

[질문]YOLO 코드 관련한 질문 드립니다. (2)

보낸 사람 김중범 (04.06 19:22)

받는 사람 nic503@naver.com

1/1명 읽음

```

1.
17 def convolutional(input_layer, filters_shape, downsample=False, activate=True, bn=True):
18     if downsample:
19         input_layer = ZeroPadding2D(((1, 0), (1, 0)))(input_layer)
20         padding = 'valid'
21         strides = 2
22     else:
23         strides = 1
24         padding = 'same'
25
26     conv = Conv2D(filters=filters_shape[-1], kernel_size = filters_shape[0], strides=strides,
27                  padding=padding, use_bias=not bn, kernel_regularizer=l2(0.0005),
28                  kernel_initializer=tf.random_normal_initializer(stddev=0.01),
29                  bias_initializer=tf.constant_initializer(0.))(input_layer)
30     if bn:
31         conv = BatchNormalization()(conv)
32     if activate == True:
33         conv = LeakyReLU(alpha=0.1)(conv)
34
35     return conv
36
37 def residual_block(input_layer, input_channel, filter_num1, filter_num2):
38     short_cut = input_layer
39     conv = convolutional(input_layer, filters_shape=(1, 1, input_channel, filter_num1))
40     conv = convolutional(conv, filters_shape=(3, 3, filter_num1, filter_num2))
41
42     residual_output = short_cut + conv
43     return residual_output
44
45 def upsample(input_layer):
46     return tf.image.resize(input_layer, (input_layer.shape[1] * 2, input_layer.shape[2] * 2), method='nearest')
47
48
49 def darknet53(input_data):
50     input_data = convolutional(input_data, (3, 3, 3, 32))
51     input_data = convolutional(input_data, (3, 3, 32, 64), downsample=True)
52

```

convolutional()에서 filters_shape의 원소 중 index 0과 -1만 사용하는 것 같은데

darknet53에서 사용할 때처럼 (3, 3, 3, 32) 이렇게 4개의 원소를 넣는 이유가 무엇인가요?

김중범

디지털혁신팀 / 매니저

서울특별시 용산구 한강대로23길 55

Mobile 010-4442-6684

Email 6363@hdc-dvp.com


주고받은 메일 2

이지용	[받은 메일함] RE: [질문]YOLO 코드 관련한 질문 드립니다. (2)	04.07 08:50
김중범	[보낸 메일함] [질문]YOLO 코드 관련한 질문 드립니다. (2)	04.06 19:22

