# Preliminaries - MATLAB

indexing arrays (linear vs ij)

xy vs ij - coordinates for arrays and imagens

concatenation

I(:), index vs for loops

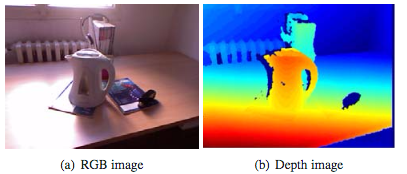
# LAB Topic 1 (2 classes)

# RGB+D Cameras: Image, 3D point clouds and camera model

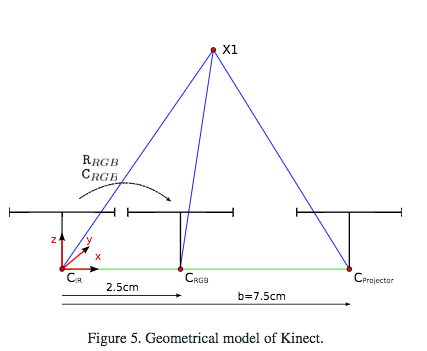
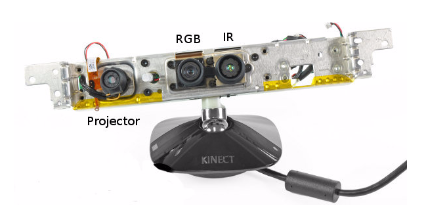


Suggested readings

"Kinect" (depth) cameras provide 2 type of images: RGB and DEPTH



Q: What coordinate is the "depth" ? How is it obtained ?



Tasks:

1. See RGB and Depth images
   1. display and interpret images
   2. display R,G,B in separate figures
2. Display depth
   1. depth as image
   2. depth as a surface
   3. point cloud, mesh
   4. depth image vs. point cloud

Question: What is the "best" representation ?

edges

Dataset:

Load depth image from the lab (link)

read image from file (link)

load point cloud (link)

http://printart.isr.ist.utl.pt/piv/lab/pivlab1/

CODE GIFT - getting the triangular mesh from ordered points.

Display a triangular surface (top triangles)

J=repmat((1:639),479,1);

I=repmat((1:479)',1,639);

I1=[I(:), I(:), I(:)+1];

J1=[J(:), J(:)+1, J(:)+1];

ind1=(J1-1)\*480+I1;

%%

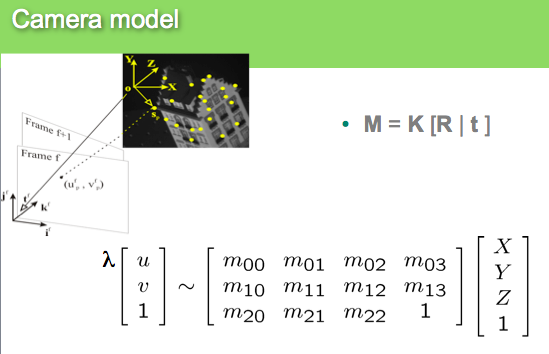
zs=xyz(ind1(:,1),3).\*xyz(ind1(:,2),3) .\* xyz(ind1(:,3),3);

ind1ok=find(zs>0);

trimesh(ind1(ind1ok,:),xyz(:,1),xyz(:,2),xyz(:,3))

Camera model -

* How to generate an image from 3D points?
* How to compute 3D positions from images/depth



Tasks:

1. given u,v and depth compute XYZ
2. generate image projection from point cloud
3. Rotate and Translate point cloud and do the same
4. Project Point cloud in the RGB image. What about the reverse: click rgb and get the 3D!