COMS0018: PRACTICAL2

Dima Damen
Dima.Damen@bristol.ac.uk

Bristol University, Department of Computer Science
Bristol BS8 1UB, UK

October 13, 2019

CIFAR-10

- Dataset size: 60,000 images
- Training split: 50,000 images
- Test split: 10,000 images (1000 from each class) **(balanced)**

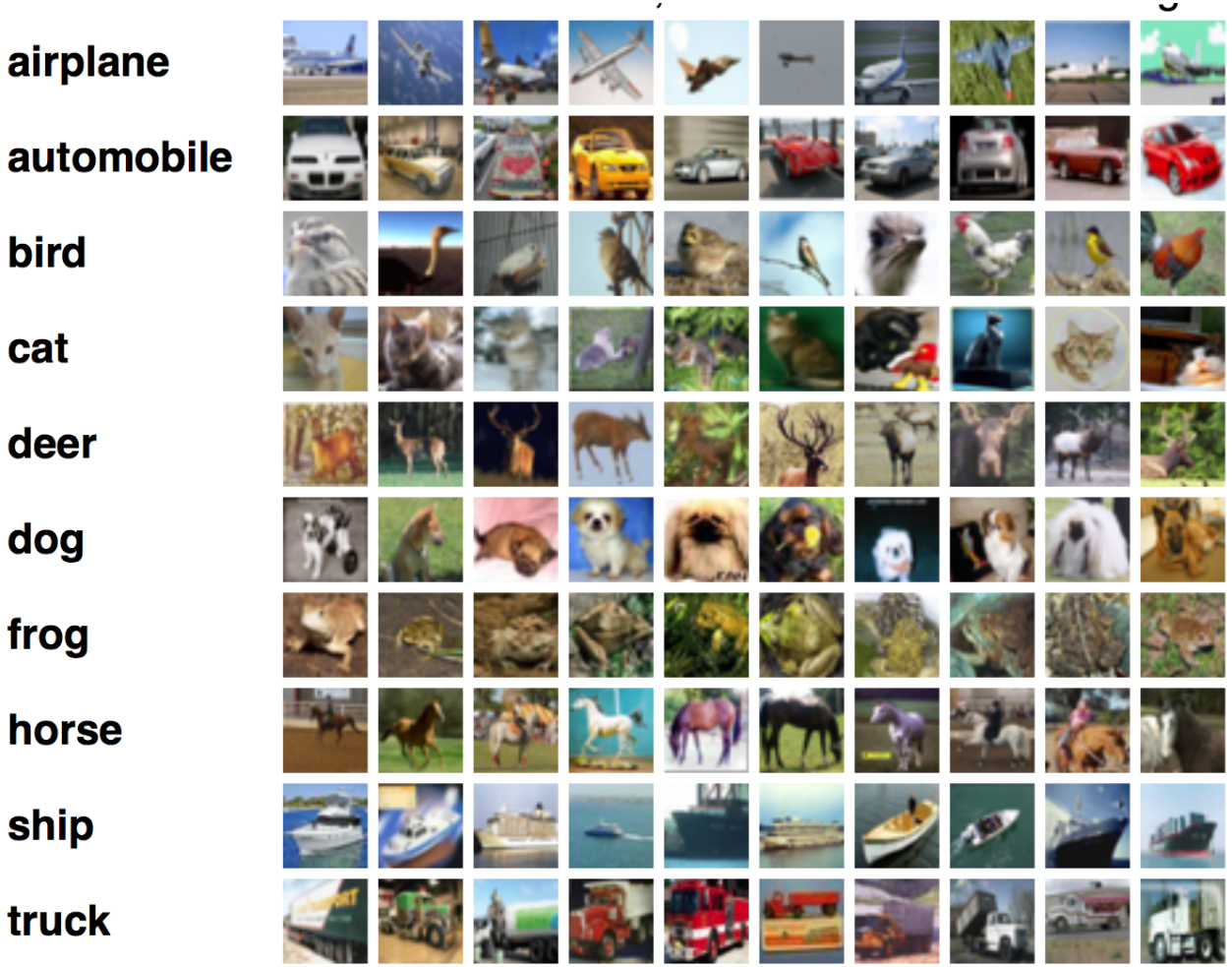
[1] http://groups.csail.mit.edu/vision/TinyImages/

CIFAR-10

- Dataset size: 60,000 images
- Training split: 50,000 images
- Test split: 10,000 images (1000 from each class) **(balanced)**
- Input size: 32×32 RGB images - $32 \times 32 \times 3 = 3072$ **(tiny images)**
- These have been collected by Rob Fergus, Antonio Torralba and Bill Freeman from MIT in 2008[1]

[1] http://groups.csail.mit.edu/vision/TinyImages/

CIFAR-10



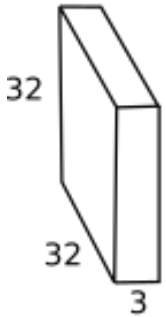
http://www.cs.toronto.edu/~kriz/cifar.html

CIFAR-10

- The current state-of-the-art results on CIFAR-10 are available at: `http://rodrigob.github.io/are_we_there_yet/build/classification_datasets_results.html#43494641522d3130`

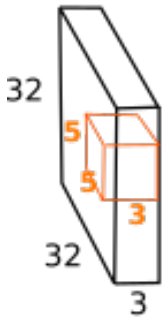
Our First Architecture

- We start with a 32x32x3 input $\mathbf{x}$



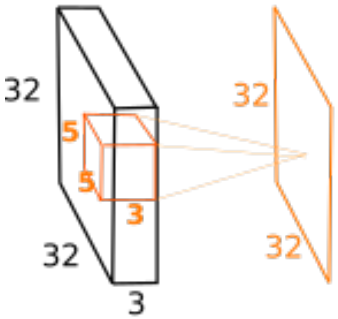
Our First Architecture

- In the first convolutional layer, one convolution filter is 5x5x3 = 75 weights



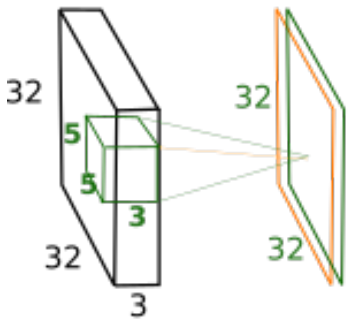
Our First Architecture

- By convolving it throughout the image, with padding,



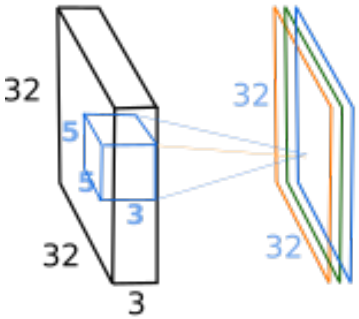
Our First Architecture

- We can have another filter of the same size, producing a different output layer



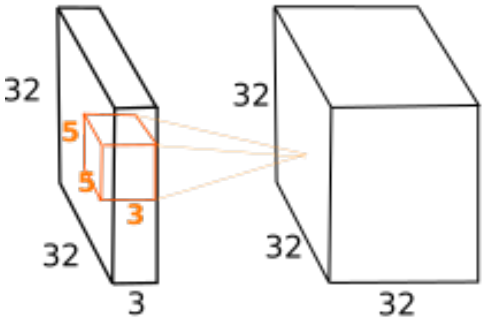
Our First Architecture

- And another one [until now 75*3 weights to learn]



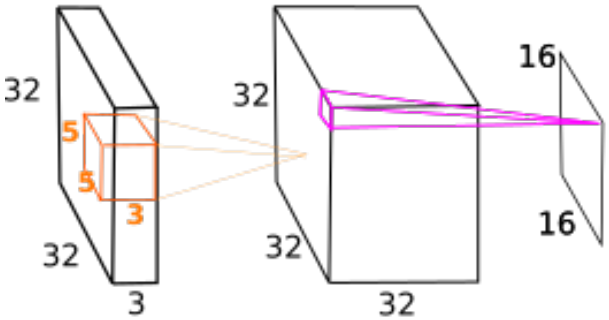
Our First Architecture

- We propose to have 32 of these = 2400 weights (CONV_1)



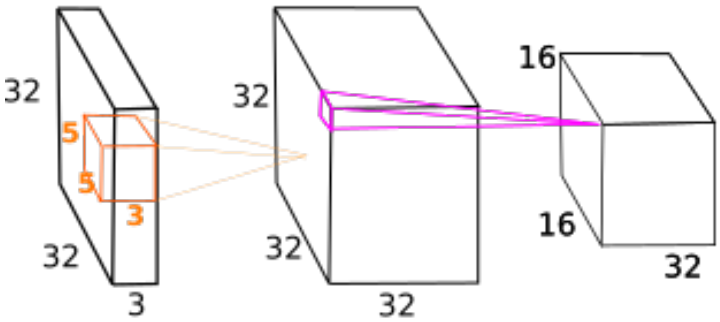
Our First Architecture

- Following an activation function, we perform max pooling on 2x2 grids



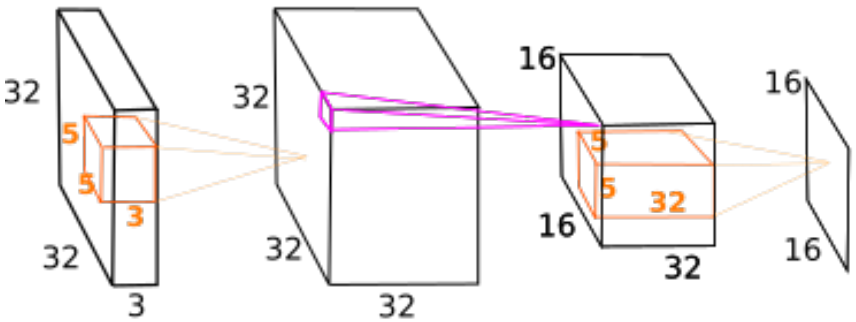
Our First Architecture

- This is applied for EACH of the 32 output layers



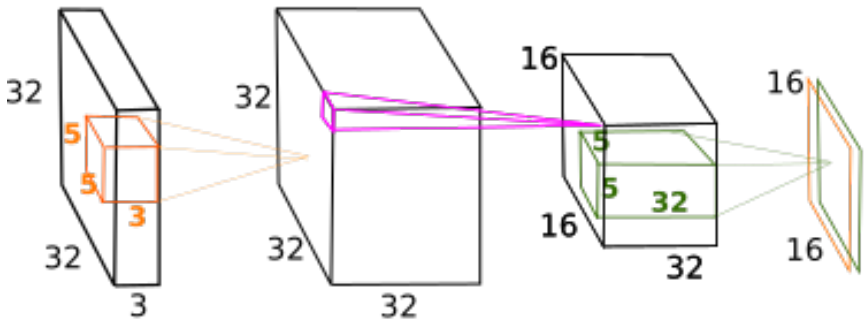
Our First Architecture

- Second conv layer will have 5x5x32 convolutional filter = 800 weights



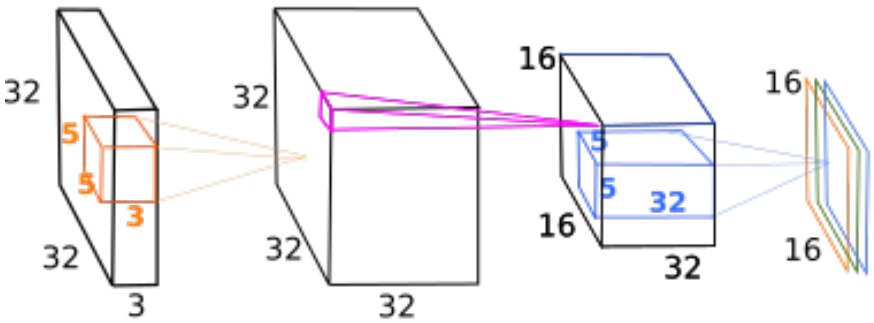
Our First Architecture

- We can have a second one of these filters



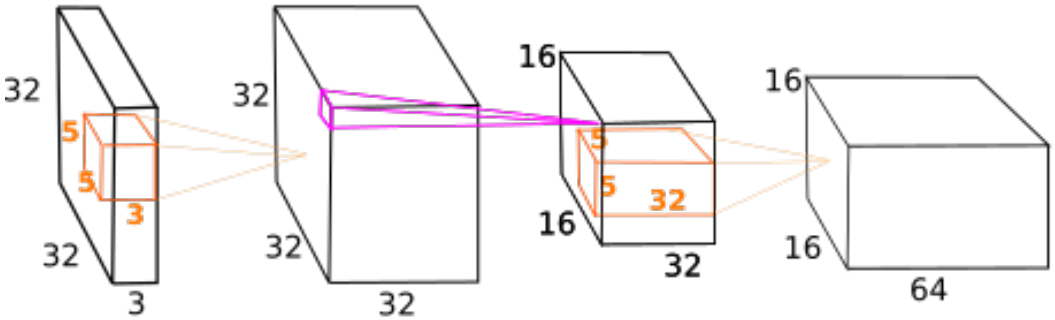
Our First Architecture

- And a third



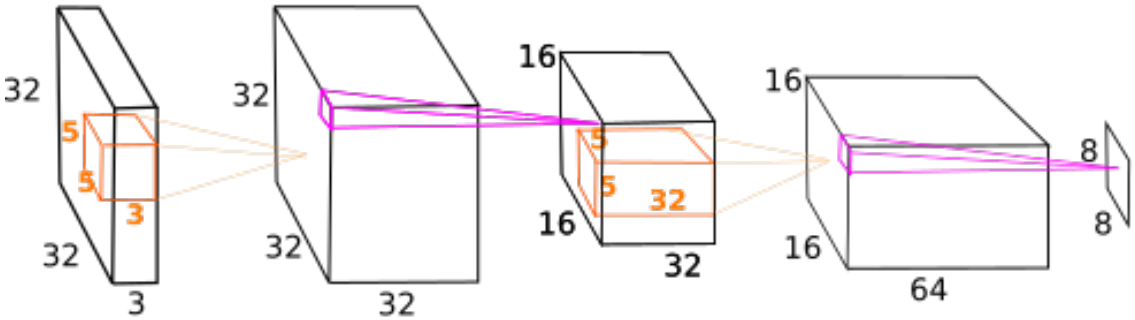
Our First Architecture

- We will have 64 of these = 51200, along with max-pooling



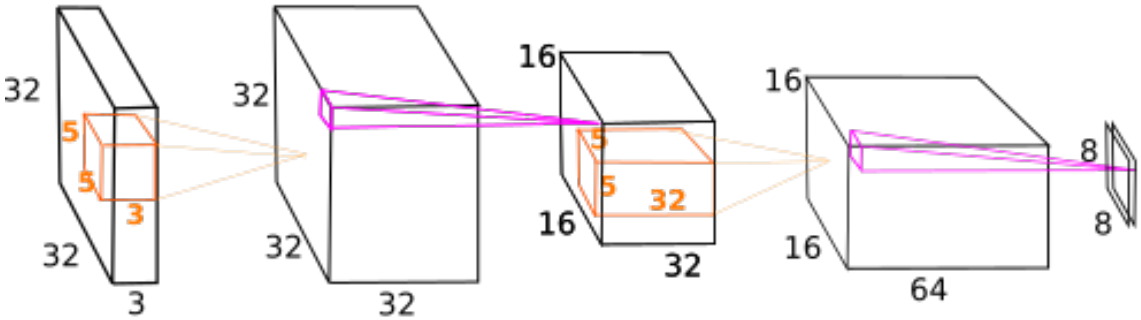
Our First Architecture

- Followed by max pooling, for each output layer



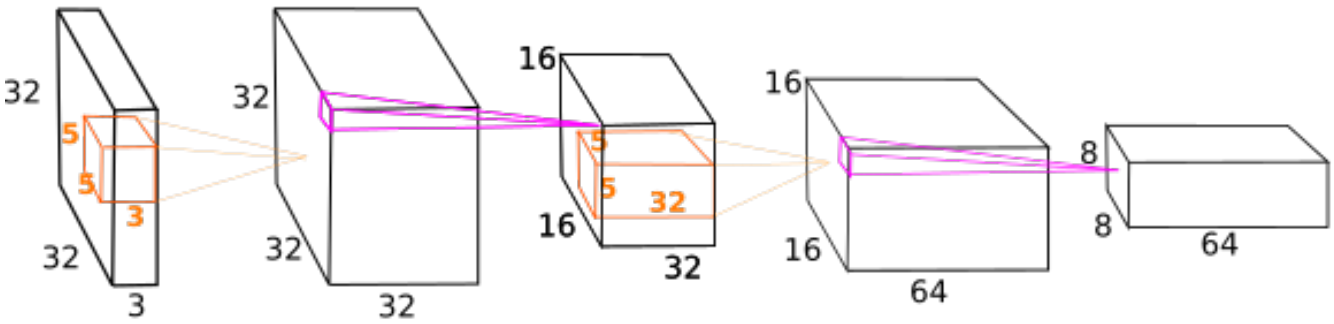
Our First Architecture

- Doing this for the second filter,



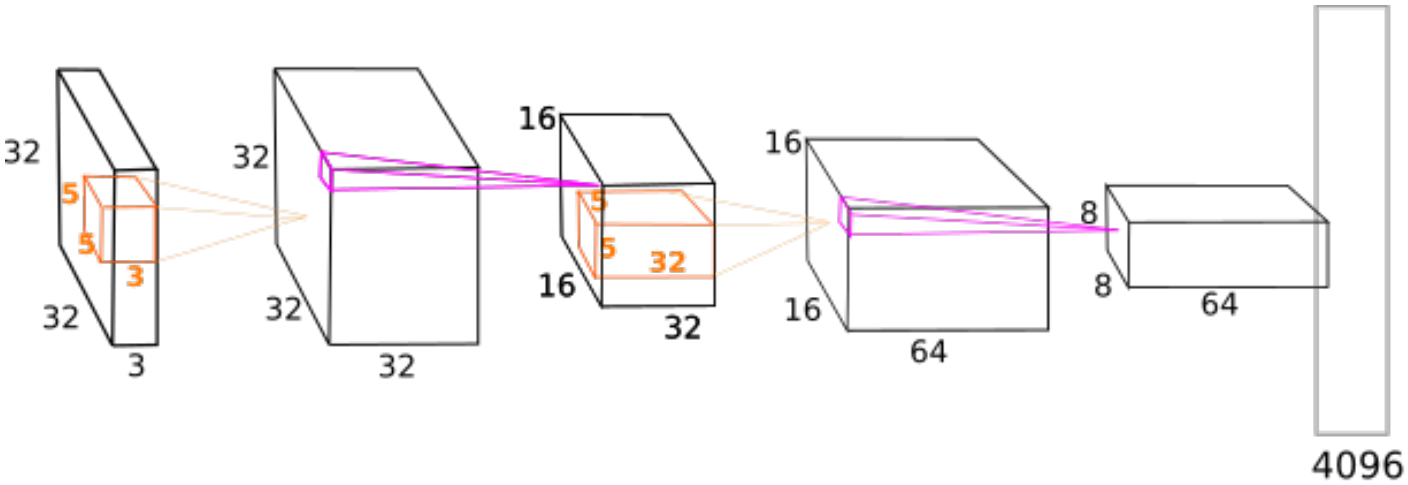
Our First Architecture

- And for all filters,



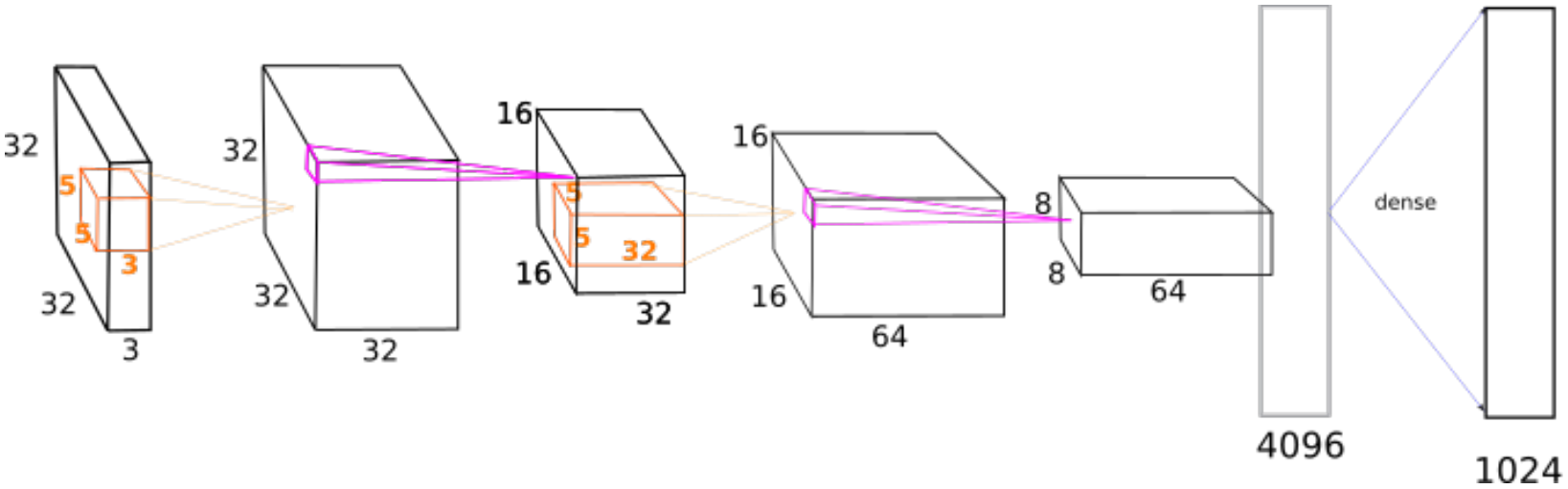
Our First Architecture

- Our output size is 4096 dimensions, which we reshape into 1D



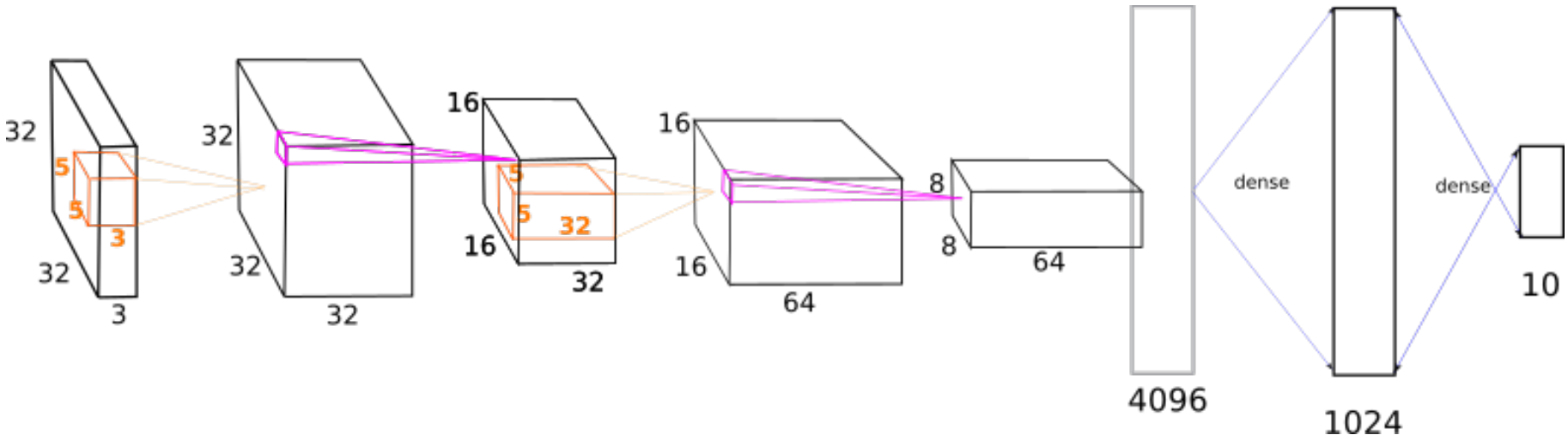
Our First Architecture

- Followed by 1 fully-connected layer, (4096x1024 weights)



Our First Architecture

- And a final fully connected layer into our 10 classes, (1024x10 weights)



The Lab Portfolio

- For this lab

Preparing Lab_2 Portfolio

Double check that you have saved the CSVs and log files we asked you to in tasks 13 & 14. Include your final version of `train_cifar.py` in the portfolio.

Zip these files up so your code, logs, and csvs follow the same structure:

```
Lab_2_<username>.zip
├── train_cifar.py
├── logs
|   ├── CNN_bs=256=0.001_run_0       <-- I chose to increase the batch size to 256 for task 14
|   |   ├── accuracy_test
|   |   |   └── events.out.tfevents.1567865348.bc4gpulogin1.bc4.acrc.priv.18893.3
|   |   ├── accuracy_train
|   |   |   └── events.out.tfevents.1567865335.bc4gpulogin1.bc4.acrc.priv.18893.1
|   |   ├── events.out.tfevents.1567865329.bc4gpulogin1.bc4.acrc.priv.18893.0
|   |   ├── loss_test
|   |   |   └── events.out.tfevents.1567865348.bc4gpulogin1.bc4.acrc.priv.18893.4
|   |   └── loss_train
|   |       └── events.out.tfevents.1567865335.bc4gpulogin1.bc4.acrc.priv.18893.2
|   └── CNN_bs=128_lr=0.001_run_0
|       ├── accuracy_test
|       |   └── events.out.tfevents.1567865348.bc4gpulogin1.bc4.acrc.priv.18893.3
|       ├── accuracy_train
|       |   └── events.out.tfevents.1567865335.bc4gpulogin1.bc4.acrc.priv.18893.1
|       ├── events.out.tfevents.1567865329.bc4gpulogin1.bc4.acrc.priv.18893.0
|       ├── loss_test
|       |   └── events.out.tfevents.1567865348.bc4gpulogin1.bc4.acrc.priv.18893.4
|       └── loss_train
|           └── events.out.tfevents.1567865335.bc4gpulogin1.bc4.acrc.priv.18893.2
├── loss-test.csv
├── loss-test-tweaked-hyperparameter.csv
├── loss-train.csv
└── loss-train-tweaked-hyperparameter.csv
```

And now....

READY....

STEADY....

GO...