## 축하합니다! 통과하셨습니다!

**받은 학점** 100% **최신 제출물 학점** 90% **통과 점수:** 80% 이상

23h 54m 후에 과제를 다시 풀어보세요.

다음 항목으로 이동

**1.** You are building a 3-class object classification and localization algorithm. The classes are: pedestrian (c=1), car (c=2), motorcycle (c=3). What should $y$ be for the image below? Remember that "?" means "don't care", which means that the neural network loss function won't care what the neural network gives for that component of the output. Recall $y = [p_c, b_x, b_y, b_h, b_w, c_1, c_2, c_3]$.

1 / 1점

https://www.pexels.com/es-es/foto/mujer-vestida-con-falda-azul-y-blanca-caminando-cerca-de-la-hierba-verde-durante-el-dia-144474/

- $y = [1, 0.66, 0.5, 0.16, 0.75, 1, 0, 0]$
- $y = [1, 0.66, 0.5, 0.75, 0.16, 1, 0, 0]$
- $y = [1, ?, ?, ?, ?, 1, ?, ?]$
- $y = [1, 0.66, 0.5, 0.75, 0.16, 0, 0, 0]$

더 보기

맞습니다

Correct. $p_c = 1$ since there is a pedestrian in the picture. We can see that $b_x, b_y$ as percentages of the image are approximately correct as well $b_h, b_w$, and the value of $c_1 = 1$ for a pedestrian.

**2.** You are working on a factory automation task. Your system will see a can of soft-drink coming down a conveyor belt, and you want it to take a picture and decide whether (i) there is a soft-drink can in the image, and if so (ii) its bounding box. Since the soft-drink can is round, the bounding box is always square, and the soft-drink can always appear the same size in the image. There is at most one soft-drink can in each image. Here are some typical images in your training set: **1 / 1점**

The most adequate output for a network to do the required task is $y = [p_c, b_x, b_y, b_h, b_w, c_1]$. (Which of the following do you agree with the most?)

- False, since we only need two values $c_1$ for no soft-drink can and $c_2$ for soft-drink can.
- (selected) False, we don't need $b_h$, $b_w$ since the cans are all the same size.
- True, $p_c$ indicates the presence of an object of interest, $b_x, b_y, b_h, b_w$ indicate the position of the object and its bounding box, and $c_1$ indicates the probability of there being a can of soft-drink.
- True, since this is a localization problem.

더 보기

맞습니다
Correct. With the position $b_x, b_y$ we can completely characterize the position of the object if it is present. We should use only one additional logistic unit to indicate if the object is present or not.

**3.** When building a neural network that inputs a picture of a person's face and outputs N landmarks on the face (assume that the input image contains exactly one face), we need two coordinates for each landmark, thus we need 2N output units. True/False? **1 / 1점**

- False
- (selected) True

더 보기

맞습니다
Correct. Recall that each landmark is a specific position in the face's image, thus we need to specify two coordinates for each landmark.

**4.** You are working to create an object detection system, like the ones described in the lectures, to locate cats in a room. To have more data with which to train, you search on the internet and find a large number of cat photos. **1 / 1점**

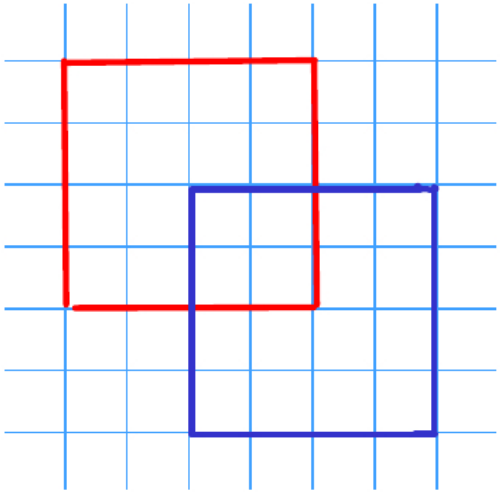
Which of the following is true about the system?

- We can't use internet images because it changes the distribution of the dataset.
- We should add the internet images (without the presence of bounding boxes in them) to the train set.
- We can't add the internet images unless they have bounding boxes.
- We should use the internet images in the dev and test set since we don't have bounding boxes.

더 보기

맞습니다
Correct. As this is a localization model, we also need the coordinates of the bounding boxes, not just the images.

**5.** What is the IoU between the red box and the blue box in the following figure? Assume that all the squares have the same measurements. **1 / 1점**



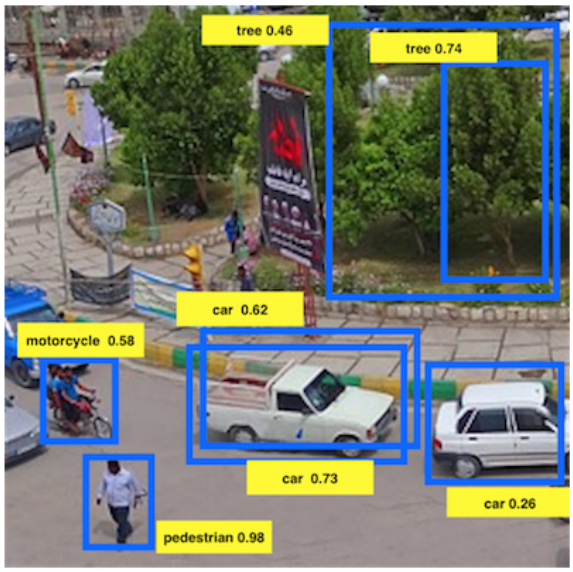
- $\frac{1}{2}$
- $\frac{1}{7}$
- $\frac{1}{8}$
- $\frac{1}{}$

4

더 보기

맞습니다
Correct. IoU is calculated as the quotient of the area of the intersection (4) over the area of the union (28).

**6.** Suppose you run non-max suppression on the predicted boxes below. The parameters you use for non-max suppression are that boxes with probability $\leq 0.4$ are discarded, and the IoU threshold for deciding if two boxes overlap is $0.5$. **0 / 1점**




Notice that there are three bounding boxes for cars. After running non-max suppression, only the bounding box of the car with 0.73 is kept from the three bounding boxes for cars. True/False? Choose the best answer.

- True. The non-maximum suppression eliminates the bounding boxes with scores lower than the ones of the maximum.
- False. Two bounding boxes corresponding to cars are left since their IoU is zero.
- False. All the cars are eliminated since there is a pedestrian with a higher score of 0.98.

더 보기

틀립니다
The algorithm doesn't act in that way; it also considers the IoU of the boxes.

**7.** Which of the following do you agree with about the use of anchor boxes in YOLO? Check all that apply. **1 / 1점**

- [ ] They prevent the bounding box from suffering from drifting.
- [x] Each object is assigned to an anchor box with the highest IoU inside the assigned cell.

Correct
Correct. This is the way we choose the corresponding anchor box.

- [ ] Each object is assigned to any anchor box that contains that object's midpoint.

Each object is assigned to the grid cell that contains that object's midpoint.

**Correct**
Correct. This is the way we choose the corresponding cell.

더 보기

맞습니다
Great, you got all the right answers.

**8.** We are trying to build a system that assigns a value of 1 to each pixel that is part of a tumor from a medical image taken from a patient. **1 / 1점**

This is a problem of localization? True/False

- (●) False
- ( ) True

더 보기

맞습니다
Correct. This is a problem of semantic segmentation since we need to classify each pixel from the image.

**9.** Using the concept of Transpose Convolution, fill in the values of **X**, **Y** and **Z** below. **1 / 1점**

(*padding = 1, stride = 2*)

Input: 2x2

| | |
|---|---|
| 1 | 2 |
| 3 | 4 |

Filter: 3x3

| | | |
|---|---|---|
| 1 | 1 | 1 |
| 0 | 0 | 0 |
| -1 | -1 | -1 |

Result: 6x6

| | | | | | |
|---|---|---|---|---|---|
| | 0 | 0 | 0 | X | |
| | Y | 4 | 2 | 2 | |
| | 0 | 0 | 0 | 0 | |
| | -3 | Z | -4 | -4 | |
| | | | | | |

- X = 0, Y =-1, Z = -7
- X = 0, Y = 2, Z = -1
- X = 0, Y = -1, Z = -4
- (●) X = 0, Y = 2, Z = -7

더 보기

맞습니다
Correct.

**10.** Suppose your input to a U-Net architecture is $h$ x $w$ x $3$, where 3 denotes your number of channels (RGB). What will be the dimension of your output ? **1 / 1점**

- (●) $h$ x $w$ x $n$ where n = number of output classes
- $h$ x $w$ x $n$ where n = number of filters used in the algorithm
- $h$ x $w$ x $n$ where n = number of of output channels
- $h$ x $w$ x $n$ where n = number of input channels

더 보기

맞습니다