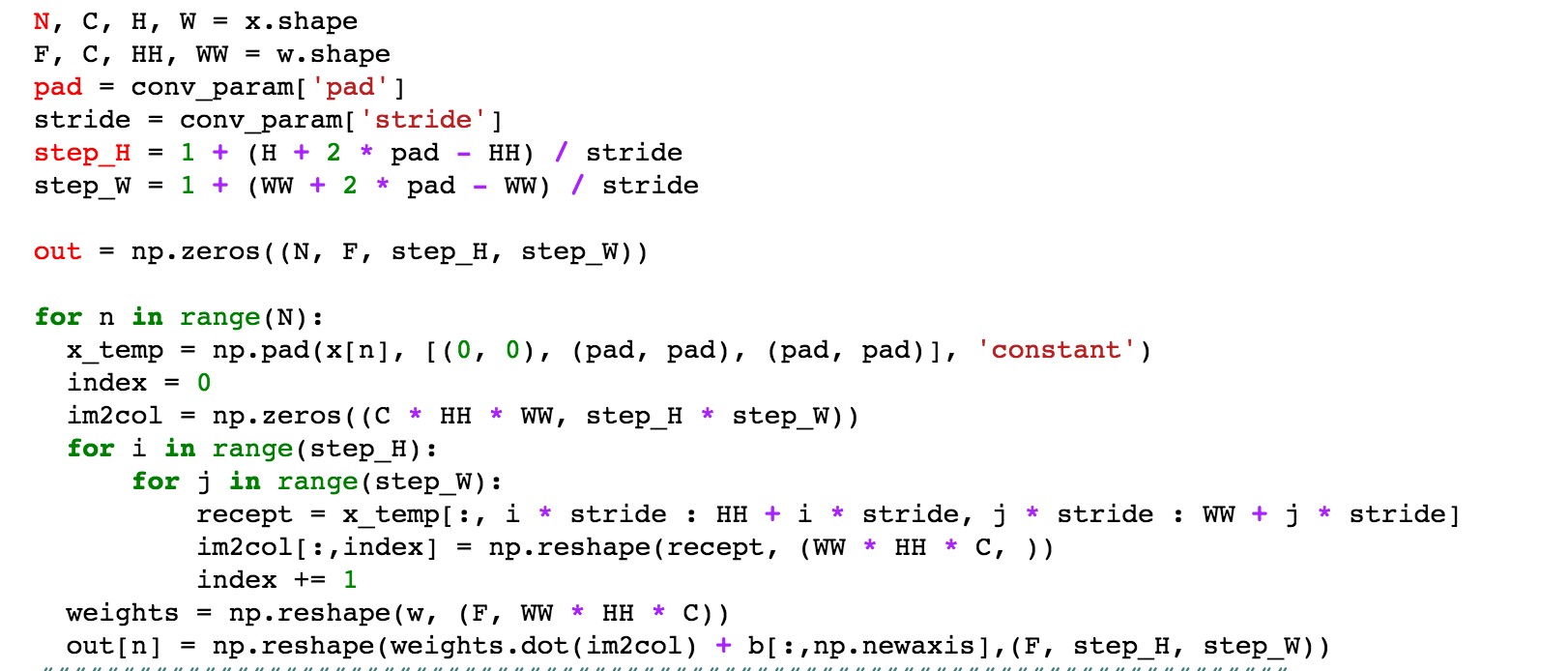
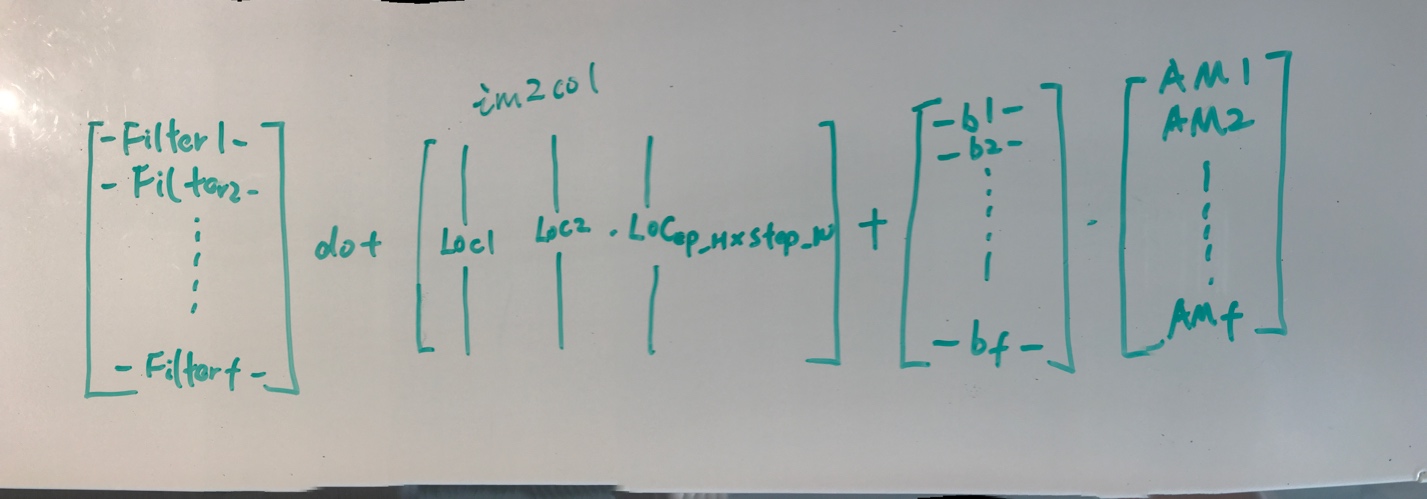


Notes from Profs

* As we slide the filter over the width and height of the input volume we will produce a 2-dimensional activation map that gives the responses of that filter at every spatial position.
* In general, setting zero padding to be P=(F−1)/2P=(F−1)/2 when the stride is S=1S=1 ensures that the input volume and output volume will have the same size spatially.
* All 96 neurons in each depth column (fiber，就像肌肉纤维一样，一根根平行得在柱形肌肉里) are connected to the same [11x11x3] region of the input, but of course with different weights
* One practical example is when the input are faces that have been centered in the image. You might expect that different eye-specific or hair-specific features could (and should) be learned in different spatial locations. (means that need specific filters to a specific spatial location, like different type of eyes filters in usual eye areas) In that case it is common to relax the parameter sharing scheme, and instead simply call the layer a **Locally-Connected Layer**.



ConvNet Forward Pass Checklist

* Decide activation map’s width and length
* Iterate over n data sets, following is for a single data set
* Fill pad to the data set
* Set im2col as zero arrays, its size should be (# of all locations, vectorized filter parameter)
* Iterate over each location
* indexing corresponding location in x and reshape it
* set reshaped local x to corresponding im2col’s index. i.e. 4th location, then set the value to im2col(:, 4)
* reshape the weights (filter) to matrix
* dot product the weight and accumulated im2col, remember to plus bias and then reshape back to output shape

Convnet Backward Pass

