VIR: Originální příklad

Hynek Zamazal

7. listopadu 2021

Questions

We want to design a convolution network, which sorts the input pictures into three classes. The input picture is 6 pixels wide, 5 pixels tall and has three color channels. We start with a convolution layer with the kernel size of 3x3, step 1 and 5 output channels.

1. We want the output of the described convolution layer to have the same height and width as the input.
   a) What padding should we use to achieve this?
   b) How many times will the kernel be applied ("stamped")?
   c) What will be the dimensions and number of channels of the output?
2. After using this layer once (with the padding from task 1), we want to get the output.
   a) How many numbers are the output? What do they represent?
   b) How will we obtain these numbers? By what layers, with which parameters and what functions?

Solution

1. We want the output of the described convolution layer to have the same height and width as the input.

a) What padding should we use to achieve this?

We want the output to be 5x6 pixels. If in one dimension we have an input of size w, padding p and kernel of size k, the output size o after using step 1 is $o = (w + 2p) - k + 1$. If we plug our values into this and solve for p, we get 1.

a) How many times will the kernel be applied ("stamped")?

The output is 6 by 5 pixels, each pixel represents one stamp. Even though the picture has 3 channels, one stamp stamps through all 3 of them. So the kernel will be applied $6 \cdot 5 = 30$ times.

a) What will be the dimensions and number of channels of the output?

The output has $6 \cdot 5 = 30$ pixels and we have one "type of stamp"for each of the 5 output channels. That is $6 \cdot 5 \cdot 5 = 150$ values.

2. After using this layer once (with the padding from task 1), we want to get the output.

b) How many numbers are the output? What do they represent?

The output should be 3 numbers between 0 and 1 adding up together to 1. Each represents the probability of the picture belonging to one of the classes.

b) How will we obtain these numbers? By what layers, with which parameters and what functions?

We will flatten the output image and connect all the channels into one vector. This vector of 150 values will be passed through a linear layer with 3 outputs (matrix 3 by 150). We pass these three outputs through the softmax function normalizing them between 0 and 1, making them add up to 1.