ELEC5306

Tutorial

https://canvas.sydney.edu.au/courses/21772/pages/course-contents

Tutorial 1

Digital Image

Introduction

Assuming knowledge:

Programming:, Matlab, Python, C++

Assignment:
Both assignments are programing assignments

Assignment 1:

20%, using Matlab. Individual

Assignment 2:

20%, using python. Individual or pair

Digital Image

Digital image:

Tensor 3 dimension (H,W,C), colorful image, rgb, (H,W,3)

RGB -> (W,H,3) -> W*H*3

YUV420P -> Y: W*H*1 U: W/2 * H/2 *1 V:W/2 * H/2=> 1.5*W*H

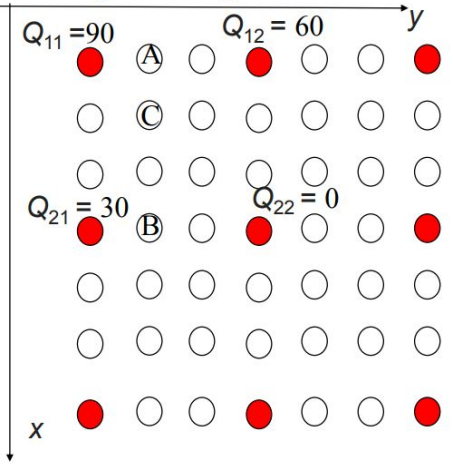
Bilinear Interpolation

Example 1

Bilinear interpolation example:

Enlarge an image to 9 times the size:

$Q_{11} = (x_1, y_1)$, $Q_{12} = (x_1, y_2)$,
$Q_{21} = (x_2, y_1)$, $Q_{22} = (x_2, y_2)$.



$$f(x_1, y) = \frac{y_2 - y}{y_2 - y_1} f(Q_{11}) + \frac{y - y_1}{y_2 - y_1} f(Q_{12})$$

$$f(x_2, y) = \frac{y_2 - y}{y_2 - y_1} f(Q_{21}) + \frac{y - y_1}{y_2 - y_1} f(Q_{22})$$

$$f(x, y) = \frac{x_2 - x}{x_2 - x_1} f(x_1, y) + \frac{x - x}{x_2 - x_1} f(x_2, y)$$

$$A = \frac{2}{3}90 + \frac{1}{3}60 = 80$$

$$B = \frac{2}{3}30 + \frac{1}{3}0 = 20$$

$$C = \frac{2}{3}80 + \frac{1}{3}20 = 60$$

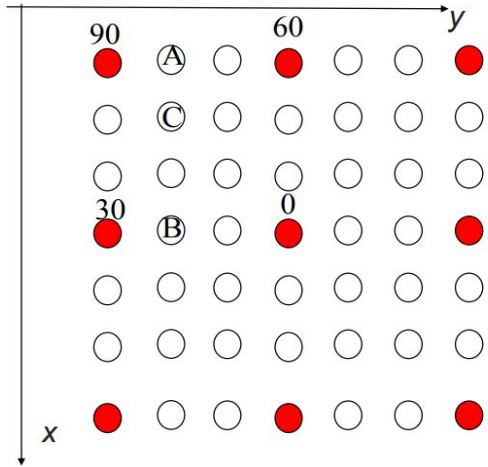
How about at the boundary of the image?

Example 2

Bilinear interpolation example:

Enlarge an image to 9 times the size:

$Q_{11} = (x_1, y_1)$, $Q_{12} = (x_1, y_2)$,
$Q_{21} = (x_2, y_1)$, $Q_{22} = (x_2, y_2)$.



$$A = \frac{2}{3}90 + \frac{1}{3}60 = 80$$

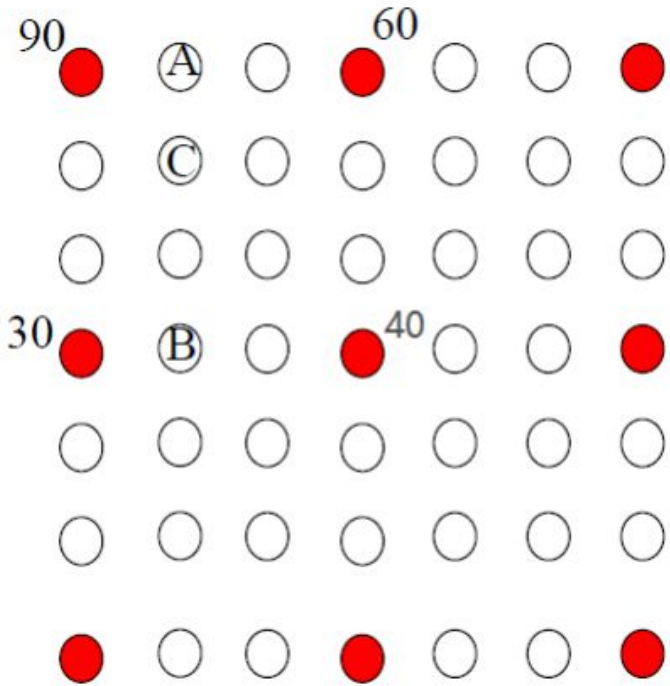
$$B = \frac{2}{3}30 + \frac{1}{3}0 = 20$$

$$C = \frac{2}{3}80 + \frac{1}{3}20 = 60$$

How about at the boundary of the image?

Question

Find the values of A, B and C?



Answer:

A = ⅔*90 + ⅓*60 = 80
B = ⅔*30 + ⅓*40 = 33.3333…
C = ⅔* 80 + ⅓ *33.33.. = 64.44

Matlab

MATLAB Installation using the USYD license

1) Visit the uni's website: https://sydney.edu.au/students/forms/student-it/matlab-use-agreement.html and accept the agreement.
2) Select the folder "2018b"
3) For Windows users, please download the R2018b_win64_dvd*.iso, for mac user please download the matlab_R2018b_maci64.dmg. Linux user please download the mirror R2017b_glnxa64_dvd*(Linux).iso
4) Follow the instructions on the page to install your matlab
5) Make sure you are already connected in the university's network (if not, please connect to the uni's VPN first), then launch your application.

Tutorial 2

Quality Assessment

Euclidean Distance: http://rosalind.info/glossary/euclidean-distance/

$$RMSE = \sqrt{\frac{1}{N \cdot M \cdot K} \sum_{k=1}^{K} \sum_{i=1}^{N} \sum_{j=1}^{M} [x(i,j,k) - \hat{x}(i,j,k)]^2}$$

$$PSNR = 20 \cdot \log_{10} \frac{255}{RMSE} \quad (\text{in } dB)$$

(x is the pixel value of the original video sequence, and $\hat{x}$ is the pixel value of the degraded video sequence.)

$$PSNR = 20 \cdot \log_{10} \frac{255}{RMSE} = 10 \log_{10} \frac{255^2}{MSE} \quad (\text{in } dB)$$

PSNR

- PSNR is not a good objective quality measure for comparing different types of video coding algorithms (e.g. comparing DCT-based algorithms with Wavelet-based algorithms).

- PSNR is a reasonable measure for comparing the same type of coding algorithms (e.g. DCT-based standards-conforming video coding algorithms).

- Use structural distortions to estimate perceptual distortions
- For still images, the SSIM index is :

$$\mathrm{SSIM}(x,y)=[l(x,y)]^{\alpha}[c(x,y)]^{\beta}[s(x,y)]^{\gamma}$$

where

$$l(x,y)=\frac{2\mu_x\mu_y+C_1}{\mu_x^2+\mu_y^2+C_1}$$

luminance

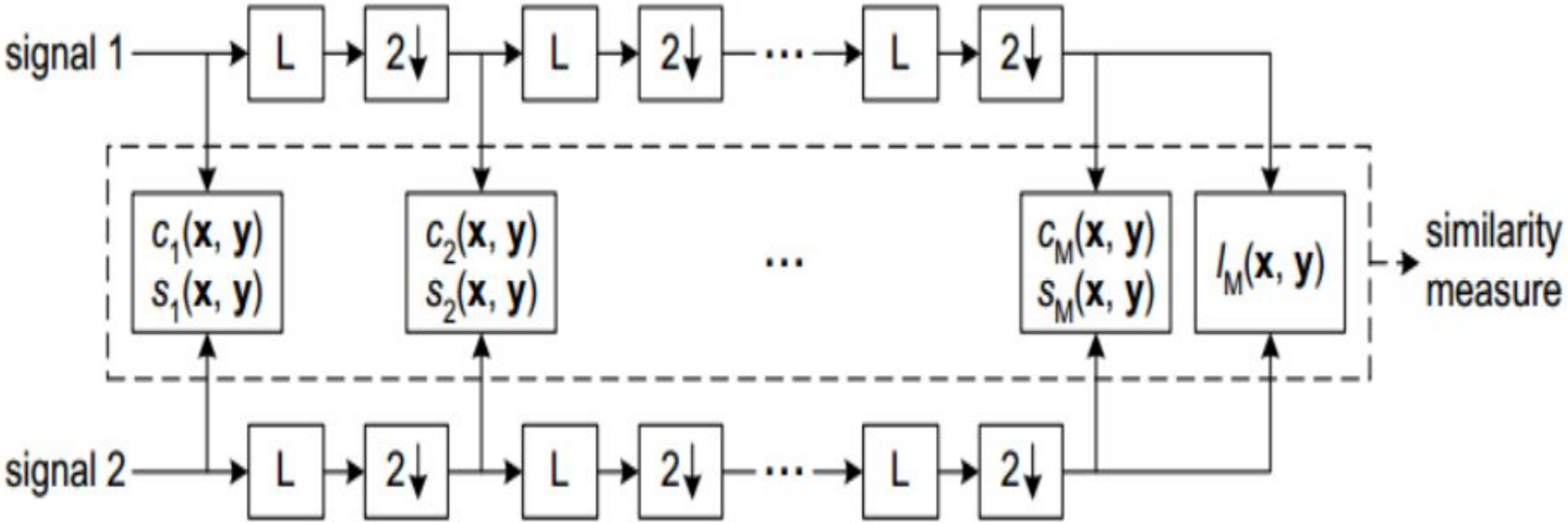
$$c(x,y)=\frac{2\sigma_x\sigma_y+C_2}{\sigma_x^2+\sigma_y^2+C_2}$$

$$s(x,y)=\frac{\sigma_{xy}+C_3}{\sigma_x\sigma_y+C_3}$$

$u_x, u_y, \sigma_x^2, \sigma_y^2$ and σ_{xy} are the mean of x, the mean of y, the variance of x, the variance of y, and the covariance of x and y respectively

are luminance, contrast, and structure comparisons respectively

MS-SSIM


signal 1
L
2↓
L
2↓
…
L
2↓
$c_1(x, y)$
$s_1(x, y)$
$c_2(x, y)$
$s_2(x, y)$
…
$c_M(x, y)$
$s_M(x, y)$
$l_M(x, y)$
similarity measure
signal 2
L
2↓
L
2↓
…
L
2↓

Tutorial 3

Lossless Coding

Notification

1. Wanli will hold discussion room 8:00-8:30 (from this week, please see anoncement in Canvas)
2. Please see the **Binary Arithmetic Decoding** (part-4) in this week lecture
3. From this week all content will be delivered online, please using Zoom
4. Assignment 1 will be released next week.

Elements of Information Theory

- Entropy (uncertainty): expected value of information associated with a set of source symbols.

$$H(z) = -\sum_{j=1}^{J} P(a_j) \log_2 (P(a_j))$$

(in bits/symbol, representing the minimum number of bits/symbol to represent the source symbols)

i = 2, p_1 = 1, p_2 = 0 => *H = 0*
i = 2, p_1 = 1/2, p_2 = 1/2 => *H = 1*
i = 4, $p_1 = p_2 = p_3 = p_4$ = 1/4 => *H = 2*
i = 4, p_1 = 0.5, $p_2 = p_3$ = 0.17, p_4 = 0.16 => *H = 1.79*

Question 1

i = 4, p1 = 0.4, p2= 0.18, p3 = p4 = 0.24, calculate the entropy H.

$$H(z) = -\sum_{j=1}^{J} P(a_j) \log_2(P(a_j))$$

H = - (**0.4***log(**0.4**) + **0.18***log(**0.18**) + **0.24***log(**0.24**)) + **0.24***log(**0.24**))

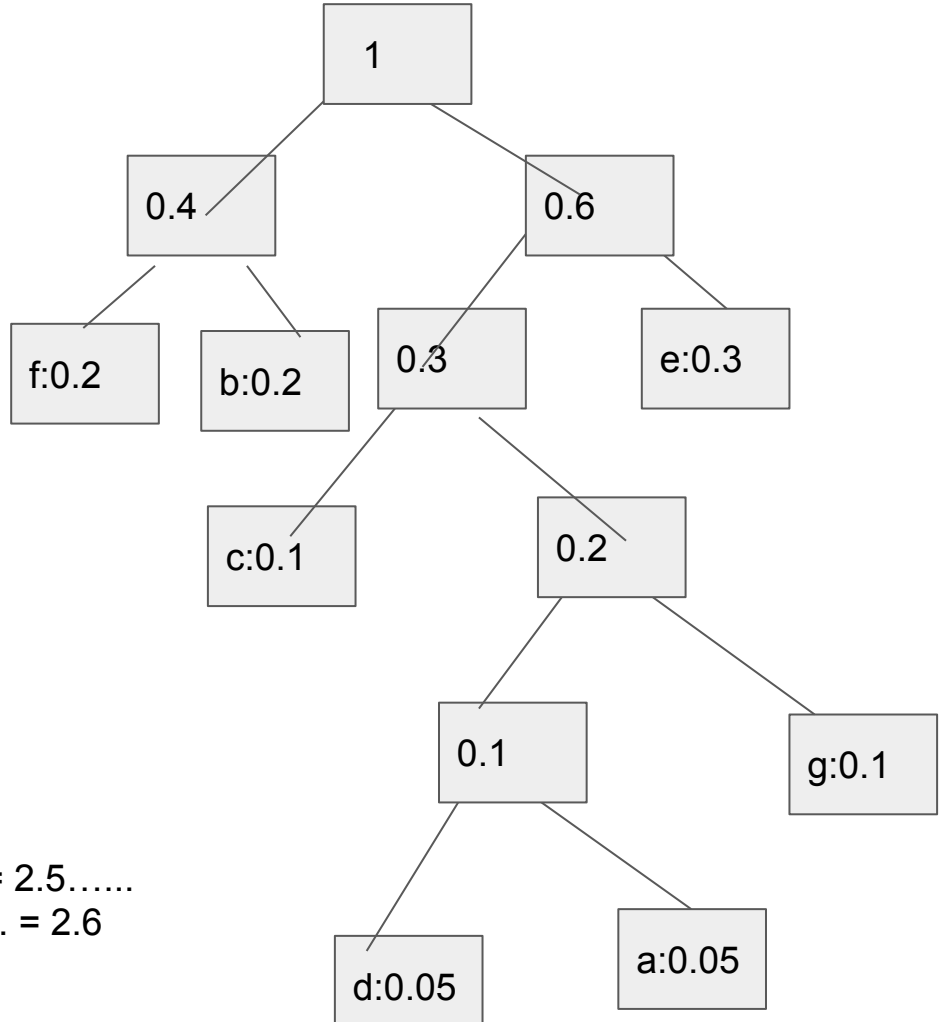
Huffman Tree

1. Tree doesn’t have to be identical, but different tree **might** have different **bits length**
2. To find out the most optimal tree for shortest bit length, you might need to use some efficient algorithm (Greedy search etc)
3. The bit length will be actually **longe**r than its cross entropy
4. **Cross entropy** is ideally the most optimal solution (but it can be achieved)

Have probability (symbol) -algorithm-> Human Tree -> (symbol -> bits based on the human Tree)

Huffman Coding

Symbol	Probability	Codeword
a	0.05	10101
b	0.2	01
c	0.1	100
d	0.05	10100
e	0.3	11
f	0.2	00
g	0.1	1011

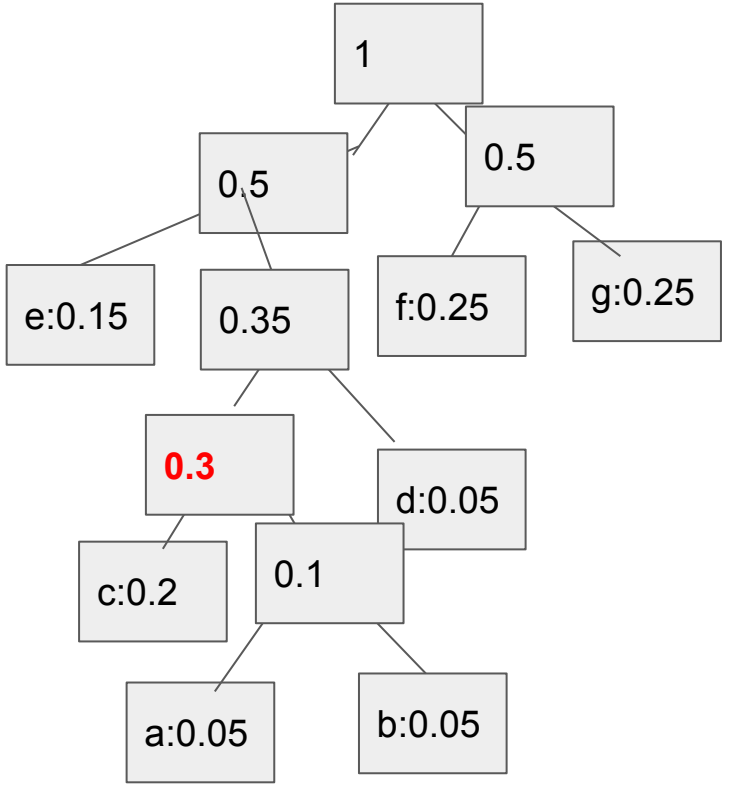


Entropy -0.05*log*0.5 - 0.2log0.2 + ..+ = 2.5…...
Average Coding rate: 0.05*5 + 0.2*2 +.... = 2.6

Huffman Coding

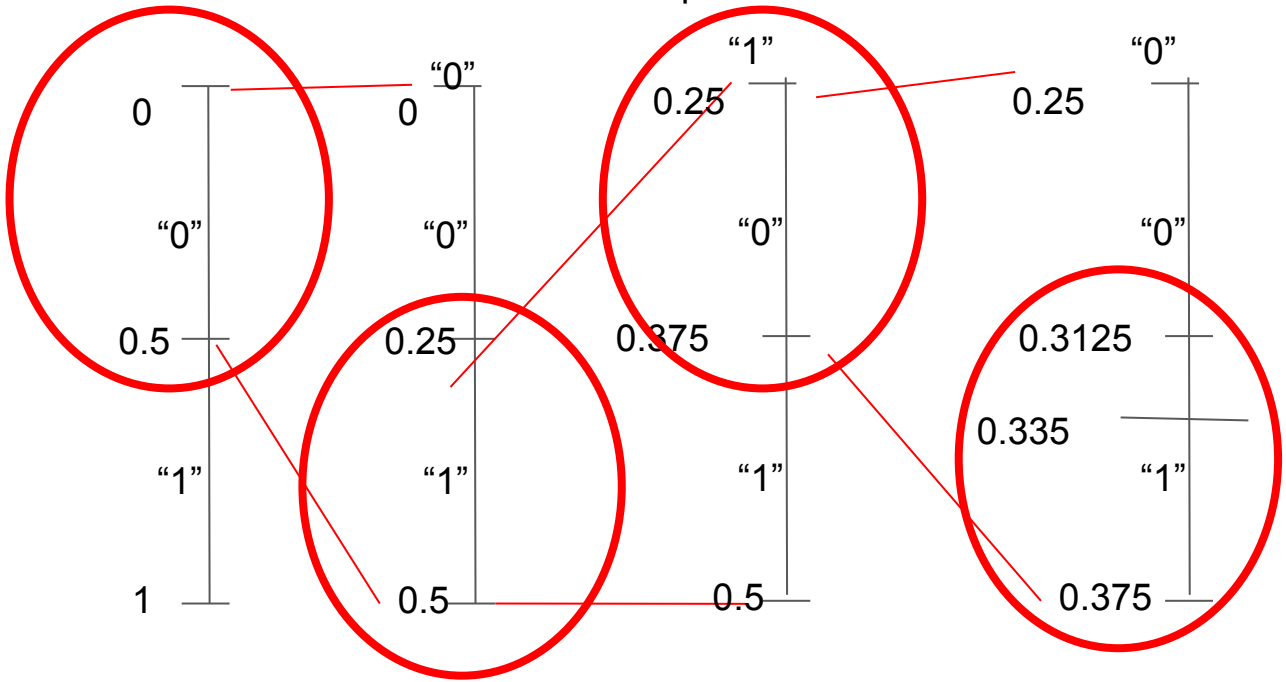
Draw the huffman tree, and answer the question: What's the probability of "010"?

Symbol	Probability	Codeword
a	0.05	01010
b	0.05	01011
c	0.2	0100
d	0.05	011
e	0.15	00
f	0.25	10
g	0.25	11



Question 3

Binary Arithmetic Coding: encoding of “0101” with probability of “0”=0.5 (P“1” = 0.5), show the final interval and the bit number after compressed.



log2(0.3125-0.375)=4

Tutorial 4

Your first assignment

Paper Reading (20%)

1. Do the oral presentation, Presentation Date is 24th April in the lecture time
2. make a video, due day is not confirmed

Paper reading list is released

https://docs.google.com/document/d/1ATxO2U-LtW1R75kBDIOfyDptTfMCCOef03YSzaz902Y/edit?usp=sharing

You can pick up hard, middle and easy paper, our marking would be different

For example, if you pick a easy paper, you need to 100% understand this paper (including detail such as math prove) no mistakes allowed

In contrast, if you pick a hard paper, we will allow you have a little bit mistakes during understand.

The deadline of deciding paper should be next week Friday 12:00 pm

Assignment 1 is Released

Due: 1th May

Content：2 main files (**part1.m, part2.m**) 1 report (report.pdf) (less than 3 pages)

You can submit more than 3 files (e.g. function_1.m to support part1.m) we won't punish, but we only run part1.m and part2.m and check report.pdf for marking.

What should you do：

Part1.m: complete task 1
Part2.m: complete task 2
Report.pdf: Objective, Introduction, Methodology, Simulation Results, Discussion, Conclusion, References (optional), etc

Assignment 1

Read YUV file

https://www.google.com/search?q=read+yuv+file+matlab&oq=read+yuv+file+matlab&aqs=chrome..69i57j0l2.4401j0j7&sourceid=chrome&ie=UTF-8#kpvalbx=_aa59XufRNd-E4-EPj-KhuAY22

Qutization:

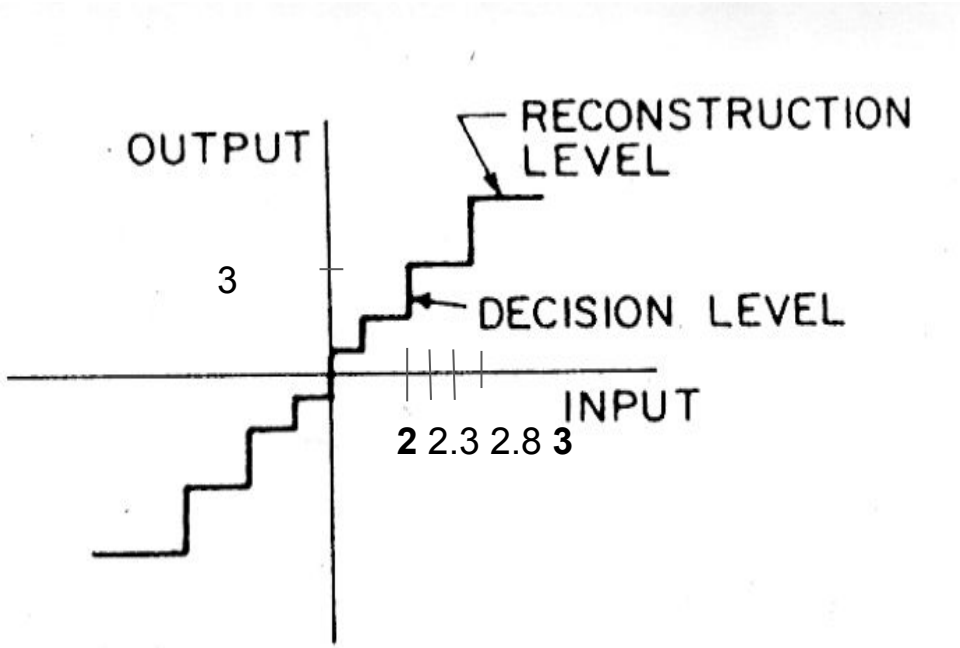
1. Round your continues value to discrete value
2. Optimize the distance

Tutorial 5

Quantization and Deep Learning

Quantization (2nd part)

Goal: quantize continues value to discrete



1. The decision level can be either equal (linear) or inequal(non-linear)
2. There are two challenges
   a. Less quantization error
      i. Psnr, msssim (big)
   b. Less bits usage
      i. Bits (less)

Quantization

S	0	1	2	3	4	5	6	7
S/2	0	0.5	1	1.5	2	2.5	3	3.5
Q_I	0	0	1	1	2	2	3	3
$Q_I \times 2$	0	0	2	2	4	4	6	6
e_q	0	1	0	1	0	1	0	1
$Q_I \times 2+1$	1	1	3	3	5	5	7	7
e_q	1	0	1	0	1	0	1	0
P	0.4	0	0	0.1	0.2	0	0.3	0

In this case, I want to quantize these numbers with 2 bits (00, 01, 10, 11)

S/4	0	0.25	0.5	0.75	1	1.25	1.5	1.75
Q_I	0	0	0	0	1	1	1	1
$Q_I \times 4$	0	0	0	0	4	4	4	4
e_q	0	1	2	3	0	1	2	3

In this case, I want to quantize these number with 1 bit (0, 1)

Normally if you use less bits, normally your reconstruct qulatily would be less

Quantization

P	0.4	0	0	0.1	0.2	0	0.3	0
S	0	1	2	3	4	5	6	7
E =	0	1	2	3	0	1	2	3

D = 0.9 + 1.2 = 2.1
D0 = S0, Dlast = 7

N < Dj + D -> rj
N > Dj + D -> rj+1

Decide the optimal decision boundary

OPTIMAL QUANTIZATION

- Original sample/symbol **f,** reconstruction **f'**.
- Decision level $\boldsymbol{d_j}$, reconstruction $\mathbf{r}_j$.
- The optimal reconstruction levels, $\{\boldsymbol{r_j}\}$, in minimum mean squares error (MMSE) sense:

$$D = E[\mathrm{e}_\mathrm{Q}^2] = E[(\mathbf{f}-\mathbf{f}')^2] = \int_{\mathbf{f}=-\infty}^{\infty} p(\mathbf{f})(\mathbf{f}-\mathbf{f}')^2 \mathrm{d}\mathbf{f}$$

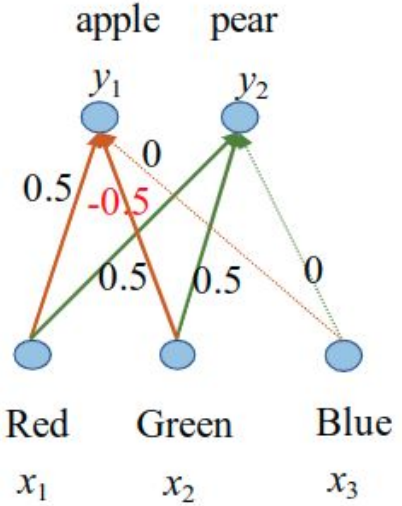
$$= \sum_{\mathrm{j}=0}^{\mathrm{J}-1} \int_{\mathbf{f}=\boldsymbol{d}_\mathrm{j}}^{\boldsymbol{d}_{\mathrm{j}+1}} p(\mathbf{f})(\mathbf{f}-\mathbf{r}_\mathrm{j})^2 \mathrm{d}\mathbf{f}$$

$$\mathrm{r_j} = \frac{\mathrm{d_{j+1}} + \mathrm{d_j}}{2} = \mathrm{d_j} + \frac{\Delta}{2}$$

Neural Network

Regression: y = f(x), y is a continue number
Classification: y = f(x), y is discrete vector

A simple example for neural network
V = [x1, x2, x3] = > y



Example :
V = [1,0,1] -> y1
1*0.5+0*-0.5 + 1*0 =0.5
1*0.5 + 0*0.5 + 0*1 = 0.5
=>[0.5, 0.5]

[1,0,1] => [0.5, 0.5]

Backpropagation :

1. Gradient descent
2. Chain Rule

=> optimize the weight to better perform the task

CNN & RNN

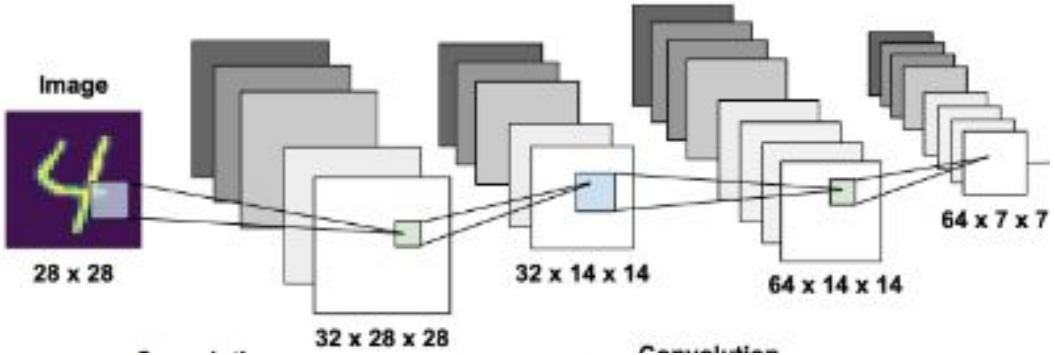
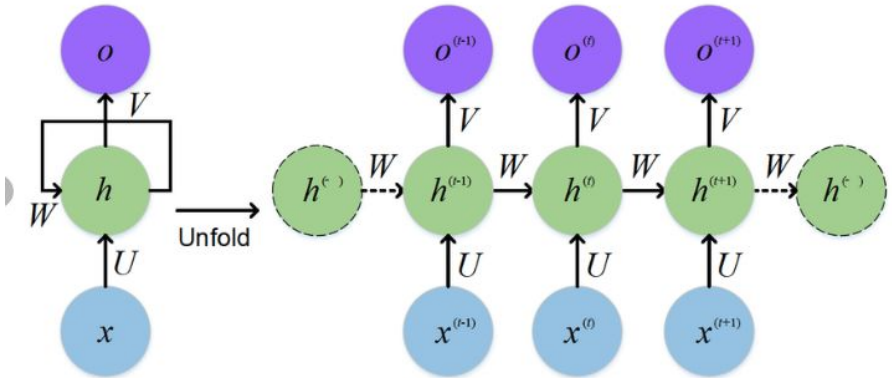


Image
28 x 28
32 x 28 x 28
32 x 14 x 14
64 x 14 x 14
64 x 7 x 7


o
V
h
W
U
x
Unfold
h(-)
W
o(t-1)
h(t-1)
x(t-1)
o(t)
h(t)
x(t)
o(t+1)
h(t+1)
x(t+1)
h(-)

Build Neural Network using pytorch

Input data x, label y
Build a neural network f

f(x, y).train() -> f with a optimized weight

Y’ = f(x’) , x’ is test data, y is test label

Tensorflow, pytorch mxnet, caffe, matlab..
Python ,c++, ….

We only provide code with pytorch

Tutorial 6

Your first CNN in Colab

Paper Reading

1. You need to submit：
2. Group ID
3. Marking form
4. Presentation time
5. QA

https://canvas.sydney.edu.au/courses/21772/discussion_topics/395402

If you want to install the environment in yours

Install anaconda on windows:
https://docs.anaconda.com/anaconda/install/windows/

Install cuda on windows: (optional：If you have a GPU in your machine)
https://docs.nvidia.com/cuda/cuda-installation-guide-microsoft-windows/index.html

Install pytorch with anaconda:
https://medium.com/@bryant.kou/how-to-install-pytorch-on-windows-step-by-step-cc4d004adb2a

Tutorial 7

Your assignment 2

DCT example

The integer DCT operates on a 4×4 block of residual data after motion-compensated prediction. The core part of the transform can be implemented using only additions and shifts. To reduce the total number of multiplications, the scaling-multiplication is integrated into the quantizer. The inverse quantization (scaling) and inverse transform operations can be carried out using 16-bit integer arithmetic with only a single multiply per coefficient, without any loss of accuracy.

The conventional DCT basis is given by

$$A_{ij} = C_i \cos\frac{(2j+1)i\pi}{2N} C_i = \sqrt{\frac{1}{N}}(i=0), C_i = \sqrt{\frac{2}{N}}(i>0) \quad (1)$$

Given the input block as X, recovered block as Y, then what's the formulation of generating Y? And what's A?

Answer:

$$Y = AXA^T$$

$$\text{where } A = \begin{bmatrix} \frac{1}{2} & \frac{1}{2} & \frac{1}{2} & \frac{1}{2} \\ \sqrt{\frac{1}{2}}\cos(\frac{\pi}{8}) & \sqrt{\frac{1}{2}}\cos(\frac{3\pi}{8}) & -\sqrt{\frac{1}{2}}\cos(\frac{3\pi}{8}) & -\sqrt{\frac{1}{2}}\cos(\frac{\pi}{8}) \\ \frac{1}{2} & -\frac{1}{2} & -\frac{1}{2} & \frac{1}{2} \\ \sqrt{\frac{1}{2}}\cos(\frac{3\pi}{8}) & -\sqrt{\frac{1}{2}}\cos(\frac{\pi}{8}) & \sqrt{\frac{1}{2}}\cos(\frac{\pi}{8}) & -\sqrt{\frac{1}{2}}\cos(\frac{3\pi}{8}) \end{bmatrix}.$$

Tutorial 8

Your assignment 2

Assignment 2

1. Dataset

https://www.dropbox.com/s/i1o74xou8bm74al/data.zip?dl=0

You can only use the data we provide for assignment not extra data

But you can re-split the data into the way you prefer

2. Code

https://github.com/chriszhenghaochen/ELEC5306

This is the main code

3. baseline-model

https://drive.google.com/open?id=1ETnahHe9EoZzCWXFvfqa6EUz8O3ty2di

4. colab-code

This help you to run your code in colab

https://drive.google.com/open?id=1ETnahHe9EoZzCWXFvfqa6EUz8O3ty2di

Tutorial 9

Improve your neural network

Assignment 2

code:

1. You need to provide the **README** file
2. You need to make sure that your code can read the test list as structure we give to you

Presentation

1. 10mins +-1 mins + 2mins(discussion)
2. You have to tell us the contribution of each members in your team during the presentation
3. You need to submit a backup video like lastime, upload your video into the youtube and submit the link.

Report

1. We list in the marking rubuic

Video concept

1. Topic
   You need register topic then do video, we would check the discussion when we marking if
2. Marking
   In this time you require to have fully understanding with the topic your choose

3. Penalty
   a. You shouldn't choose the same topic others choose by others first
      i. If you are second person to choose the topic you will receive 30% deduction
         If you are the third one to choose the topic you will receive 60% deduction
      ii. If your presentation slides have more than 20% as the person(s) who choose the topic before you, it counts as **plagiarism** (The university treat this problem very serious, normally you will directly fail the unit,
         (see https://www.sydney.edu.au/students/academic-dishonesty.html)

   b. You can't choose the same topic you have done it in ELEC5304, If you do so, you will receive 0% for both units.
   c. You don't register any topic you will receive a 0% in anyhow

Data

1. Data is probably the most important thing in the network,
   1-1. you **can't** change the **test data loader**, but you can write your own training dataloader
   1-2 Argumentation: 1. random crop, 2. random flip, 3. add guassian noise,
   Ref: Pytoch transform https://pytorch.org/docs/stable/torchvision/transforms.html
   Ref: https://arxiv.org/pdf/1512.02325.pdf sec 3.6

   More data is not guarantee to give a better results, but emperically it will improve

2. You can only the data we give you, remember you can re-split the val and training

Overfitting and network design for better Bp

1. I would personally suggest do not add to much parameters for your network (aka make your newtwork too big), otherwise it is easy to get you network overfiting.

2. Normally the deeper network is much harder to train and converge. But there is something you can do

   2-1 Use normalization：it can make your output to be a better distribution that good for back-propagation
   a. BN in torch: https://discuss.pytorch.org/t/batch-normalization-of-linear-layers/20989
   b. BN paper: https://arxiv.org/abs/1502.03167

   2-2 Use skip connection
   a. Skip connection: https://arxiv.org/pdf/1512.03385.pdf (resnet)
   b. Torch **x = x + f(x)**

   3-3 Use dropout
   a. http://jmlr.org/papers/v15/srivastava14a.html
   b. https://stackoverflow.com/questions/53419474/using-dropout-in-pytorch-nn-dropout-vs-f-dropout

What you can do during the training

1. Select a better **learning rate**:
   a. Too big, it converge fast but it might jump over the best optimized point
   b. Too small it converge slow and it might stuck on certain local optimized

2. Try different way of decay
   a. This will help you to discover different local optimize

3. Try different Batch size
   a. This is will be especially helpful if you use a Batch Norm

4. Try different optimizer
   a. SGD
   b. Adam
   c. ……...

Last word for you

a. Keep tuning and reading more papers to get your idea!
b. Good luck