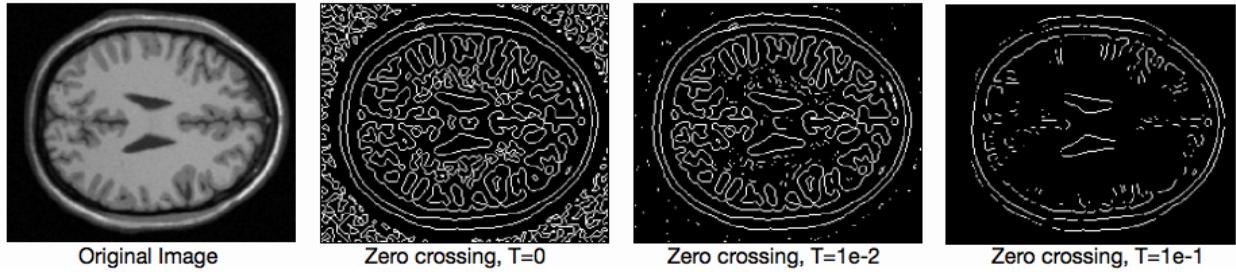
- **Canny detector**

- **Step 1** : gradient contain wide ranges around local maxima
 - use the **gradient direction** $\alpha(x,y)$ → thin the ridges
 - Method : non-maxima suppression
 - compute gradient magnitude & direction
 - find neighbor in the nearest basic edge direction (0, 45, 90, 135 degree) → $\alpha(x,y)$
 - suppression : if neighbor gradient is greater → set current magnitude 0
- **Step 2 : Hysteresis thresholding**
 - Method :
 - create 2 images:
 - containing **strong edges**: $\|\nabla I(x, y)\|_N > T_h$
 - containing **weak edges**: $(\|\nabla I(x, y)\|_N > T_l) - (\|\nabla I(x, y)\|_N > T_h)$
 - mark as valid : all strong edge pixels (connected to a valid pixel)



- **Laplacian zero-crossing**

- Method:
 - compute Laplacian
 - check every pixel → if there is a zero-crossing
 - check if the edge is strong enough



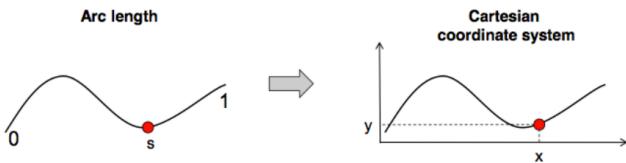
What is Active counters?

- advanced image segmentation
- Principle:
 - define initial contour
 - evolve the contour towards an optimal shape and position
- 2 approaches :
 - Explicit formulation → **Snakes**
 - Implicit formulation → **Level sets**

What is Snakes?

- the contour is described by parameters $\mathbf{C}(s) = [x(s) \ y(s)]$

$$C : [0, 1] \mapsto \mathbb{R}^2 \quad s \rightarrow C(s) = [x(s) \ y(s)]$$



- points are moved by applying forces
 - **External** → relates to image intensity information (edge, intensity region...)
 - **Internal** → relates to regularity of the contour (curves, length...)

$$E(C) = \alpha \int_0^1 E_{int}(C) ds + \beta \int_0^1 E_{ext}(C) ds$$

$$E(C) = \alpha \int_0^1 E_{int}(C) ds + \beta \int_0^1 E_{ext}(C) ds$$

$$E_{int}(C) = w_a(C) \left(\frac{\partial C}{\partial s} \right)^2 + w_b(C) \left(\frac{\partial^2 C}{\partial s^2} \right)^2$$

controls contour's elasticity controls contour's rigidity

weights

convolution of image with Gaussian filter before edge filter (common practice)

$$E_{ext}(C) = |\nabla(G_\sigma * I(C))|^2$$

exemplary edge-based term

Histology

“study of the microscopic anatomy of cells and tissues of plants and animals”

- Process:
 - Sample fixation & embedding
 - Cutting
 - Object plate pinning & cover slipping
 - Staining 染色
 - Viewing
- Whole Slide imaging
 - Scanning of whole specimen 標本 section on high magnification
 - lense & illumination
 - motorize stage
 - slide loader
 - Portion scanning
 - **Ribbon-based**: „scanlines“ produced, 1-D stage → less mechanical jittering, less stitching artefacts
 - **Tile-based**: tiles produced, 2D stage → faster illumination, less anatomy destruction

Part1 : Image Standardization & Color Deconvolution

- Image standardization
 - Problem : Acquisition-to-acquisition → Signal Intensity Variation
 - Pre-processing : Corrected by mapping the intensities to a **Standardized Scale** → using **median**

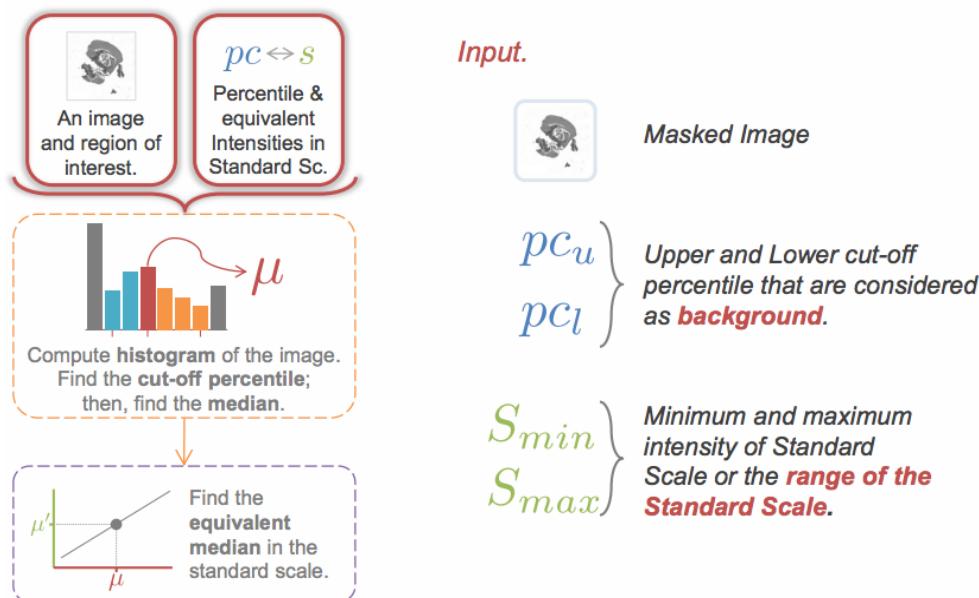
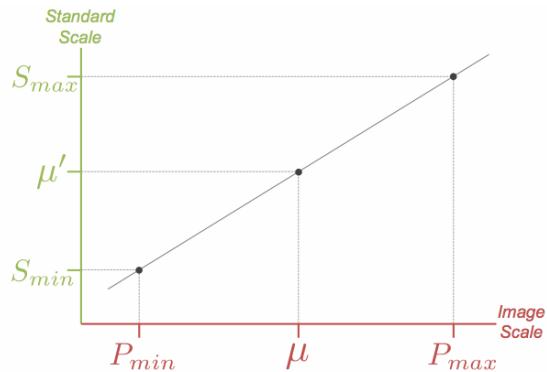
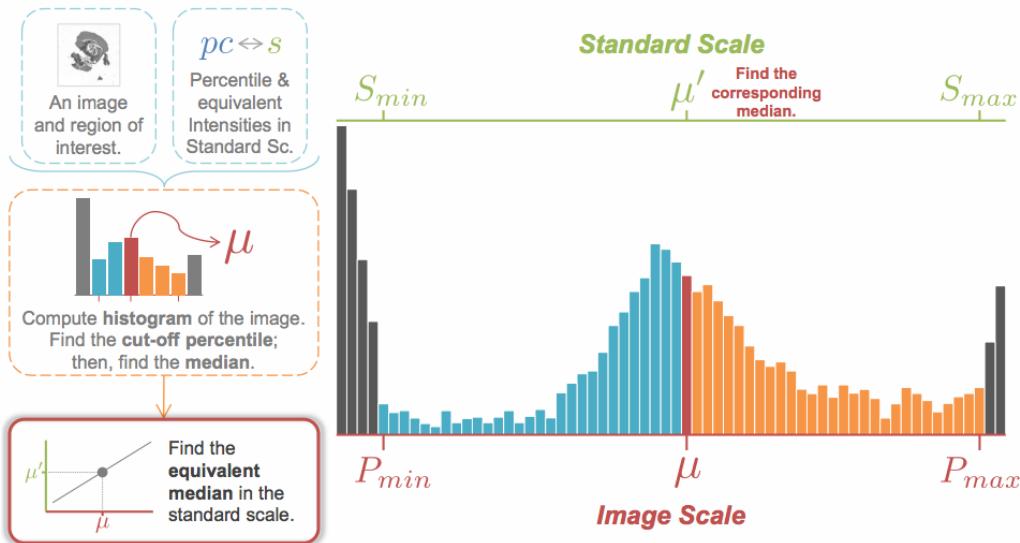
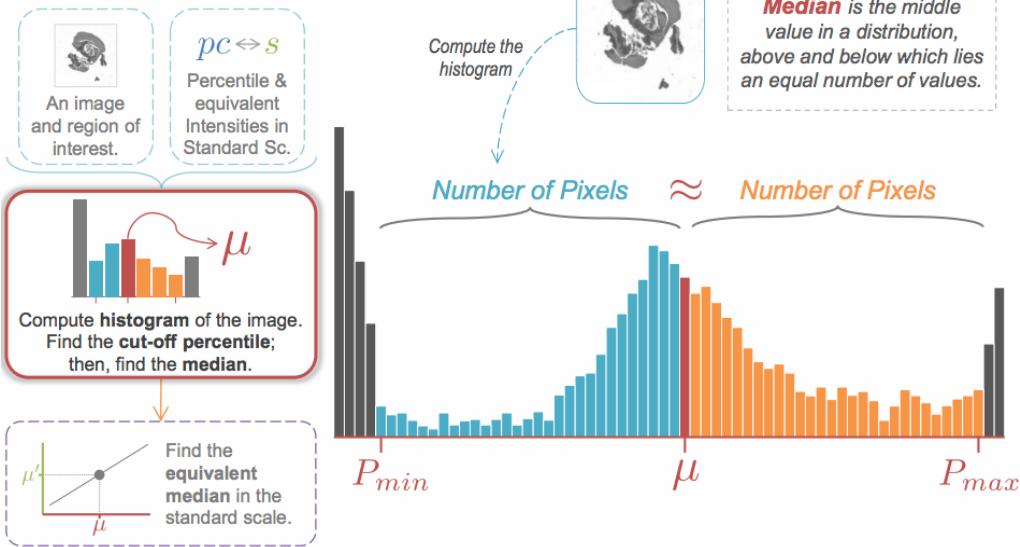


Image Standardization



Therefore, for each image j ,

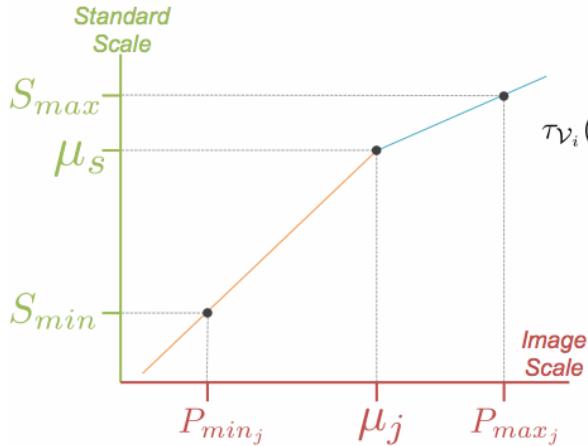
$$\mu'_j = S_{min} + \frac{\mu_j - P_{min,j}}{P_{max,j} - P_{min,j}} (S_{max} - S_{min})$$

- To aggregate all the medians in the Standard Scale, find the **Rounded Mean**:

$$\mu_s = \frac{1}{N} \sum_{j=1}^N \mu'_j$$

Output

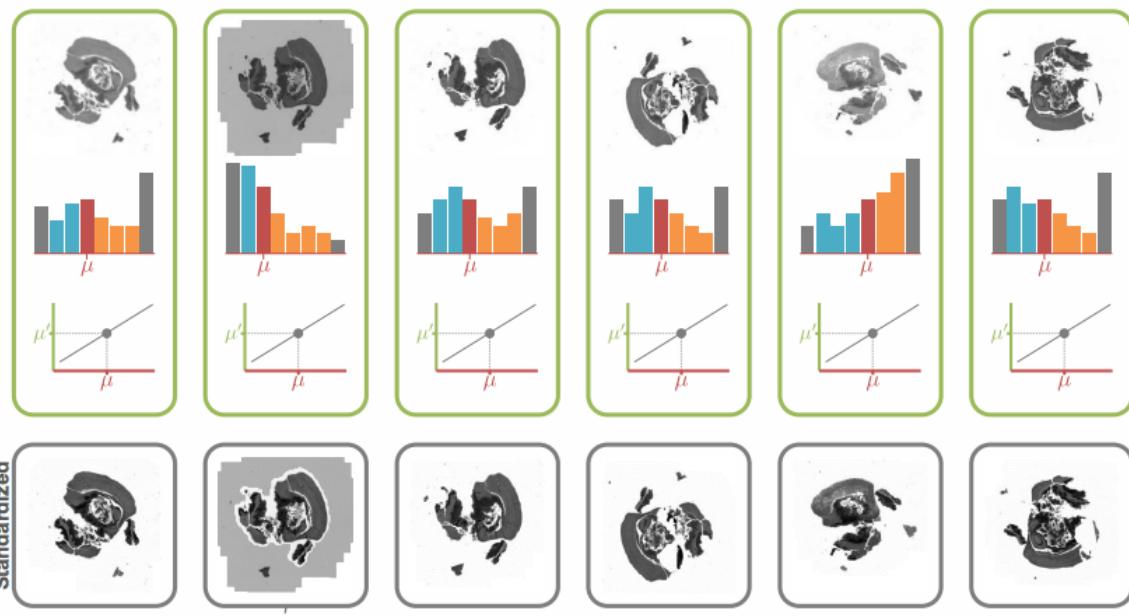
Hence, for all pixels in image j ,



$$\tau_{V_i}(x) = \begin{cases} \left[\mu_s + (x - \mu_i) \frac{S_{min} - \mu_s}{P_{min_j} - \mu_j} \right] & \text{if } x \leq \mu_j \\ \left[\mu_s + (x - \mu_i) \frac{S_{max} - \mu_s}{P_{max_j} - \mu_j} \right] & \text{if } x > \mu_j \end{cases}$$

where x is the image intensity.

*Map sections of the histogram to standard scale
(Linear Mapping)*



Masked Image but still includes small background.

- **RGB & Stain Space**

- Staining is a process of colorization of tissue using chemical dyes
- Different dyes → highlight different anatomical structure
- Usually $\Rightarrow >1$ dyes on each sample
- Typical ⇒
 - Hematoxylin → stain nuclei **blue** 核
 - Eosin → stain cytoplasm **pink** 細胞質
- **stain separation → analysis tasks on microscopic images (cells counting)**

- **Color representation**

- Lambert-Beer's Law :
 - I (light after travelling) = I_0 (light entering the material) * e^{-Ac} (amount of stain) c (absorption factor)
 - light gets absorbed when traveling through a stained specimen
- For **RGB** images → Lambert-Beer's law is applied to each channel separately
- **Color Deconvolution → separate the Stains**
(we want to know the amount of A)

$$\begin{pmatrix} I_R \\ I_G \\ I_B \end{pmatrix} = \begin{pmatrix} I_{0,R} e^{-Ac_R} \\ I_{0,G} e^{-Ac_G} \\ I_{0,B} e^{-Ac_B} \end{pmatrix}$$

- **Color Deconvolution**

- **Calibration**
→ A pure stain's concentration can be measured by staining with only one dye

$$\begin{pmatrix} s_H \\ s_E \end{pmatrix} = \begin{pmatrix} 134 & 125 & 195 \\ 232 & 98 & 192 \end{pmatrix} \quad \begin{matrix} \text{Hematoxylin} \\ \text{Eosin} \end{matrix}$$

→ All color can be transformed to **H, E-concentration-space**

(If the pure stain concentrations are used as a basis!)

→ Normalize **H, E** stain vectors

(balance absorption factor for each separate stain)

→ Deal with 3 vectors \Rightarrow find the third basis vector orthogonal to the others

$s_H \times s_E$

- **Concentration Computation**

→ Get rid of Nonlinear relationship between A & R,G,B colors

\Rightarrow transform to **Optical density space : C = {R,G,B}**

$$OD_C = -\log\left(\frac{I_C}{I_0}\right) = Ac_C,$$

→ Given a pixel color in OD space, $p = (OD_R, OD_G, OD_B)$

\Rightarrow relationship between p and the amount of each stain :

\rightarrow Compute a with $a = M^{-1} * p$

$p = M a,$

$M = (\hat{s}_H^T, \hat{s}_E^T, \hat{s}_R^T)$

Part 2 : Image Registration & Volume Reconstruction

- **Image Registration in Histology**
 - Registration of histology → Colocalization
 - Analysis of the same anatomy with different stains
 - 2 continuous sections & 1 single re-stained section
 - Registration of virtual slides is not trivial!
 - multi-modality registration due to different staining
 - nonlinear transformation due to motion
 - fixation & embedding & sectioning
 - **Problem** : image size is toooo large to fit into RAM
 - **Possible Solution :**
 - Register on Higher pyramid level & propagate result
⇒ fast & less accurate
 - Compute similarity measure patch by patch & sum up for global result
⇒ accurate & very slow
 - Do Entire Registration Patch by Patch!
 - **Patch-wise Registration**
 - Idea : divide virtual slide into small patches & perform registration per patch independently!
 - Problem : Transformation of 1 pixel depending on All neighboring transformation!
 - Interpolation? → **Polyaffine Framework** → smooth transition between transformations at border pixels!
 - **Polyaffine Transformation**
$$T = \sum_{i=1..4} A_i T_i \text{ is valid if } T_i \text{ is „reasonably small“}$$
 - Interpolation using Polyaffine Transformation
 - Using the Log-Euclidean framework
→ compute “small” transformations & interpolate between them iteratively
 - **Input** : Transformation T_1, T_2 / Transformation center c_1, c_2 / Point p
 - **Output** : Interpolated transformation T
- ⇒ **Scaling step** → **Exponentiation step** → **Iterative interpolation**

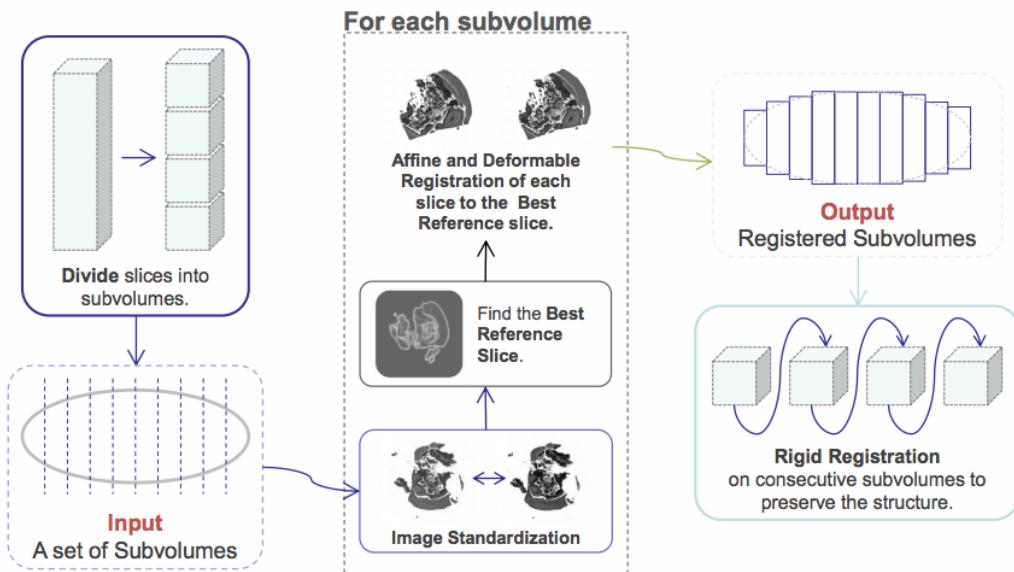
$$\hat{T}_i = \frac{\log(T_i)}{N} \quad \tilde{T}_i = e^{\hat{T}_i}$$

for i=1 to N

$$p = p + w_1(\tilde{T}_1 p' - p') + w_2 (\tilde{T}_2 p' - p')$$

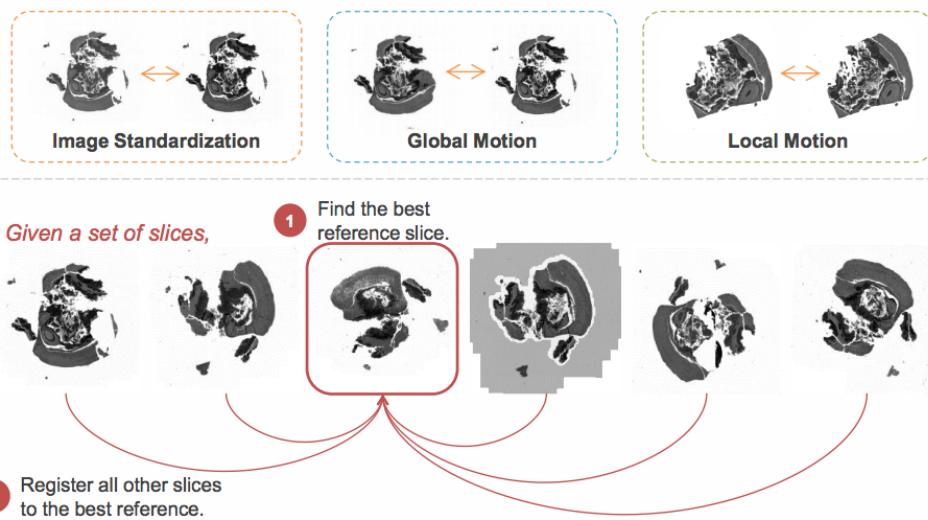
end for

- **Divide Image into Patches → Compute transformation for each patch individually → Transform entire image using polyaffine transform**
- **Reconstruction of Image Stacks**
 - Sometimes it's important to analyse anatomy in 3D on a micron level
 - Reconstruction of Section Data
 - Given an (ordered) stack of section images!
 - Re-stack images → build geometrically coherent volume (pair-wise registration between neighbor slices)
 - Problems:
 - Tissue deforms during sectioning
 - Corrupted slices hamper reconstruction
 - Error propagation → Avoid Drift

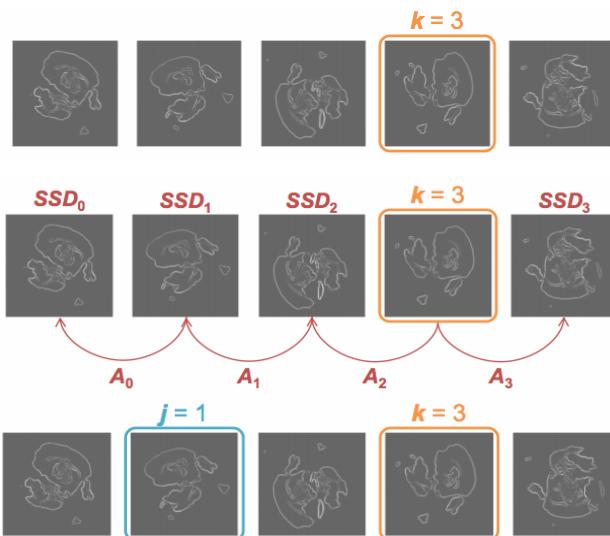
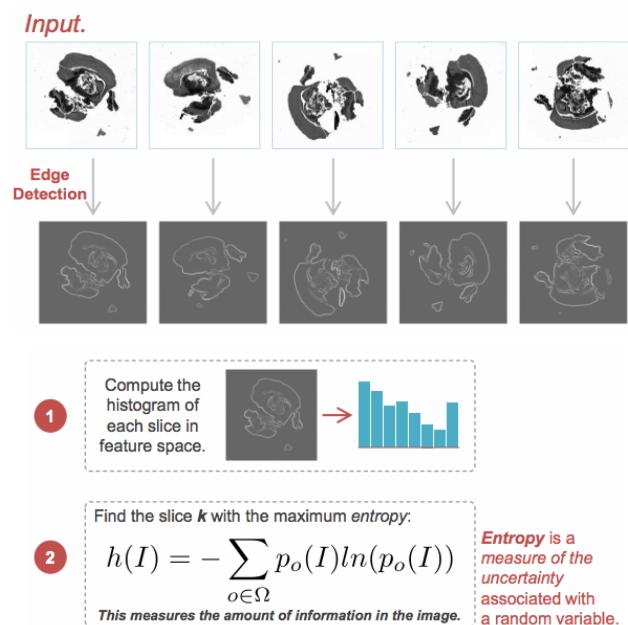
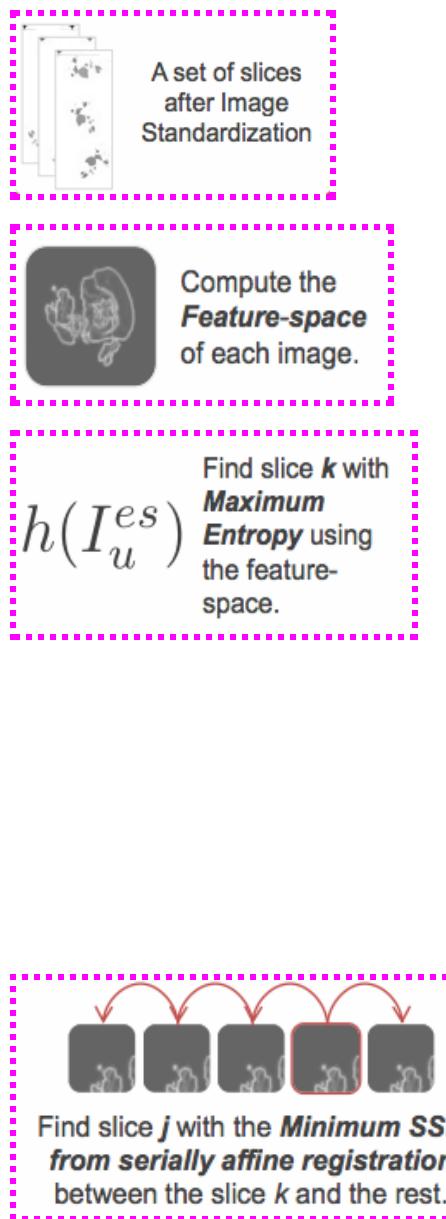


Algorithm

Overview



Best Reference Slice Selection



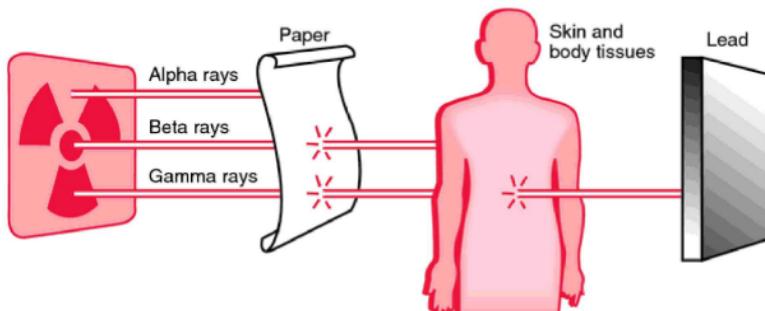
Biomedical Physics

- Key words:
 - Dose **D**
 - The absorption of radiation energy by matter (Unit : Gray)
 - $1\text{Gy} = 1\text{J/kg}$
 - Equivalent Dose **H**
 - Measure the health effect of absorbed radiation by weighting it with a dimensionless factor (Unit : Sievert)
 - $H = qxD$

Radioprotection Units

Ionising radiation - Protection dose quantities in SI units					
Quantity	Absorbed dose D_T	W_R	Equivalent dose H_T	Effective dose E	
SI unit or modifier	gray (Gy)	Radiation weighting Factor - W_R	sievert (Sv)	Effective dose to all tissue $= E$	Effective dose to tissue T_1 $= E$
Derivation	joule/kg	Dimensionless factor	joule/kg	Dimensionless factor	joule/kg
Meaning	Energy absorbed by irradiated sample of matter - a physical quantity.		Biological effect on whole body uniformly irradiated by radiation type R with weighting factor W_R . Multiple radiation types require calculation for each, which are then summated.	Biological effect on tissue type T having weighting factor W_T . Partial irradiation Effective dose = summation of effective doses to those parts irradiated	Complete (uniform) irradiation If whole body irradiated uniformly, the weightings W_T summate to 1. Therefore, Effective dose = Equivalent dose

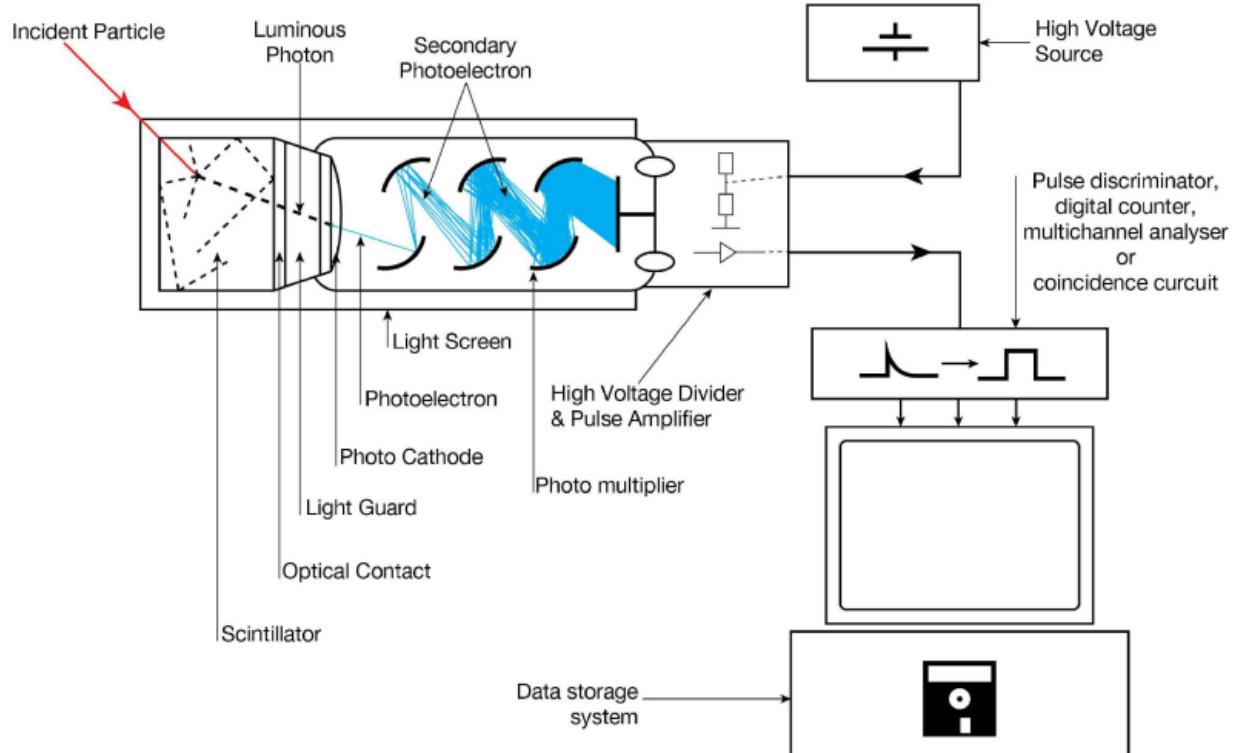
- Penetration
 - alpha particles → few cm of air, a sheet of paper
 - beta particles → few mm of aluminium
 - gamma rays → depends on the energy



Radiation Detection

- Gaseous Ionization Detectors
- Track detector, Microchannel plate detectors, Neutron detectors
- **Scintillating Crystal Detectors**
- Semiconductor Detectors

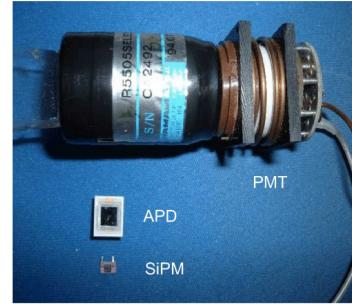
- **Scintillation Counter (with a PMT)**



- **Scintillating Crystal**

- Mechanism :
 - energy deposition by ionization
 - energy transfer to impurities
 - emission of scintillation photons
- Unwanted Effects:
 - Trapping (emission delay)
 - Quenching (light loss)
- Time constants:
 - recombination from activation center : ps... μ s
 - recombination from trapping : ms...s
- **Idea Scintillator → High density & High light yield & Fast decay time & Fast rise time & Peak emission adapted to photodetector!**

- **Photodiodes** : a semiconductor light sensor → generate current or voltage when the p-n junction is illuminated by photons
 - **PMT** → PhotoMultiplier Tubes
 - old technology
 - bulky
 - very sensitive to magnetic fields
 - **Silicon : APD, SiPM**
- **Avalanche Photodiodes (APD)**
 - Principle : U_bias → width of the depletion 消耗 layer
 - decrease the capacitance
 - reduce response time
 - APD:
 - if U_bias Increase
 - secondary electron hole pairs Increase
 - amplification of the signal
 - **Geiger Mode** → if U_bias above a certain threshold
 - break-down voltage 電壓
 - new e-h pair causes an exponential increase in charge
 - making the diode conductive
 - **SPAD (Single Photon Avalanche Diode)**
 - essentially a binary photon detector
 - **SiPM** → **an array of SPADs to detect multiple photons**
- **SiPM (geiger-mode APDs)**
 - **Classical SiPM** : common SPAD readout → created an analogues signal & requires electronics
 - **Digital SiPM** : individual fully digital SPAD readout → single photon counting & optimum timing
- **Semiconductor Detectors**
 - Similar principle to photodiodes
→ radiation photons creates a current or voltage when entering the p-n junction
 - No need Scintillating Crystal!
 - **BUT** : thinner & less dense → lower stopping power → Not suited for high energies!
 - Typical materials: Silicon, Germanium, Diamond, Cadmium



Radionuclide

- Requirement :
 - Clean γ or β emitter (no other emission)
 - energy high enough → reduce interaction with the body!
 - energy low enough → stopped in the detector
 - Half-life Long enough → allow for measurement
 - Half-life short enough → avoid human radiation sources
- Production :
 - Via bombardment 衝擊 of stable atoms with small particles
- Nuclear reactor :
 - Produce long-life “generators”
 - Put stable target into reactor → produce long-life isotopes → ship generator to hospital → long-lived isotopes decay into short-lived ones
 - ex : 99Mo generator (half-life 66 hours) → scintigraphy
99mTC (6 hours) → SPECT
 - Cyclotron : needed locally for short lived isotopes

Radiotracers

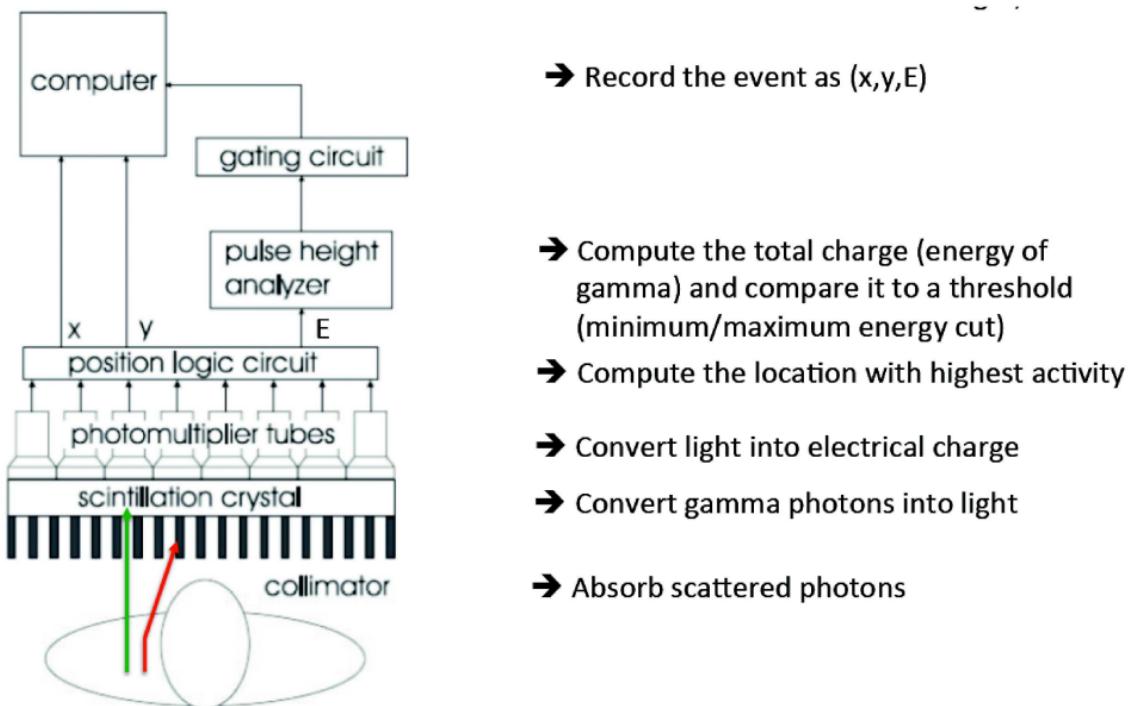
- 3 principles
 - tracer behaves in the human body is known → reproducible fashion
 - tracer does not alter or perturb the system in a measurable way
 - tracer concentration can be measured
- internal Radiotherapy
 - Second principle broken → tracer damages unwanted tissues
- Production : Biochemistry / Radiochemistry

FDG

- A glucose 葡萄糖 molecular where a hydroxyl group is replaced by **18F (110 mins.)**
- Metabolic Principle:
 - cells need energy from glucose
 - cancer wants energy
 - cells cannot make difference between glucose & FDG
 - cancer cell will also consume more FDG than normal cells
 - radioactivity accumulates in the cancer cell!!!!

SPECT

- planar scintigraphy



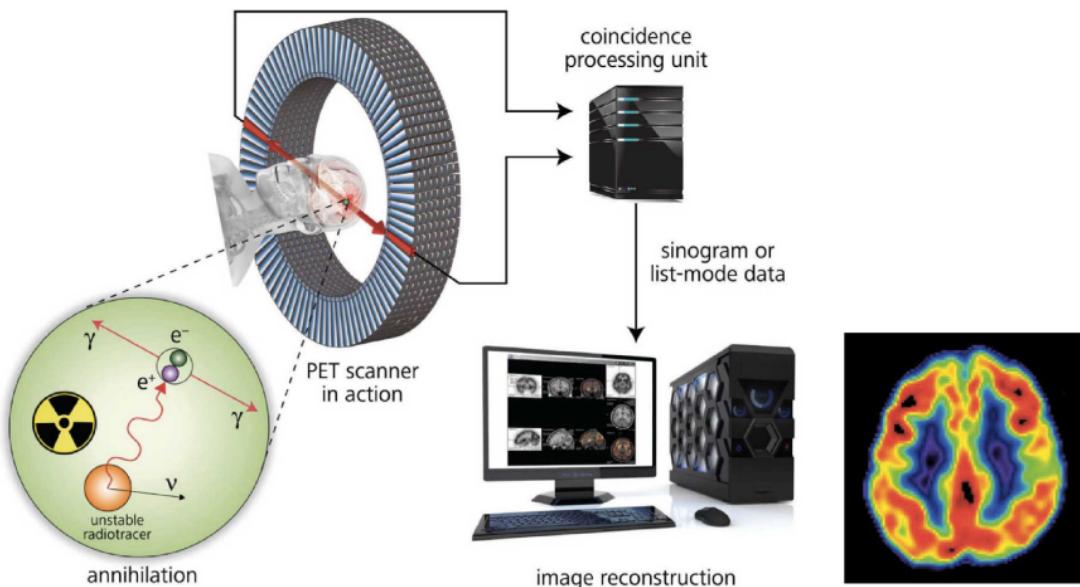
- collimator & detector
 - Collimator
 - By absorbing oblique photons
 - improve resolution
 - reduce sensitivity
 - Types: Parallel holes / converging holes(magnifies) / diverging holes (minifies) / pin-hole
 - Detector → either crystal (NaI) & photodetector / semiconductor
- Electronics
 - Position-logic circuit → find location with highest activity
 - Center of mass of pulse
 - Energy Discrimination :
 - computer total charge E & apply thresholds to reduce unwanted (scattered) photons
- Imaging Assumption
 - no compton scattering
 - lines defined by collimator holes
 - radioactivity stable
 - monoenergetic photons
 - no attenuation
 - Problem : brightness difference of close/far parts of the body!

SPECT → Single Photon Emission Computer Tomography

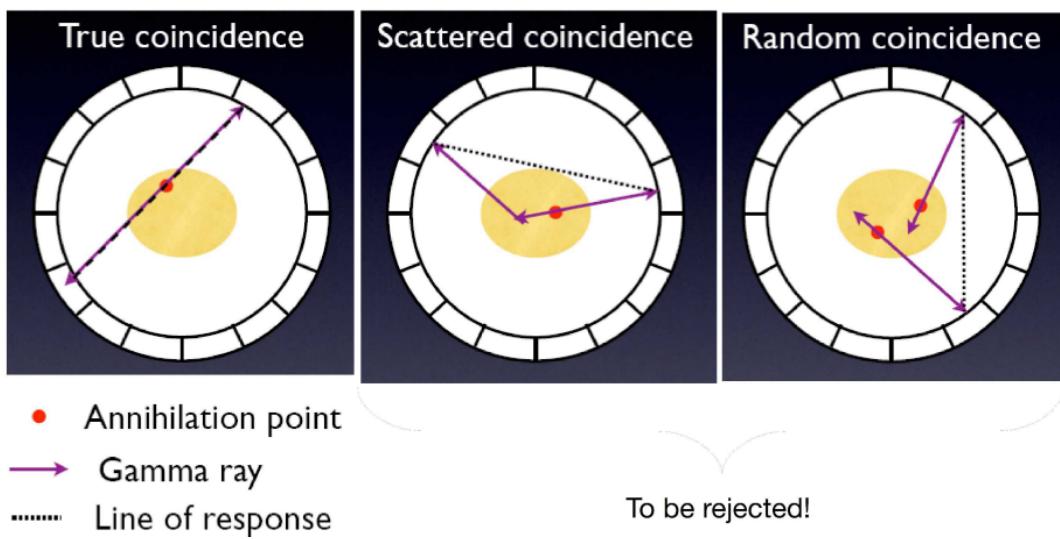
- gamma camera rotates around the volume of interest → acquire several planar scintigraphy & reconstruct 3D image
- Energy : 140keV
- In combination with CT (SPECT/CT) → attenuation correction
 - longer acquisition time!

PET → Positron Emission Tomography

- Positron : The antimatter partner of an electron
- Emission : Positron-emitting radionuclide attached to a tracer injected into the body
- Tomography : imaging by sectioning



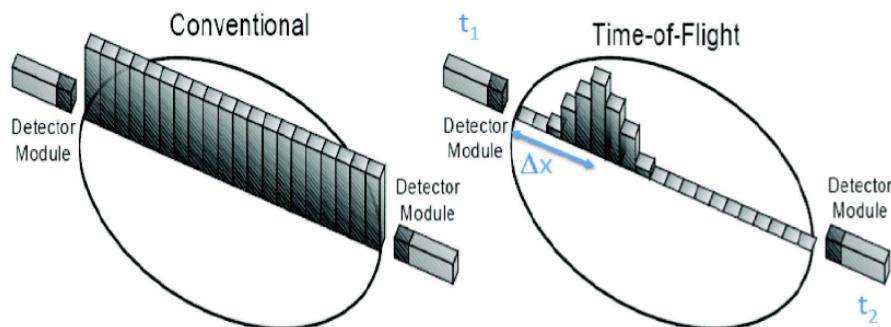
• Types of Coincidences



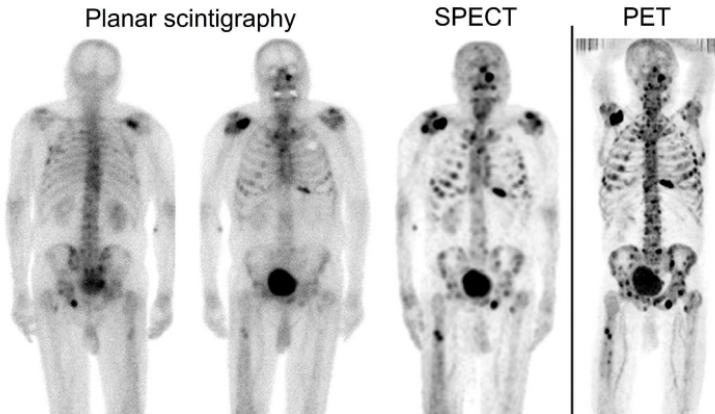
- **Energy** : 511keV → requires scintillating crystals (typically BGO, LOS) with photoconductor
- **Computational complex**
 - compute coincidences between all detector elements
 - reject random by lowering coincidence window
 - reject scattered with good energy discrimination
- **ADP-based Detector**
 - + High spatial resolution
 - + no pile-up
 - + no scattering in the crystal
 - - expensive
 - - many channel
 - - difficult tuning
- **Time-of-Flight**

$$\text{Position determination along the LOR: } \Delta x = c \frac{(t_2 - t_1)}{2}$$

Commercial systems: $\Delta t=500\text{ps}$, equivalent to $\Delta x=7.5\text{cm}$



- **Spatial Resolution**
 - Requirement:
 - small size detectors (high pixelization)
 - individual detectors / “perfect” coding!!
 - Limitation:
 - Technological
 - Crystal size
 - Electronics
 - Physical
 - non-collinearity of double gamma
 - positron range in tissue (1-2mm)
 - Lower limits:
 - clinical whole-body PET : 1.8-2.0mm FWHM (currently 4-6mm)
 - dedicated small systems : 0.7-0.8mm FWHM (currently 1.2-2.0mm)

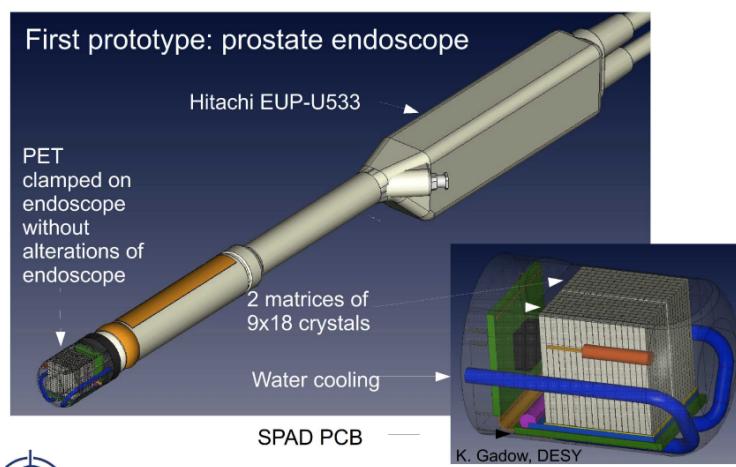


From left to right: posterior and anterior 99mTc-MDP planar scintigraphy, 99mTc-MDP SPECT, and 18F-fluoride PET of a patient with numerous bone metastases.

- **Multimodal imaging**

- combine metabolic imaging (SPECT / PET) → anatomic imaging (CT / MR)
- Intrinsic registration information
- scatter correction

ClearPEM	endoTOFPET-US
<ul style="list-style-type: none"> ● Image resolution : ~1.5 mm (in 3D) ● Detection efficiency > 1% → faster exam & less radiation ● Developed by the Crystal Clear Collaboration <ul style="list-style-type: none"> ○ Breast exams → patient in prone position ○ plate rotate around breast ● Good for Breast tumor visible 	<ul style="list-style-type: none"> ● a novel imaging system → endoscopic exams → pancreas or prostate ● combination with high resolution metabolic imaging TOFPET+US ● development of targeted biomarkers

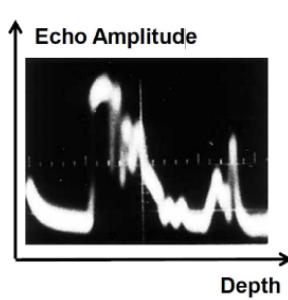


Elastography

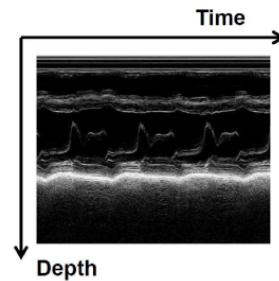
US-Elastography

- What is US?

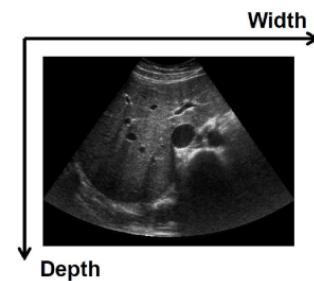
- High frequency sound wave
- US transducer → send short US pulse sequence into body → waves partly reflect partly transmit at multiple depth → same transducer record echoes → signal is process to display
- image acoustic impedance
- differences in acoustic impedance create interfaces reflecting sound
- different mode!



A-Mode: amplitude of reflected echo plotted against depth



M-Mode: echo as gray-values plotted over depth and time



B-mode: echo as gray-values plotted over depth and width

Medical Visualization

Goals: Map, Record, Abstract, Clarify, Interact, communicate

Visualization Pipeline

Acquire	Process	Map	Render
<ul style="list-style-type: none">• Measure• Simulate• Model	<ul style="list-style-type: none">• Filtering• Derive Data• Interpolation	<ul style="list-style-type: none">• Map to optical properties• Map to geometry• Generate renderable data	<ul style="list-style-type: none">• Compositing 合成• Visibility Computation• Lighting

Data Characteristics

- Source : Real world (measuring) → Theory (simulation)
- Domain : Discrete metric → Continuous metric
- Dimensionality : Scalar → Vector → Tensor
- Structure : Regular grid → Scattered data

Volume Visualization

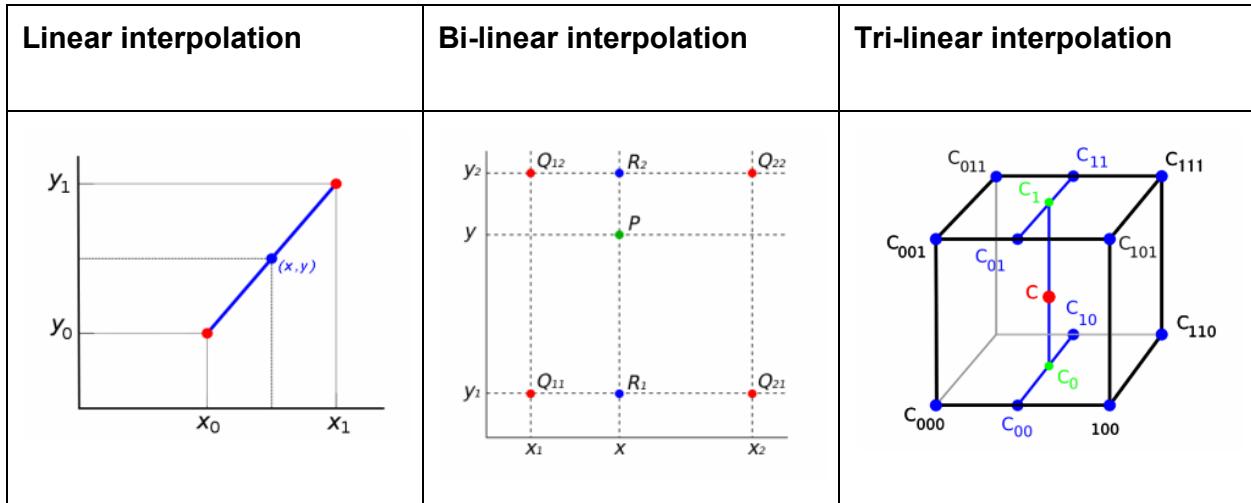
- Bring volumetric 3D data → 2D screen
 - 2D MPR (multi planar reconstruction)
 - Photo-realistic projection
- Information seeking mantra:
 - Overview First
 - Zoom & Filter
 - Detail on Demand

Direct Volume Rendering (image order approach) ←→ Indirect Volume Rendering (object..)

- simulate physics of light transport
- directly project 3D → 2D screen
- render 1 pixel at a time
- **Sampling → Classification → Compositing**
 - each steps work independently
 - Sampling & Classification → may be reversed
 - Pre & post classification (better quality!)

Sampling → Interpolation

- Why do we need interpolation?
 - image data has discrete representation (pixels/ voxels)
 - ray sample may lie between voxels
- Nearest neighbor! → poor quality
- (Bi-/Tri) Linear interpolation → acceptable quality, fast implementation

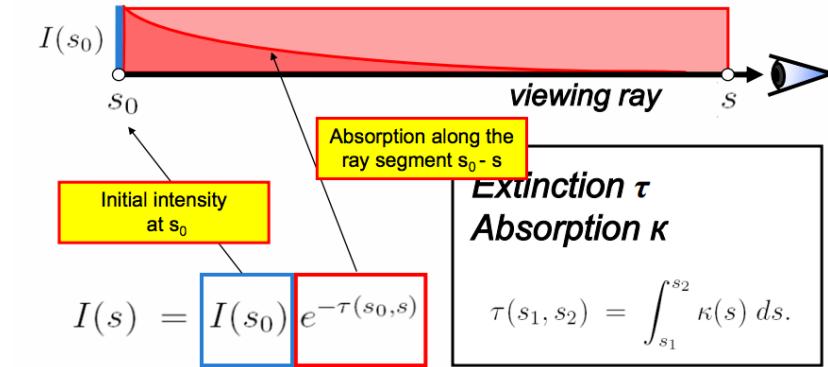


Classification

- medical images usually only have scalar intensities as domain
- How can we define the look of data?
- Transfer Function:
 - simple concept → map **scalar intensities** to optical properties
 - *Scalar intensity* ⇒ *Transfer Function* ⇒ *Color (Emission)* or *Optical (Absorption)*

Compositing

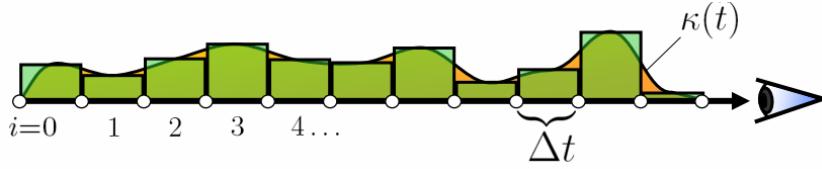
- The heart of direct volume rendering → **Simulate the physics of light transfer**
- **The Volume Rendering Integral**
 - How do we determine the radiant energy along the ray?
 - physical model : emission & absorption, no scattering
 - Without absorption → All the initial radiant energy would reach the point s



Every point \tilde{s} along the viewing Ray emits additional radiant energy

$$I(s) = I(s_0) e^{-\tau(s_0, s)} + \int_{s_0}^s q(\tilde{s}) e^{-\tau(\tilde{s}, s)} d\tilde{s}$$

- Numerical Solution



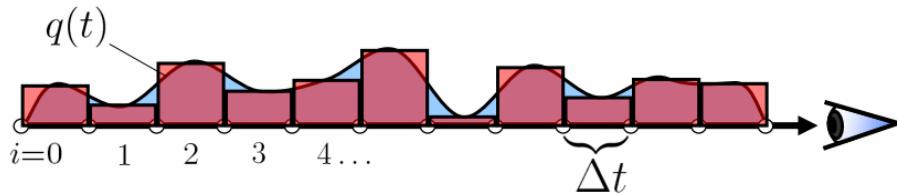
$$\text{Extinction: } \tau(0, t) = \int_0^t \kappa(\hat{t}) d\hat{t}$$

Approximate Integral by Riemann sum:

$$\tau(0, t) \approx \sum_{i=0}^{\lfloor t/\Delta t \rfloor} \kappa(i \cdot \Delta t) \Delta t$$

- introduce the opacity:

$$A_i = 1 - e^{-\kappa(i \cdot \Delta t) \Delta t}$$



$$\tilde{C} = \sum_{i=0}^{\lfloor T/\Delta t \rfloor} C_i \prod_{j=0}^{i-1} (1 - A_j)$$

can be computed recursively

$$C'_i = C_i + (1 - A_i)C'_{i-1}$$

Radiant energy observed at position i	Radiant energy emitted at position i	Absorption at position i	Radiant energy observed at position $i-1$
---	--	----------------------------	---

- Back-to-Front compositing :

$$C'_i = C_i + (1 - A_i)C'_{i-1}$$

- Front-to-Back composition :

$$C'_i = C'_{i+1} + (1 - A'_{i+1})C_i$$

$$A'_i = A'_{i+1} + (1 - A'_{i+1})A_i$$

Besides Photo-realistic DVR

The flexible of the Volume Rendering Pipeline allows for all kinds of visualization :

- **MIP** (maximum intensity projection)
 - Classification : Identity function
 - Compositing : Accumulate maximum intensity
- **DRR** (Digitally Reconstructed Radiograph)
 - Classification : Compute x-ray attenuation
 - Compositing : Integrate x-ray attenuation
- **Focus- & Context Rendering**
 - Classification : Determine additional importance value
 - Compositing : Relevance-based integration

Perception 知覺

- Eye : broadband channel to the mind
 - ~80% of our environment is perceived via our eyes
 - ~50% of the human brain deals with processing the visual input
- Perception / cognition : complex system
- understanding of how we work → Design better visualization

Visual Processing

- start parallel & continuous sequentially:
 - parallel processing of properties : Orientation, color, movement
 - detection of 2D → Patterns, Contours, Regions
 - object identification & visual working memory!
- interaction responsiveness
 - ~0.1 sec → motion, cause&effect
 - ~1 sec → unprepare response
 - ~10 sec → routine cognitive task

Pre-Attentive Processing 預知

- Iconic Memory
 - short-live visual buffer
 - hold image for 1-2 secs prior to transfer to visual working memory
- Pre-Attentive processing
 - Mechanism underlying pop-out
 - occurs prior to conscious attention
 - visual identification within ~200ms

Color Perception

- Perceptual dimensions of colors
 - **Hue**: color of the rainbow (wavelength)
 - **Saturation** 鮑和度: chromaticity/paleness (spectral distribution光譜分佈)
 - **Intensity**: lightness/darkness (amount of light entering the eye)

The Usage of Color Type

- The optimal color table depends on the data type
 - Normal Data → A selection of distinct colors
 - High frequency ordinal data → Luminance 亮度 contrast color map
(eg. heated body scale)
 - Colormap on surface → isoluminant colormap to preserve perceived shape
 - Interval & Ratio data
 - single colormap
 - diverging colormap
 - **Rainbow colormap**
 - misleading → 1st column : appearing bands are not present in the data
→ 2nd column : sharp gradient is not perceivable
 - missordering