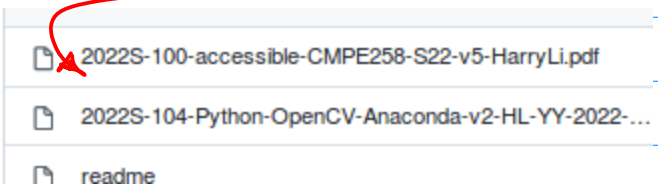
CMPE258 Spring2022

1/

Feb1st. Organizational meeting.

1. Today's Topics "Greensheet"



Naming Convention Yr+Semester+ID +Name+Date

Contact Information: E-mail: hua.li@sjsu.edu

Text message to (650)400-1116.

Office Hours: M.W. 4:30-5:30pm.

Zoom (link to be shared in the email)
Join from PC, Mac, Linux, iOS or Android: https://sjsu.zoom.us/j/85616325978?pwd=MzlRbDJXVHBDQ2g1U0RPM2tYc045Zz09

Password: 451032

On-Line materials on github
https://github.com/hualili/opencv/tree/master/deep-learning-2020S
https://github.com/hualili/opencv/tree/master/deep-learning-2022s

Also, CANVAS → mostly for Assignments and projects.

All Assignment/Projects are posted on Both github & SJSU CANVAS.

Lecture Material consists of PPT. posted on github, and Lecture Notes (White-Board Written Notes)

Core Emphases of the Class: Deep Convolutional Neural Networks, And their Application in Image Analysis, Video Analysis.

1. Text Book:
https://github.com/hualili/opencv/blob/master/IP120-AI-DL/2018F/2018F-6-DeepLearningCh02.pdf

2. Computer Vision Book By Horn as a reference for Convolution & Image Segmentation, Contours Analysis (Binary Image).

3. OpenCV Reference Book (2nd Edition) together with On-Line Document (OpenCV).

Note: OpenGL (GL: Graphics Library) is just for Reference purpose, no need for this class. (But maybe helpful for the future research).

Unity is game Development platform, interactive 3D Graphics Design platform.

Programming Languages:
1' Python. 3.6 or 3.7
2. C/C++

Feb.8th.
Homework (Due A week from today)
No Submission. Submit A screen

Capture that shows OpenCV installed successfully, with Jpg or png file with Naming of the file as follows:

FirstName_LastName_SID_OpenCV.jpg

This Homework will be posted on CANVAS, Submission is on CANVAS

Homework, Installation of Tensor Flow, Due 2weeks Feb 15th.

Submission: Screen Capture that shows the installation is successful. Submission on CANVAS.

Submit jpg, png file with the Naming convention as follows:

FirstName_LastName_SID_TF.jpg

Note: Optional, for Edge AI Computing, Consider using NVDA Jetson NANO (4GB) version.

5% Bonus

Grading:

Homework, projects : 30%
5% 25%

Project 1. Computer Vision for Preprocessing. plus Deep Convolutional Neural Nets To give Real Time Detection Result of Last 4 Digits of a Student ID.

10%.

Project 2. "Semester Long" project, with technical requirements (List)

Team project. 4 person Team.

Each person has clear Definition of the tasks (Programming/Coding) And Balanced Contribution.

Final TPT, Demo Presentation

15%.

Midterm Exam: 30%.

Need to use your Laptop Computer, to Run/Execute code, modify the code.

Final 40%

Introduction

Topics { Neural Networks formulation (Basic Building Blocks)
Digital Images/Videos.

Example: A Single Neuron Formulation (Some kind Brain Cell)

Step 1. Summation function.

Fig 1.

"Σ" Summation function,

Note: $\sum_{K=1}^{N} x_K = x_1 + x_2 + \cdots + x_N$

Step 2. Inputs

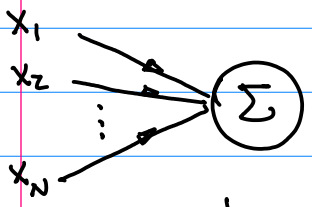
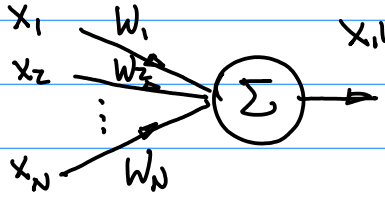


Fig 2.

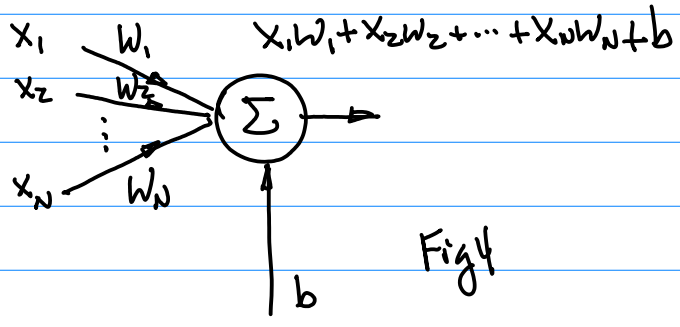
Step 3. Weights (Knowledge)



$x_1 w_1 + x_2 w_2 + \cdots + x_N w_N = \sum_{K=1}^{N} x_K w_N$

Fig 3.

$$\sum_{K=1}^{N} x_K w_N = x_1 w_1 + x_2 w_2 + \cdots + x_N w_N \quad \cdots (1)$$



$x_1 w_1 + x_2 w_2 + \cdots + x_N w_N + b$

Fig 4

$$\sum_{K=1}^{N} x_K w_N = x_1 w_1 + x_2 w_2 + \cdots + x_N w_N + b \quad \cdots (1b)$$

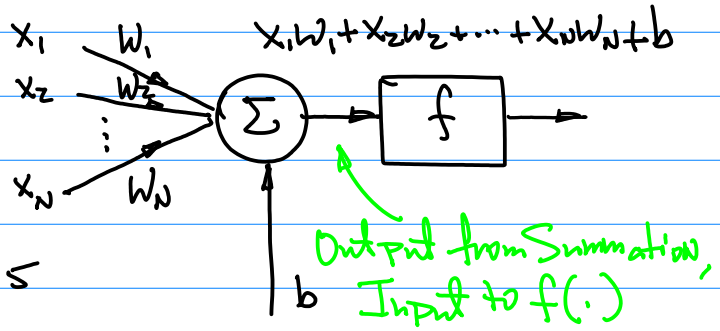


Fig 5

Note: Activation function f, denoted as f(·) (A function of Independent Variable "·", or A function of Input "·")

$$f(\cdot) = f(x_1 w_1 + x_2 w_2 + \cdots + x_N w_N + b)$$

$$= f\left(\sum_{K=1}^{N} w_K x_K + b\right) \quad \cdots (2)$$

Summary: The output of a Single is given by Eqn (2). Where Activation function f(·) can take different forms, it affects the Learning, Learning Speed.

Example: Digital Image, I(x,y)

$I(x,y)$

(x,y) Location of A picture element, "pixel"

Intensity, And/OR Color of An Image

In Case of a Single pixel, (x,y) is the Location of this pixel, I is Color/Intensity of the pixel

Note: For An Image $I(x,y)$ its features include { Resolution $M \times N$ (No. of pixels per Row, No. of Rows per frame); Pixel Depth: bpp (Bit per pixel)

$I(x,y)_{M \times N}$ OR $I(x,y)$

1024 x 768 = pixels/Row = Rows

For A color Image, A pixel depth very often is equal to 24 (bpp)

r, g, b Primitive color of red (r), green (g), blue (b) has 8 bits quantization level, e.g.

r: [0, 255], g: [0, 255], b: [0, 255].