Lecture Notes
PART II

CmpE258
Spring 2023

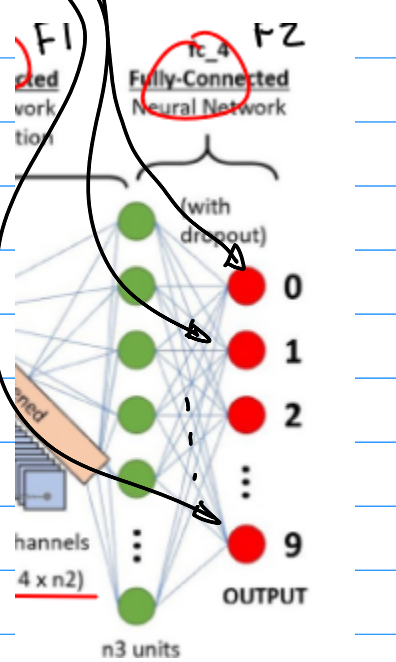
March 21 (Tue).
Midterm is scheduled on the 23rd (Thu). 1 hr Exame.
16:30 - 17:30
Then 15 minutes for prep & Uploading the file.

Example: Softmax Activation Function. MNIST CNN

Output from Each Neuron:

$$f(z_1) = \frac{e^{z_1}}{\sum_{j=1}^{10} e^{z_j}} \quad \ldots (4)$$

for Digit "0" (1st Output)



And

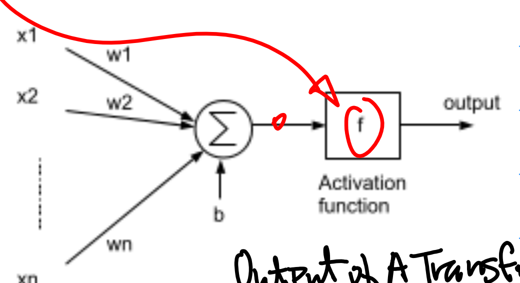
Note:

$$\sigma(\mathbf{z})_i = \frac{e^{z_i}}{\sum_{j=1}^{K} e^{z_j}} \quad \text{for } i = 1, \ldots, K \text{ and } \mathbf{z} = (z_1, \ldots, z_K) \in \mathbb{R}^K. \quad \ldots (1)$$

Dimension

Let K=1

1° $f(z)$ Notation



Output of A Transfer function

$$f(z_2) = \frac{e^{z_2}}{\sum_{j=1}^{10} e^{z_j}}$$ for Digit "1" (2nd output) ...(5)

$\vdots$

$$f(z_{10}) = \frac{e^{z_{10}}}{\sum_{j=1}^{10} e^{z_j}}$$ for Digit "9" (10th o/p)

2° Index

$$f(z_i) = \frac{e^{z_i}}{\sum_{j=1}^{K} e^{z_j}} \quad \ldots (2)$$

Total No. of Output Neurons $= K$ for Hand Written Digits Recognition $K=10$;

Where

$$\sum_{j=1}^{K} e^{z_j} = e^{z_1} + e^{z_2} + \cdots + e^{z_K} \geq e^{z_i} \quad \ldots (3)$$

if Add Eqn (1) + Eqn (2) + ...

$$f(z_1) + f(z_2) + \cdots + f(z_{10})$$

$$= \frac{e^{z_1}}{\sum_{j=1}^{10} e^{z_j}} + \frac{e^{z_2}}{\sum_{j=1}^{10} e^{z_j}} + \cdots\cdots + \frac{e^{z_{10}}}{\sum_{j=1}^{10} e^{z_j}}$$

Property 1.

$$= \frac{\sum_{j=1}^{10} e^{z_j}}{\sum_{j=1}^{10} e^{z_j}} = 1. \quad \ldots (6)$$

CmpE258

Spring 2023

The softmax function takes as input a vector z of K real numbers, and normalizes it into a probability distribution consisting of K probabilities proportional to the exponentials of

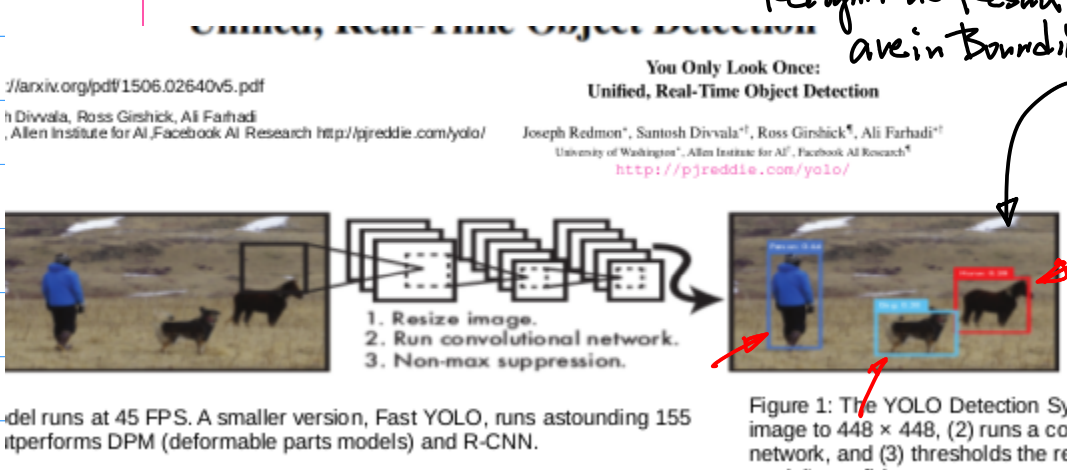
$$f(z_i) = \frac{e^{z_i}}{\sum_{j=1}^{K} e^{z_j}}$$

See Eqn (6).

Now, move to the 2nd Half of DCNN.

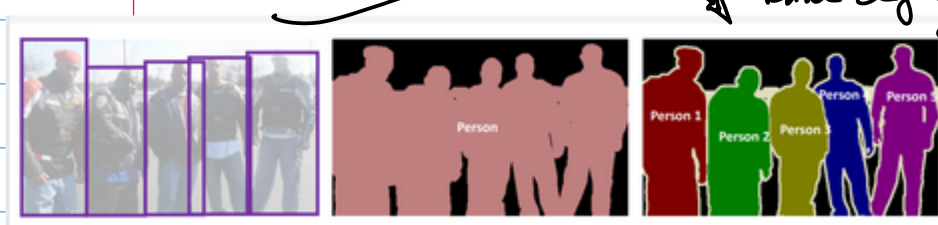
2022F-108a-Yolo-architecture-loss-function-2022-10-10.pdf

Typical Classification/Recognition Results are in Bounding Boxes



Then, we would like to Achieve Semantic Segmentation.

Pixel by pixel Based Segmentation/Detection/Recognition



PART II (After the midterm) → Semantic Segmentation.

April 4 (Tue)

Road map: YoLo (You-Only-Look-ONCE)

{ Project Assignment = Yolo Implementation
Due $2\frac{1}{2}$ weeks. April 20th. (Thursday)

CmpE258

Spring 2023

Homework (In-Class Presentation)

Requirements Due April 6 (Thu).

1° One Paragraph Description (Abstract) of the proposed Semester-Long Project.

2° Title

Team members: First Name, Last Name, Major

Team Coordinator.

Contact E-mail.

3° Abstract Part.

Objective(s): a) What is the proposed work;

b) What is the Coding/Training/Testing Task involved in the project?

c) Anticipated Result? And deliverable?

d) Tools, platform, Programming Language Version, T.F., Pytorch, ChatGPT etc. Also, Define Python Packages, OpenCV.

Example: On Yolo.

Ref:

2022F-106-README-Tiny-Yolo4-GP...

34

Note 1: Readme for Yolo github Installation & Testing for the project.

Title: README Tiny Yolo v4 GPU Ubuntu

Document Number: 105-1b

CTI One Corporation

Table 1a. Document History

2022-10-6	Establish this document, document archive: (base) harry@workstation:~/yolo-2022-10-19$	YY

1. Setup YOLO v4 environment

1.1. Clone the GitHub folder;

```
$ git clone https://github.com/pythonlessons/TensorFlow-2.x-YOLOv3.git
```

1.2. Create YAML file for building the YOLO v4 Anaconda environment;

Create TensorFlow-2.x-YOLOv3/conda-gpu.yml as the following;

```
================================================
name: yolov4-gpu
```

Ref: Introduction

2022F-108a-Yolo-architecture-loss-function-2022-10-10.pdf

Base-Line Ref for Yolo Technique

2022S-112-yolo-paper.pdf

You Only Look Once:
Unified, Real-Time Object Detection

Joseph Redmon*, Santosh Divvala*†, Ross Girshick¶, Ali Farhadi*†
University of Washington*, Allen Institute for AI†, Facebook AI Research¶
http://pjreddie.com/yolo/

Lecture Notes: Base Line Ref/Requirements.

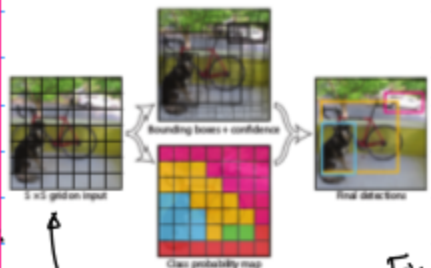
2022F-101-cmpe258-note-2022-11-1.pdf

Example: Notations for YoLo.

Ref, PP36.

CmpE258
Oct.13,22

Fig 1.



Note:

~~Fig.2a~~

1. Image $I(x,y)$ is divided into $S \times S$ Grids.

Denote it as $G_{p,q}(x,y)$, ... (1)

where $p,q = 0,1,2,\dots$ indicate the

1) Image $I(x,y)$. with Resolution

$M \times N$

No. of Col. No. of Rows.

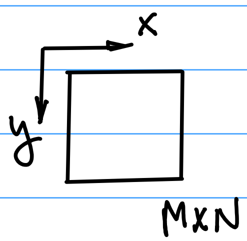


Fig.1b.

Divide $I(x,y)$ into $S \times S$ Grids.

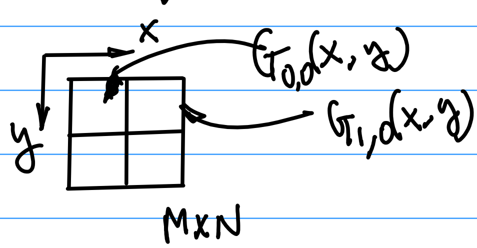
Each Grid is Denoted as

$G_{pq}(x,y)$ ... (1)

matches to x

where $p,q = 0,1,2,\dots,S-1$

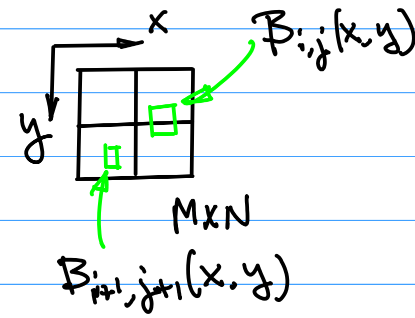
Fig.1c



2. Bounding Boxes $B_{ij}(x,y)$ ... (2)

i for x-direction; j for y-direction

Fig.1a



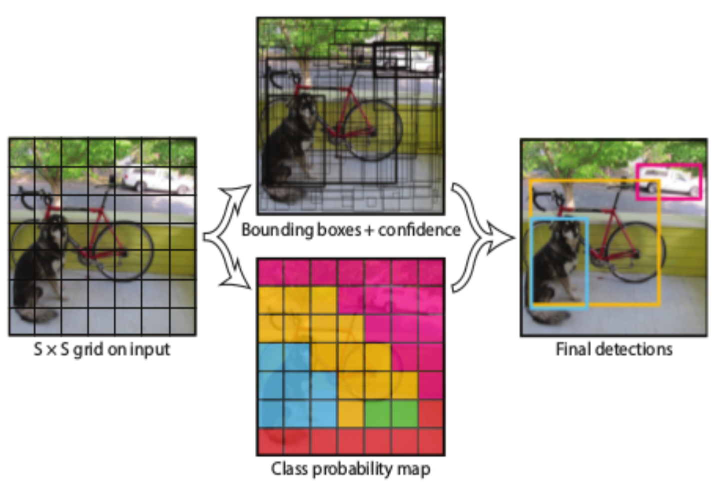
3. Five Parameters to define each Bounding Box.

(x,y) : Location of the top Left Corner of $B_{ij}(x,y)$

w,h : Width and Height of $B_{ij}(x,y)$

f : Confidency Level, Probability distribution to Describe the likelihood of the B.B (B^2) belongs to a certain Class of objects.

(x,y,w,h,f) ... (3)



CmpE258
Spring 2023

Note 1. a. (Gpq(x,y) Grid.

Figure 2: The Model. Our system models detection as a regression problem. It divides the image into an $S \times S$ grid and for each grid cell predicts B bounding boxes, confidence for those boxes, and C class probabilities. These predictions are encoded as an $S \times S \times (B * 5 + C)$ tensor.

b. $B_{ij}(x,y)$

c. (x, y, w, h, f)

Class probability.

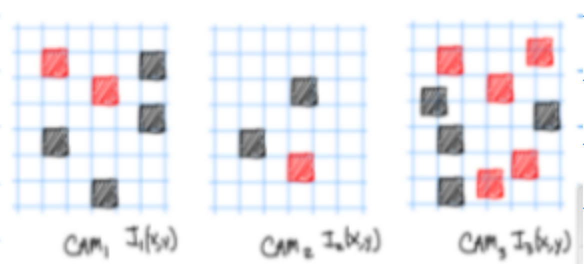
Probability

Confidency.

April 6 (Th).

Example: Discussion on Notation / Formulation.

Ref. 2022F-101-cmpe258-note-2022-11-1.pdf Pp38



Camera 1: $I_1(x,y)$ Camera 2 $I_2(x,y)$ Cam 3. $I_3(x,y)$

$$R = RI_1 + RI_2 + RI_3 \quad \ldots (1)$$

R: Red Squares, Persons.
B: (Black) Vehicles.

for "Union" " $\cup$ "

Intersection. " $\cap$ "

Consider the probability of the event "R" (meaning Person(s) being Capture on any one of these images).

$$Prob(R) = Prob(RI_1) + Prob(RI_2) + Prob(RI_3) \quad \ldots (2)$$

$I_1 \cap I_2 \cap I_3 = \phi$ (Empty set)

Consider Each Individual Camera:

$$Prob(RI_1) = Prob(R|I_1)\,Prob(I_1) \quad \ldots (3a)$$

Similarly,

$$Prob(RI_2) = Prob(R|I_2)\,Prob(I_2) \quad \ldots (3b)$$

$$Prob(RI_3) = Prob(R|I_3)\,Prob(I_3) \quad \ldots (3c)$$

Rewrite Eqn (2):

$$Prob(R) = Prob(R|I_1)\,Prob(I_1) + Prob(R|I_2)\,Prob(I_2) + Prob(R|I_3)\,Prob(I_3) = \sum_{i=1}^{3} Prob(R|I_i)\,Prob(I_i) \quad \ldots (4)$$

Ref: 2022S-112-yolo-paper.pdf

$$\Pr(\text{Class}_i|\text{Object}) * \Pr(\text{Object}) * \text{IOU}^{\text{truth}}_{\text{pred}} = \Pr(\text{Class}_i) * \text{IOU}^{\text{truth}}_{\text{pred}} \quad (1)$$

Example: pp36

Note 1. Conditional Probability. Prob(C_i)

Prob(C_i|Obj) Prob(Obj)

1. IOU (Intersection of Union)

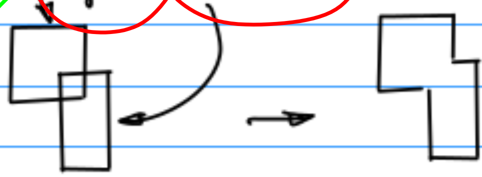
Index for the purpose of Comparing 2 Bounding Boxes at a time

pp.37

The Union of $B_{ij}(x,y)$ $B_{GT}(x,y)$

Note: "GT" Ground Truth.

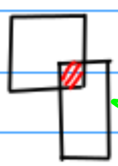
Note: $B_{ij}(x,y)$ for the i,j B.B



The Union

The Intersection of $B_{ij}(x,y)$, $B_{GT}(x,y)$

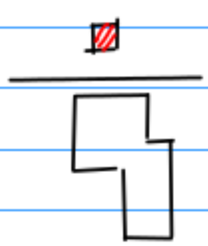
Fig 1



The prediction matches the Ground Truth. (GT) is on the left

$$\text{IOU} = \frac{\text{Intersection of } B_{ij}(x,y) \text{ and } B_{GT}(x,y)}{\text{The Union of } B_{ij}(x,y) \text{ and } B_{GT}(x,y)} \quad \cdots (5)$$

pp37

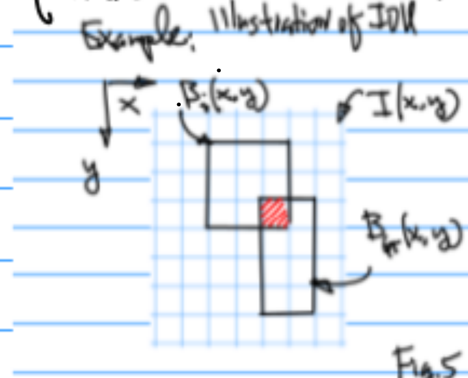
C =  ~~...(1)~~

(IOU) Confidence, denoted as

Hand Calculation of IOU.

5. IOU (Intersection of Union).

Fig.3



Sol. First, find the Number of pixels of the $B_{ij}(x,y) \cap B_{GT}(x,y)$,

$B_{ij}(x,y) \cap B_{GT}(x,y) = 1$

then, Find the Union

$$B_{ij}(x,y) \cup B_{GT}(x,y) = N[B_{ij}(x,y)] + N[B_{GT}(x,y)] - N[B_{ij}(x,y) \cap B_{GT}(x,y)]$$

$$= (3 \times 3) + (2 \times 4) - 1 = 9 + 8 - 1$$

$$= 17 - 1 = 16$$

$\therefore$ IOU $= 1/16$

Now, from Eqn (1) of the "Ref. (Research Paper)"

Note

$$\Pr(\text{Class}_i|\text{Object}) * \Pr(\text{Object}) * \text{IOU}^{\text{truth}}_{\text{pred}} = \Pr(\text{Class}_i) * \text{IOU}^{\text{truth}}_{\text{pred}} \quad (1)$$

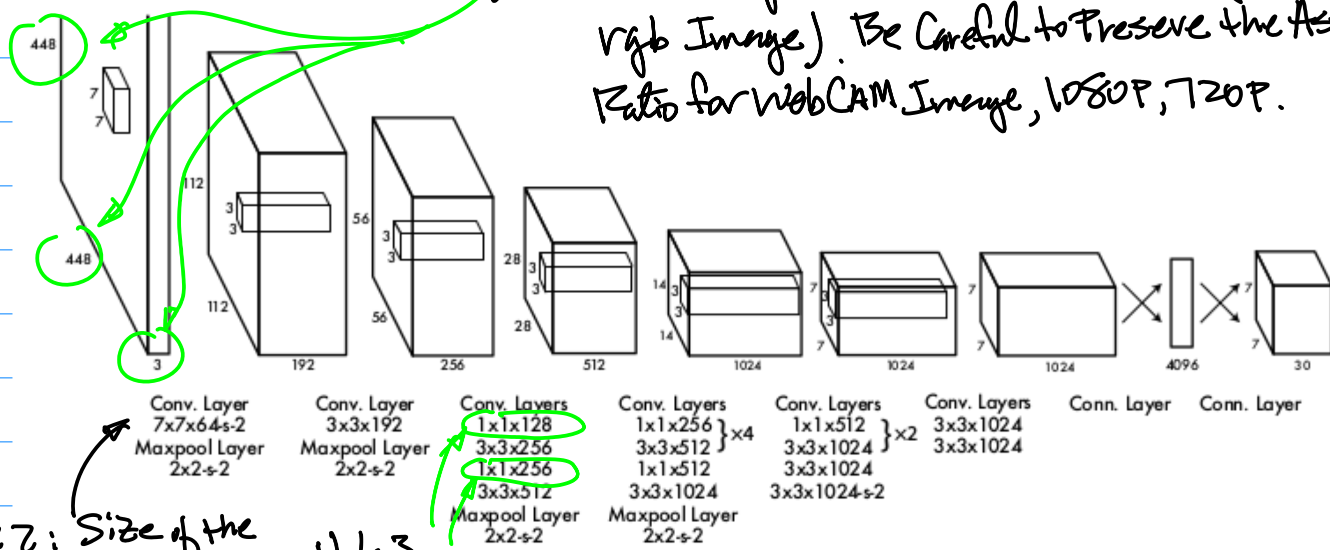
IOU truth ← "G.T." B.B.

pred ← Predicted B.B.

YOLO

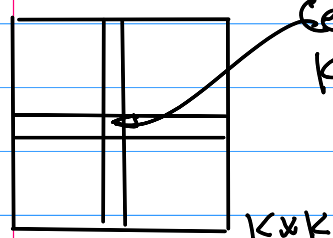
Example: Architecture :

Note 1. Input Image Resolution : 448x448x3 (Channels : rgb Image). Be Careful to Preserve the Aspect Ratio for WebCAM Image, 1080P, 720P.



Note 2 : Size of the Kernel : 7x7, 64 of them.

Note 3. 1x1 Convolution is utilized here



Center of the KxK Kernel

for KxK 2D Convolution, the extent of the convolution. "Spatial Information", Neighbouring pixels under the KxK kernel are Counted for (for feature extraction @ the center of the Kernel)