Project Teams

- **EARTH**: Aryaman, Bhargava, Khadija, Weiting, Young
- **WIND**: Farhan, Kimon, Marshall, San-Yu, Xuefeng
- **FIRE**: Arjun, Ekarat, Keon, Peter, Vincent
- **WATER**: Brandan, Devin, Konstantinos, Miguel, Sicheng, Umid
- **SKY**: Abby, Ava, Feras, Ivana, Jules, Malieka

*Initial Team Set-Up Tasks*
- Have a team meeting (first during breaks in this lecture)
- Set up regular meeting schedules
- Decide
  - A common OS/IDE to be used
  - C++ vs Python (ITK vs SimpleITK)
  - how to share codes (Git etc).
- Make sure all can run ITK/VTK/FLTK (Check everyone’s ITKs and help each other to set them up for all if there are any remaining issues)

Project Design: Due in one week!

- Study resources in the assignment following links.
- Discuss about the overall goal/focus of your team
- Segmentation and Registration
  - Choose one or a few algorithms for each component
  - Divide your team into two subgroups and assign them to the two tasks.
- Quantification
  - Choose three or more tasks from the assignment.
  - Divide your team into three or more subgroups and assign them to the chosen quantification tasks.

- Each team is to submit a design plan (goals, chosen algo/tasks, personnel assignments, milestones, platforms, meeting style) by next Tuesday for my approval. Use Canvas: [Project Planning & Management]
## Project Planning Tips

- Use existing ITK functions/apps as much as possible.
- Use existing viewer (ImageJ, ITK-snap) as much as possible to focus on algorithms and their experiments.
- Allocate enough time for integration and performance evaluation experiments (More Research than Development).
- Allocate enough time for preparing group presentation.
- Make sure to report individual task assignments in your team’s project plan submitted in Canvas forum.
- The plan is working documents. **You can revise it over the course of your project.** Consult me when any issue arises asap.

---

### Midterm #1 in Two Weeks

- 100 min for 6 questions from Lec 1-5 (Tight)
- Open handwritten notes (start to prepare!)
- Zoom Proctoring (make sure your video work)
- iLearn Submission: you scan your answers written on your own papers by a scan-app of your choice then upload it to iLearn link. Tablet/iPad not allowed.
Image Data Structure & Visualization

CSC621-821
Biomedical Imaging and Analysis
Dr. Kazunori Okada

Overview

• Last lecture: Imaging Methods & Physics II
  – MRI
  – PET
  – US
  – PET-CT

• Today’s lecture
  – Image Data Structure
  – Image File Formats & Visualization

Image Formation
Data Structure of Images: Overview

- How to organize numbers encoding all information present in an image?
- Graphical/Numerical Representation of Images
- Standard required because
  - Viewing
  - Database
  - Algorithm Development

Image as Spatial Lattice and Feature

Image = Feature annotated lattice
Spatial Coordinates

\[ x_{ij} = (i, j) \]

\[ N_i \]

\[ N_j \]

Spatial Dimension

1D

\[ x_i = (i) \]

2D

\[ x_{ij} = (i, j) \]

3D

\[ x_{ijk} = (i, j, k) \]

\[ x = (i, j, k, \ldots, z) \]

\[ N \]
Pixel and Voxel: Image Unit

**Pixel** is a feature-annotated node of 2D image lattice

**Voxel** is a feature-annotated node of 3D image lattice

\[ x_{ij} = (i, j) \quad I(x_{ij}) \]

- **Intensity**
- **Feature**

\[ x_{ijk} = (i, j, k) \quad I(x_{ijk}) \]

Raster Scan

- Successively scanning each lattice node for entire image

```c
for (j=0;j<N;j++) {
    for (i=0;i<N;i++) {
        ...
    }
}
```

**Double loop!**

```c`
```
Temporal Dimension

- 1D space: independent (perpendicular) to the spatial dimensions
  - Video is 3D data: (2D + time)
  - 4D imaging: (3D + time, 3D video)

Types of Intensity Feature

- **Scalar**: 8-bit Grayscale (1D, 256 possible tones)
  \[ (v) = I(x_{ij}) \]

- **Scalar**: 1-bit Binary (1D, 2 possible values)
  \[ (v) = I(x_{ij}) \in \{0, 255\} \]

- **Vector**: 24-bit Color Feature (3D) \( (r, g, b) = I(x_{ij}) \)
Types of Vector Feature

- **Vector**: Spatial-Intensity Feature (3D)
  \[ (i, j, v) = I(x_{ij}) \]
  Spatial coordinates can also be a feature!

- **Vector**: Intensity-Time Feature (2D)
  \[ (v, t) = I(x_{ij}) \]
  Video!

- **Vector**: Multispectral feature (n-D)
  \[ (v_1, v_2, ..., v_n) = I(x_{ij}) \]
  Wavelength

- **Matrix**: Covariance Feature (m x m-D)
  \[
  \begin{pmatrix}
  c_{11} & c_{21} \\
  c_{12} & c_{22}
  \end{pmatrix} = I(x_{ij})
  \]

---

Image Matrix

- **3D Matrix** \( A_{ijk} \)
- **Color Image Matrices** \( (A_{ij1}, A_{ij2}, A_{ij3}) \)
- **ND Matrix** \( A_{i,j,k,...,z} \) Tensor
Nature of 2D Image: Projection

- 3D-to-2D projection causes **information loss**
- Reflected Lights
  - Only viewing the visible surface
  - Behind mountain?
- Refracted Lights
  - Superimposition over depth
  - Depth information lost

Nature of 3D Image: Volumetric Image

- “3D volume” as spatial coordinates
- No dimensional reduction $\text{3D} \rightarrow \text{3D}$
  - The world is in 3D
  - Image is also 3D
  - No apparent loss of depth information!
  - Because of this it is hard to view.
Continuous Function as Image

- Another formulation
- Let a 2D continuous function model a 2D image

\[ f(x) : \mathbb{R}^{2} \rightarrow \mathbb{R} \]

\( x \in \mathbb{R}^{2} \): spatial coordinate vector \((x, y)\)

\( f \in \mathbb{R} \): function response

N-Dimensional Vector Space

\[ f(x) : \mathbb{R}^{2} \rightarrow \mathbb{R} \quad f(x) : \mathbb{R}^{3} \rightarrow \mathbb{R} \quad f(x) : \mathbb{R}^{N} \rightarrow \mathbb{R} \]

Number of Axis = Spatial Dimensionality of Image
**Digital Image from Analog Model**

- **Analog Image**
  - $\mathbf{x} \in \mathbb{R}^n$
  - Continuous space

- **Digital Image**
  - Discretization
  - Quantization
  - Continuous response
  - Discrete space

**Discretization of Image**

- Approximating the continuous function input by a discrete and finite set of numbers (regular sampling)

- # of possible locations: $\infty$

- # of possible locations: $N_x \times N_y$

$$i = 0, \ldots, N_x - 1$$

$$j = 0, \ldots, N_y - 1$$
Spatial Resolution

- Smallest discernible detail in an image
  - DPI: dots per inch: how many dots/pixels within an unit area?

\[
\begin{array}{ccc}
1024 \times 1024 & 512 \times 512 & 256 \times 256 \\
64 \times 64 & 32 \times 32
\end{array}
\]

Spatial Resolution

- \( N_x \times N_y \): Width by Height: The number of pixels
- \( N \) megapixels: \( N = N_x \times N_y / \text{(one million)} \)
Quantization of Image

Approximating the continuous function output by a discrete and finite set of numbers

Example: rounding a real number in the interval [0, 100] into an integer 0, 1, 2, ..., 100

Intensity Resolution

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

<table>
<thead>
<tr>
<th>Number of Bits</th>
<th>Number of Intensity Levels</th>
<th>Examples</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2</td>
<td>0, 1</td>
</tr>
<tr>
<td>2</td>
<td>4</td>
<td>00, 01, 10, 11</td>
</tr>
<tr>
<td>4</td>
<td>16</td>
<td>0000, 0101, 1111</td>
</tr>
<tr>
<td>8</td>
<td>256</td>
<td>00110011, 01010101</td>
</tr>
<tr>
<td>16</td>
<td>65,536</td>
<td>1010101010101010</td>
</tr>
</tbody>
</table>
### Intensity Resolution Example

<table>
<thead>
<tr>
<th>256 grey levels (8 bits per pixel)</th>
<th>128 grey levels (7 bpp)</th>
<th>64 grey levels (6 bpp)</th>
<th>32 grey levels (5 bpp)</th>
</tr>
</thead>
<tbody>
<tr>
<td>![Skull Image]</td>
<td>![Skull Image]</td>
<td>![Skull Image]</td>
<td>![Skull Image]</td>
</tr>
<tr>
<td>![Skull Image]</td>
<td>![Skull Image]</td>
<td>![Skull Image]</td>
<td>![Skull Image]</td>
</tr>
<tr>
<td>![Skull Image]</td>
<td>![Skull Image]</td>
<td>![Skull Image]</td>
<td>![Skull Image]</td>
</tr>
</tbody>
</table>

---

### Grayscale

- **Successive tone levels from black to white**
- Black = 0
- White = Max#
- **8-bit grayscale**
  - 256 levels [0, 255]
  - computer display
- **12-bit grayscale**
  - 4096 levels [0, 4095]
  - CT scans
- **16-bit grayscale**
  - 65,536 levels [0, 65,535]
  - PET scans
Medical Image Features (BREAK)

• CT numbers
  – Intensity value for CT scan with Hounsfield Unit normalized with the attenuation coefficient of water
  – Quantized (Translated and Scaled) to [0, 4095] range

\[
\text{HU} = \frac{\mu - \mu_{\text{water}}}{\mu_{\text{water}}} \times 1000 + 1000
\]

• SUV: standard uptake value
  – Semi-standardized tissue activity measure for PET normalized over injected dose and body weight
  \[
  
  SUV = \frac{c_{\text{PET}}(t)}{ID/BW}
  \]
  \( ID: \) injected dose
  \( BW: \) body weight
  \( c_{\text{PET}}(t): \) measured radioactive concentration at time \( t \)

Color

• Truecolor (RGB color)
  – Direct Color, typically 3D feature
  – 24 bit = 8-bit Red + 8-bit Green + 8-bit Blue
  – Total of 16.7 million different colors
  – Old systems: 8-bit direct, 12-bit direct, HighColor

• Indexed Color (Pointer!)
  – Fixed-size color palette ~ 16 bit
  – Each image pixel contains an index to the palette
  – Adjustable Palette (Pseudo Color/Color Quantization)
Color Spaces: RGB / CMYK

**RGB Space**
- Red/Green/Blue
- Additive
- TV Monitor

**CMYK Space**
- Cyan/Magenta/Yellow/Black
- Subtractive
- Printing

---

Color Spaces: HSV / YUV

**HSV Space**
- Hue/Saturation/Value
- Cylindrical Graphics

**YUV Space**
- Luma(Y)/Chrominance(UV)
- NTSC/Composite Color
Color Space: Human Vision

CIE 1931 XYZ Space
Mathematically derived from human experiments
Chromaticity-Brightness

CIE L*a*b* Space
Non-linearly compressed CIE XYZ space to achieve better Perceptual uniformity

Color Quantization

- Reduce the total number of colors by using a set of smaller representative colors with minimum impact on appearance
  - Compression for reducing file size
  - Intensity Quantization + Indexed Color (Palette)
  - Clustering of Color Space
Color Quantization Example

24 bit

RGB Color to Grayscale

- 24-Bit RGB color (R,G,B)
- 8-Bit Grayscale (I)
- Grayscale is along the diagonal line (R=G=B) of the RGB color space
- RGB color can be transformed to Grayscale value by

\[ I = \text{round}\left(\frac{R+G+B}{3}\right) \]
Data File Formats: Overview

• Many existing image file formats to handle all these variations in image types
  – Brief overview of 2D Image File Formats
  – Brief overview of Video File Formats
  – 3D image file formats
  – DICOM
  – Analyze

2D Image File Formats

• **JPEG** (Joint Photographic Experts Group)
  – Most common on WWW
• **BMP**
  – Bitmap (storing pixel features in the raster order in binary format)
  – Used mainly by Microsoft
• **GIF** (Graphics Interchange Format)
  – Limited 8-bit color, GIF animation
• **PNG** (Portable Network Graphics)
  – Replacing GIF due to Unisys patent claim
• **TIFF** (Tagged Image File Format)
  – Suitable for high-quality color image
  – Flexible
Compressions

Reducing the size of image files

• Lossy Compression
  – Loses some picture details by compression
  – But achieve highest rate of compression
  – JPEG

• Lossless Compression
  – Does not lose any picture information
  – DEFLATE Algorithm: PNG
  – LZW Algorithm: GIF, TIFF

• Uncompressed (Better for Image Processing!)
  – No compression is performed
  – BMP
  – TIFF

Tags

• Store non-image information such as
  – Size
  – Definition
  – Image-data arrangement
  – Compression types
  – Any other geometric information

• Makes TIFF format very flexible


### 3D Image File Formats

- **2D Video (2D Image + 1D time)**
  - mpeg (moving picture experts group), lossy compression
  - wmv (windows media video), lossy compression, Microsoft
  - avi (audio video interleave), container format by Microsoft
  - mov (quick-time) container format by Apple
  - mp4, 3gp, divx, rm, etc

- **Row 3D volume data by a 2D stack**
- **File Format Standards specific to Medical Images**

---

### Volumetric Image as 2D Stack

Row 3D volume data can be stored in

- **Binary bitmap (with a small header information)**
  - Matlab
  - You can do this from your code

- **2D Image Stack**
  - A set of 2D images in a 2D image file format with a header or file list specifying z-coordinates
  - Z-coord in filename
  - Z-coord in a list file of filenames (Header file)
Medical Image File Format Standards

- Medical Imaging Standards include Specific File Formats
  - DICOM
  - Analyze
  - More…

- Tags (introduced in TIFF) is used heavily for storing various information about clinical patients, hospitals, imaging protocols and parameters

DICOM Overview

- Digital Imaging and Communications in Medicine
  - Standard for handling, storing, printing and transmitting medical imaging information, including…
  - File format definition
  - Communication Protocol (TCP/IP)
  - Copy-righted by NEMA (National Electrical Manufacturers Association)
  - [http://medical.nema.org/](http://medical.nema.org/)
### DICOM file

- DICOM data consists of a set of DICOM files
- **(dcm)** DICOM file extension
- DICOM file contains all data as **tags**
  - Image Data in binary form
  - Tags for IDs
  - Tags for patient information
  - Tags for image geometry etc
- Data is organized by “**Study**”
- “**Series**”: different type scans in a study
- **Patient ID** is not unique but **Study ID** is

---

### DICOM Tags

<table>
<thead>
<tr>
<th>Tag Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Study/Series IDs</td>
<td></td>
</tr>
<tr>
<td>Transfer Syntax UID</td>
<td></td>
</tr>
<tr>
<td>Modality: CT/MR/PT/US</td>
<td></td>
</tr>
<tr>
<td>Slice Thickness: in mm</td>
<td></td>
</tr>
<tr>
<td>Pixel Spacing: pixel size in mm</td>
<td></td>
</tr>
<tr>
<td>Image Location: 3D coordinates</td>
<td></td>
</tr>
<tr>
<td>Image Orientation</td>
<td></td>
</tr>
<tr>
<td>Rows: Ny</td>
<td></td>
</tr>
<tr>
<td>Columns: Nx</td>
<td></td>
</tr>
<tr>
<td><strong>Bits Stored</strong>: 12–16 bit</td>
<td></td>
</tr>
<tr>
<td>Rescale Intercept</td>
<td></td>
</tr>
<tr>
<td>Rescale Slope</td>
<td></td>
</tr>
<tr>
<td>Window Center</td>
<td></td>
</tr>
<tr>
<td>Window Width</td>
<td></td>
</tr>
</tbody>
</table>

CT# to intensity windowing

- **128 bytes** used by DICOM format followed by the character “DICOM”
- This preamble is followed by extra information e.g.:
  - `0002.0001.File Meta Info Version: 255`
  - `0002.0010.Transfer Syntax UID: 1.2.840.10008.1.2.1.3.1.1.4.1.1`
  - `0008.0000.Lesion Length: 152`
  - `0038.0050.Modality: MR`
  - `0010.0010.Manufacturer: MR Image`
  - `0010.0000.Manufacturer: Siemens`
  - `0010.0010.Manufacturer: Philips`
  - `0028.0000.Image Presentation Group Length: 148`
  - `0028.0002.Slice Per Pixel: 1`
  - `0028.0004.Photometric Interpretation: MONOCHROMATIC2`
  - `0028.0000.Pixel Data: 2`
  - `0038.0010.Rows: 1100`
  - `0038.0010.Columns: 1100`
  - `0038.0002.Pixel Spacing: 2.00200`
  - `0038.0002.Pixel Spacing: 2.00200`
  - `0038.0002.Window Center: 0.0000`
  - `0038.0002.Window Width: 32002.00`
  - `0038.0002.Window Data: 19538`
**DICOM Header**

- A header file associated with a set of DICOM files for a study
- Includes basic information of the study
  - Study ID
  - Key attributes
  - Information of DICOM software
- Some DICOM file viewing software may require this file. Some not (e.g., ImageJ)

**Analyze**

- Developed at Mayo Clinic (leading medical research)
- Used heavily in functional MRI neuroimaging
  - SPM, FreeSurfer etc
- Two files for one volume
  - `.img`: volumetric image data in a binary format
  - `.hdr`: other image-related information such as voxel size and the number of voxels and slices etc
- Viewing Analyze format image
  - MRIcro (freeware)
- Converting DICOM to Analyze Format
  - MRIConvert
  - [http://lcni.uoregon.edu/~jolinda/MRIConvert/](http://lcni.uoregon.edu/~jolinda/MRIConvert/)
Data Variations: CT/MRI/PET

- **CT**
  - Slice dimensions: 512 x 512
  - Slice thickness: >0.5mm
  - Intensity: 12 bit

- **MRI**
  - Slice dimensions: 512 x 512
  - Slice thickness: 2mm for 1.5T, 1mm for 3T
  - Intensity: 16 bit

- **PET**
  - Slice dimensions: 256 x 256
  - Slice thickness: 1 ~ 5 mm
  - Intensity: 16 bit
  - FWHM Resolution: 4 ~ 8 mm

- Whole-Body Scan has larger slice thickness

PACS Systems (Break)

- **Picture Archiving and Communication System**
  - Software system designed to manage medical images using DICOM standard
  - Paperless, Remote access, Electronic image integration, Radiology workflow management

- **EHR (Electronic Health Records)**
  - Currently developed in various hospitals
  - Local system without standards
  - OpenMRS
  - PACS can be integrated into EHR?
  - VistA: Veterans Health Information Systems and Technology Architecture (at the VA hospitals)
Image Visualization: Overview

• How to display these biomedical images on your computer screen?
  – 2D Images: Simple (2D image on 2D display)
  – 3D Images: Difficult (3D image on 2D display!)

• Particularly for biomedical imaging application, visualization is critical!
  – Measurement accuracy is demanded
  – Radiologists use human-eye observation and reasoning as their main diagnostic tool and they are considered to be the best “system” around

2D Viewers

• Various image viewing software available for generic image formats
• Many are freeware
  – ImageMagick
  – IrfanView
  – Gimp

• But cannot be used to view medical images…
ImageMagick

- Read, Convert, Compose images

IrfanView

- IrfanView: Simple/Screen capture function
- http://www.irfanview.com/
Gimp

- GNU Image Manipulation Program: Photoshop like
- http://www.gimp.org/

3D Visualization: Overview

- Viewers must be compatible for biomedical image format standards
- Special data processing is required to display 3D images on 2D monitor screen
  - Intensity Windowing
  - Axial View
  - Cine
  - MPR: Multi-Planar Reconstruction
  - MIP: Maximum Intensity Projection
  - Volume Rendering
DICOM Viewers

- Image Viewers compatible to DICOM format images and volumes
- Open-Source Viewers
  - Clear Canvas: [http://www.clearcanvas.ca/dnn/](http://www.clearcanvas.ca/dnn/)
  - 3D Slicer: [http://www.slicer.org](http://www.slicer.org)
- Matlab

ImageJ & Fiji

- Public domain, Java-based biomedical image processing
- Developed at National Institute of Health
- Extendable via Java plugins
Intensity Windowing

- Contrast enhancement for specific organs of interests
- Computer displays only 8-bit gray levels but our data is 12-16 bit

Slice Plane Convention

Sagittal View

Axial View

Coronal View
Axial View

- Traditionally radiologists are trained to study this view
- Computer screen or on a film

Cine Display & Imaging

- Successive slices can be displayed as a movie in order to process a large number of slices
- Cine Imaging
  - Cardiac MRI with ECG gating
  - Wall motion and blood flow analysis
MPR: Multi-planar Reconstruction

- Three orthogonal views (sagittal, coronal, axial)

Plane Re-Sampling

- Slice thickness re-sampling (re-slicing)
  - A voxel area corresponds to a non-square region of patient
  - Rearrange the data into a regular lattice with isotropic voxels by interpolation

- Non-orthogonal & Curved plane MPR
  - Oblique slice plane
  - Curved slice plane along for vessels to straighten it out
**Plane Re-Sampling via Interpolation**

2D Example

- Introduce a new arbitrary lattice onto the current image
- Compute intensity at each lattice node by **interpolating** neighboring pixel values

**Bilinear Interpolation**

Compute \( I(Q) \) given all point locations and \( I(P_{11}), I(P_{12}), I(P_{21}), I(P_{22}) \)

\[
I(Q) = \frac{y_2 - y}{y_2 - y_1} I(R_1) + \frac{y - y_1}{y_2 - y_1} I(R_2) \\
= \frac{(x_2 - x)(y - y_1)}{(x_2 - x_1)(y_2 - y_1)} I(P_{11}) \\
+ \frac{(x - x_1)(y - y_1)}{(x_2 - x_1)(y_2 - y_1)} I(P_{21}) \\
+ \frac{(x_2 - x)(y_2 - y)}{(x_2 - x_1)(y_2 - y_1)} I(P_{12}) \\
+ \frac{(x - x_1)(y_2 - y)}{(x_2 - x_1)(y_2 - y_1)} I(P_{22})
\]

Coeff is the ratio of the area!
Bilinear Interpolation Cond.

- Interpolation function $I(Q)=I(x,y)$ is a quadratic function (non-linear)
  \[
  I(x, y) = (a_1 x + a_2)(a_3 y + a_4)
  \]
  \[
  I(x, y) = b_1 + b_2 x + b_3 y + b_4 x y
  \]

- For 3D, you can extend this to “trilinear interpolation” Derive this as a home exercise!

- For smoother interpolation, there are other functions
  - Polynomial
  - Cubic Spline
  - Radial Basis Function
  - B-Spline
  - Thin Plate Spline

Surface Rendering

1. Do 3D target segmentation
2. Render the 3D geometric surface by any CG tool
MIP: Maximum Intensity Projection

- Project the intensity value maximum along the projection direction
- By rotating the projection plane around the target object, one can observe 3D depth
- No segmentation required
- Does not work for very dense targets
- Used in MRA and PET studies

Volume Rendering

- Generalized MIP by assigning each voxel “opacity” (the visibility weight) and/or color
- Assign high opacity only for voxels on target areas
- RGBA: (red, green, blue, alpha)
  - Alpha channel used for opacity
  - 0% transparent
  - 100% opaque
- Volume Ray Casting
  - Projects 3D volumes in RGBA to 2D RGB image
Volume Rendering Examples

Summary

- Image Data Structure
- Image Visualization
- Next Week:
  - Practical Foundation of Digital Image Processing I
- Homework Exercise/Project
  - Try trilinear interpolation at home.
  - Project Design Due in one week by Tuesday Sep.19...
  - Please submit each team’s plan using Canvas’s Discussion.
- Midterm #1 in two weeks. Covers materials from the second to the next lecture. (start to review now & make your hand-written notes)