

for Computer

Vision ICAT3260-

3003

1. Subsampling

Points for the section: 6

- ! Answer the following six (yes/no) questions that are related to the same piece of code.

Look at the code below and answer the question, correct (yes) or no.

```
import numpy as np
def downsample_by_two (img_in):
    h_input, w_input, c_input = img_in.shape
    img_lr = np.zeros(shape=(h_input//2, w_input//2, c_input),
                      dtype=np.uint8)
    for j in range(0, h_input//2):
        for i in range(0, w_input//2):
            for c in range(0, c_input):
                pixel1 = img_in[i*2,j*2,c].astype(np.uint32)
                pixel2 = img_in[i*2+1,j*2,c].astype(np.uint32)
                pixel3 = img_in[i*2,j*2+1,c].astype(np.uint32)
                pixel4 = img_in[i*2+1,j*2+1,c].astype(np.uint32)
                sum = (pixel1 + pixel2 + pixel3 + pixel4) // 4
                img_lr[i,j,c] = sum.astype(np.uint8)
    return img_lr
```

Question: In this code, the variable 'sum' can also get negative (< 0) values.

- no
 yes

1 points

Machine Learning for
Computer Vision

Instructions

Exam section 1: Subsampling

Answered
questions: 0 / 6

Move to

Exam section 2: Question set 3

Answered
questions: 0 / 6

Move to

Exit preview

Look at the code below and answer the question, correct (yes) or no.

```
import numpy as np
def downsample_by_two (img_in):
    h_input, w_input, c_input = img_in.shape
    img_lr = np.zeros(shape=(h_input//2, w_input//2, c_input),
                      dtype=np.uint8)
    for j in range(0, h_input//2):
        for i in range(0, w_input//2):
            for c in range(0, c_input):
                pixel1 = img_in[i*2,j*2,c].astype(np.uint32)
                pixel2 = img_in[i*2+1,j*2,c].astype(np.uint32)
                pixel3 = img_in[i*2,j*2+1,c].astype(np.uint32)
                pixel4 = img_in[i*2+1,j*2+1,c].astype(np.uint32)
                sum = (pixel1 + pixel2 + pixel3 + pixel4) // 4
                img_lr[i,j,c] = sum.astype(np.uint8)
    return img_lr
```

Question: The code reads the input image pixels following the pattern of the attached image file.

The attachment of the question

PATTERN.PNG

- no
 yes

1 points



Learning for

Computer

Vision

ICAT3260-3003

2. Question set 3 (Randomized questions)

Points for the section: 6

Which one of the following filter kernels can be used for nearest-neighbor 2x2 image upsampling with a transposed convolution layer?

a) [-1, 0], [0, 1]
b) [1, 0], [0, 0]
c) [1 1], [1, 1]
d) [0, 1], [-1, 0]

a)
 b)
 c)
 d)

1 points

Machine Learning for Computer Vision

Instructions

Exam section 1: Subsampling

Answered
questions: 0 / 6Move
to

Exam section 2: Question set 3

Answered
questions: 0 / 6Move
to

Exit preview

You have a Yolo v2 tiny object detector that uses 5 bounding boxes per grid cell and has been trained with the VOC dataset that has 20 different classes. What is the output size (shape) of the model's last layer?

(B, 13, 13, 425)
 (B, 13, 13, 125)
 (B, 13, 13, 100)
 (B, 13, 13, 20)

1 points

Your object detector neural network has detected 5 objects with scores

Box 1: 0.99,
Box 2: 0.95,
Box 3: 0.90,
Box 4: 0.4, and
Box 5: 0.1.

The threshold for a positive match with ground truth is IoU > 0.5, and boxes 1, 2 and 5 reach this IoU > 0.5 with ground truth boxes. What is the *precision* of this object detector in this example?

1
 0.8
 0.6
 0.4

1 points

If your neural network training ends with a training log like this

```
Epoch 96/100  
training accuracy: 0.7095 - loss: 1.1807 - val_accuracy: 0.4565 -  
val_loss: 1.7404  
Epoch 97/100  
training accuracy: 0.7099 - loss: 1.1735 - val_accuracy: 0.4590 -  
val_loss: 1.7413  
Epoch 98/100  
training accuracy: 0.7093 - loss: 1.1732 - val_accuracy: 0.4570 -  
val_loss: 1.7408  
Epoch 99/100  
training accuracy: 0.7090 - loss: 1.1723 - val_accuracy: 0.4585 -  
val_loss: 1.7443  
Epoch 100/100  
training accuracy: 0.7162 - loss: 1.1621 - val_accuracy: 0.4565 -  
val_loss: 1.7434
```

Which of the following adjustments to your neural network would likely lead to improving test-time accuracy?

- Increase strength of regularization
- Decrease strength of regularization
- Increase dense layer parameter count
- Increase number of training epochs

1 points

If the input image size is 32x32 with 3 color channels, and the image classification model consists of a single dense layer that classifies the input images to 10 classes, the linear layer trainable parameter count is

- 10240
- 30720
- 30730
- none of the above

1 points

Which one of the following code lines normalizes RGB pixel values that are originally in the range of [0, 255], into the range [-1, 1[?

- normalized = (original.astype('float32')-1) / 128
- normalized = (original.astype('float32')-1) / 256
- normalized = (original.astype('float32')-128) / 128
- normalized = (original.astype('float32')-128) / 256

1 points

[Move to Exam section 1. Subsampling](#)