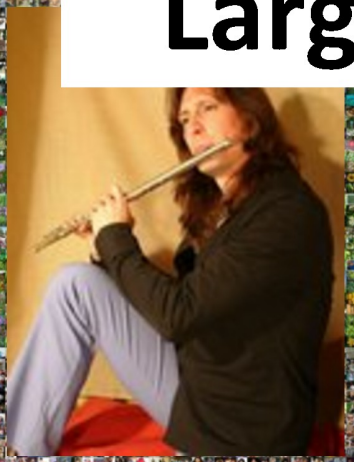
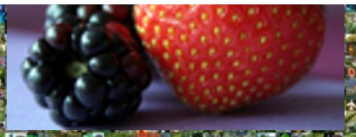
Deep Learning

Episode 6

How you *actually do it :)*

Large-scale recognition





CIFAR

- 60k images
- 10 classes (left)
- 32x32 RGB
- subclasses

aquatic mammals	beaver, dolphin, otter, seal, whale
fish	aquarium fish, flatfish, ray, shark, trout
flowers	orchids, poppies, roses, sunflowers, tulips
food containers	bottles, bowls, cans, cups, plates

The only problem: you only have 5k images

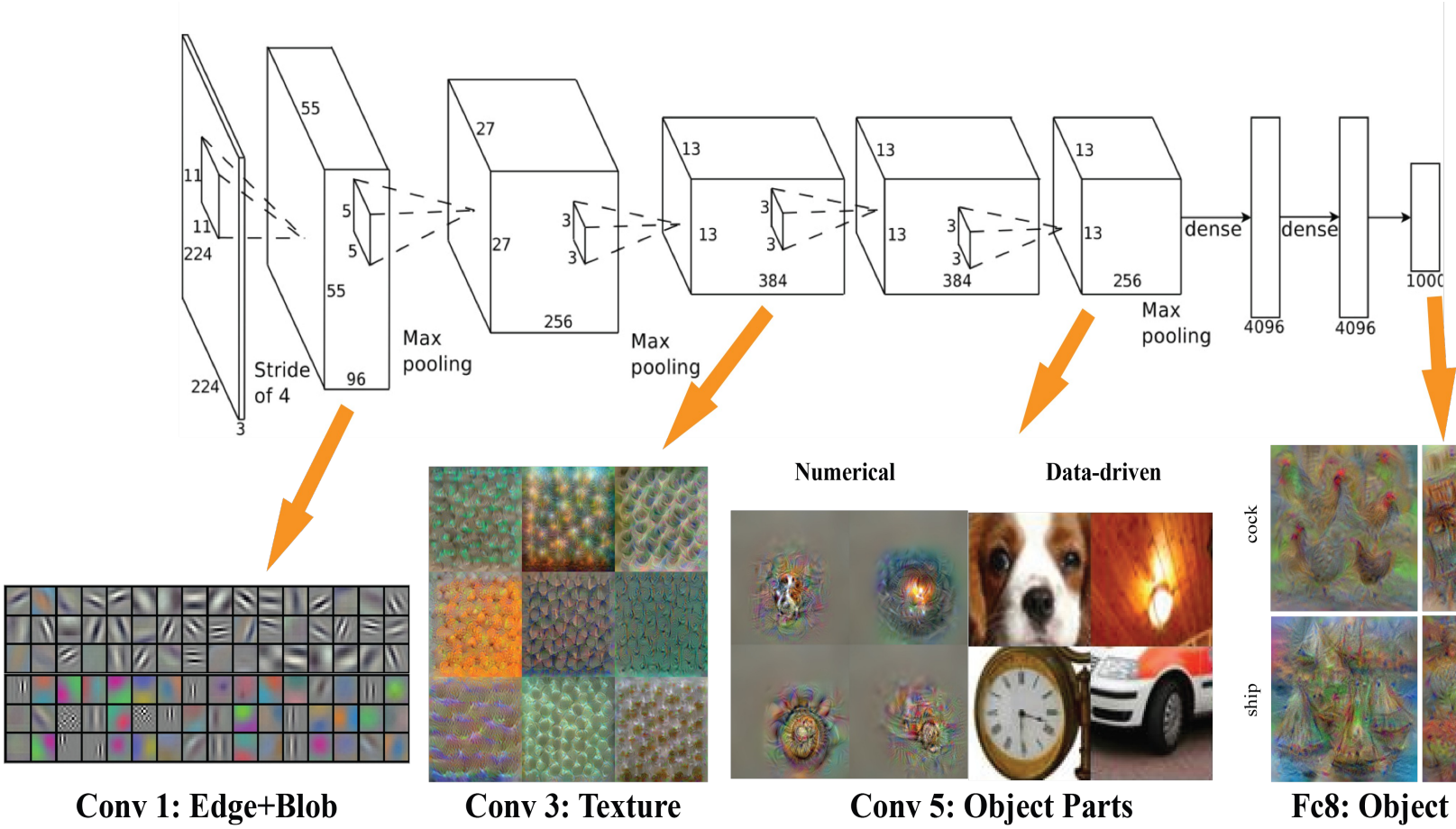
e.g. transport vs military aircrafts

Ideas?

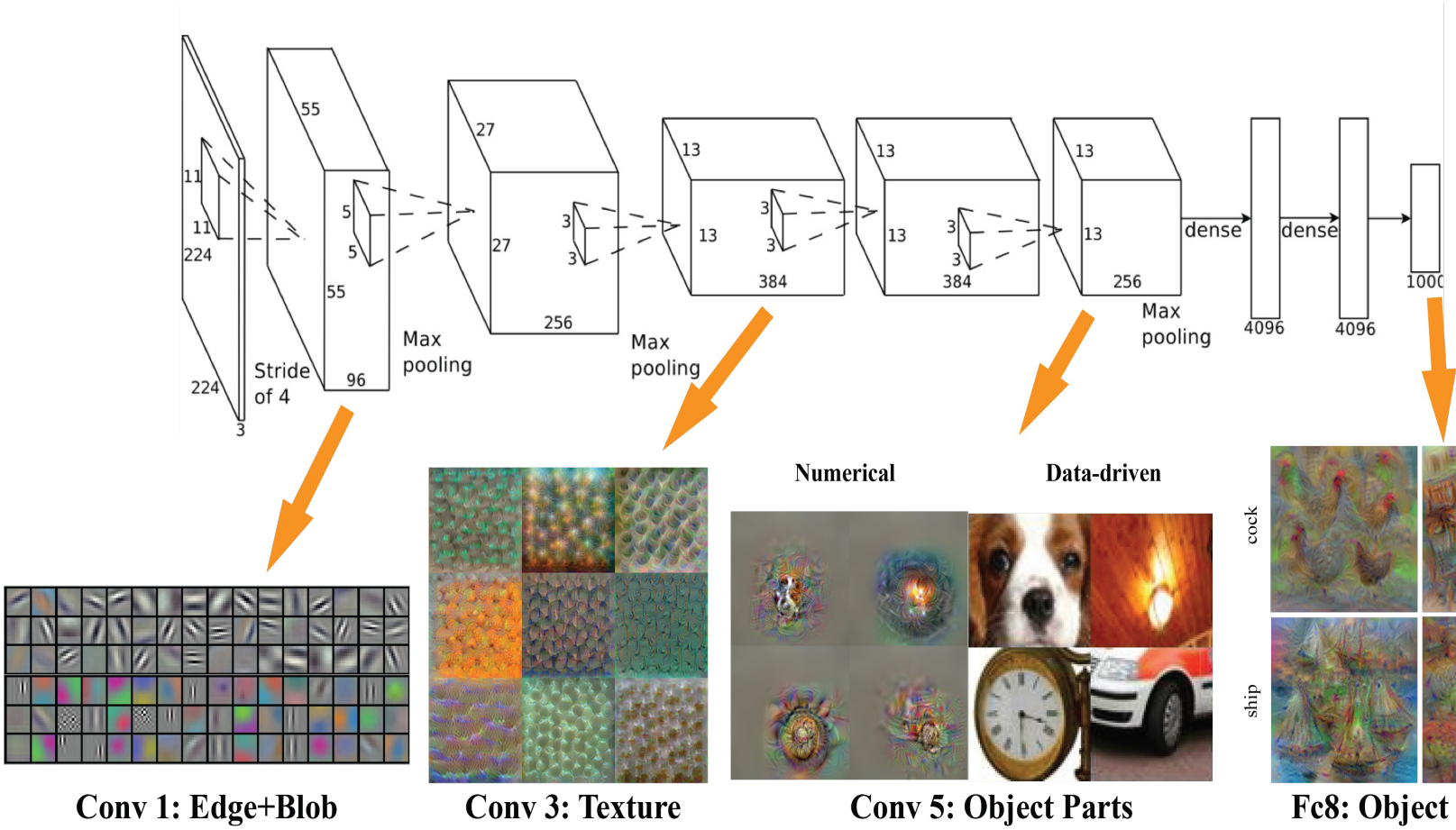
Regular solutions

- Regularize really hard
- Super small network
- Data augmentation
- Whatever

Feature learning



Feature learning



Idea: let's pre-train network on a larger dataset

Pre-training

- 1. Train a network on large dataset

cifar X

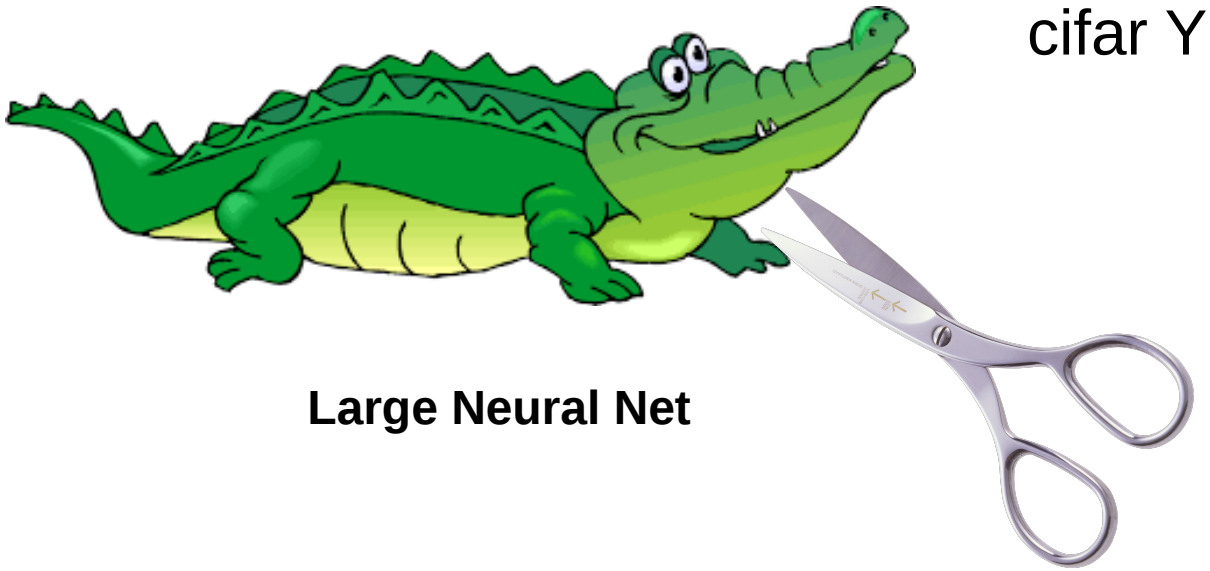
cifar Y

Large Neural Net

Pre-training

- 1. Train a network on large dataset
- 2. Take some intermediate layer

cifar X



cifar Y

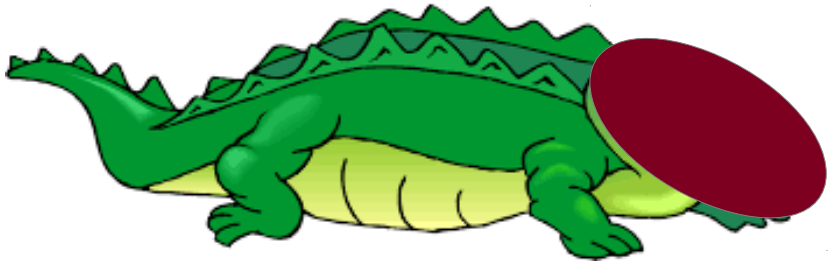
Large Neural Net

Pre-training

- 1. Train a network on large dataset
- 2. Take some intermediate layer

cifar Y

cifar X

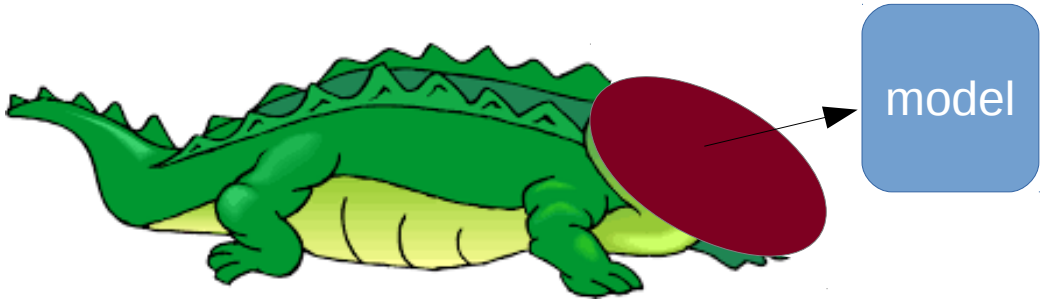


Large Neural Net

Pre-training

- 1. Train a network on large dataset
- 2. Take some intermediate layer
- 3. Build model on top of it

cifar X

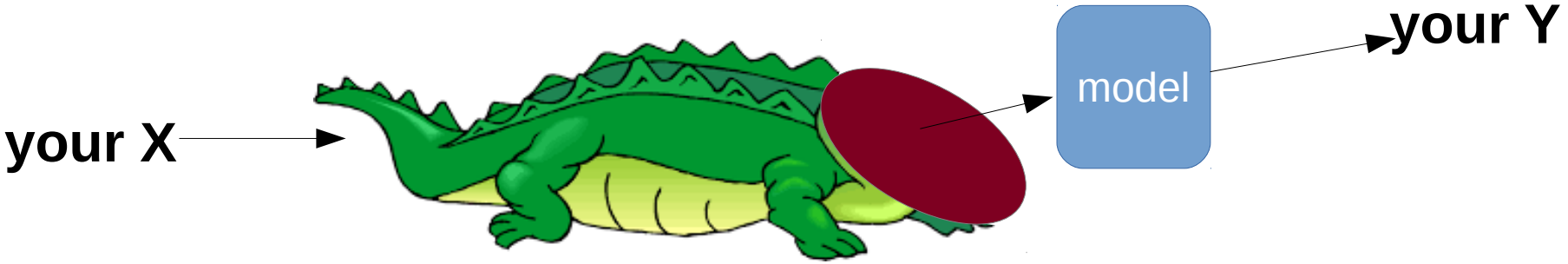


cifar Y

Large Neural Net

Pre-training

- 1. Train a network on large dataset
- 2. Take some intermediate layer
- 3. Build model on top of it
- 4. Train model for your objective



Large Neural Net

CIFAR

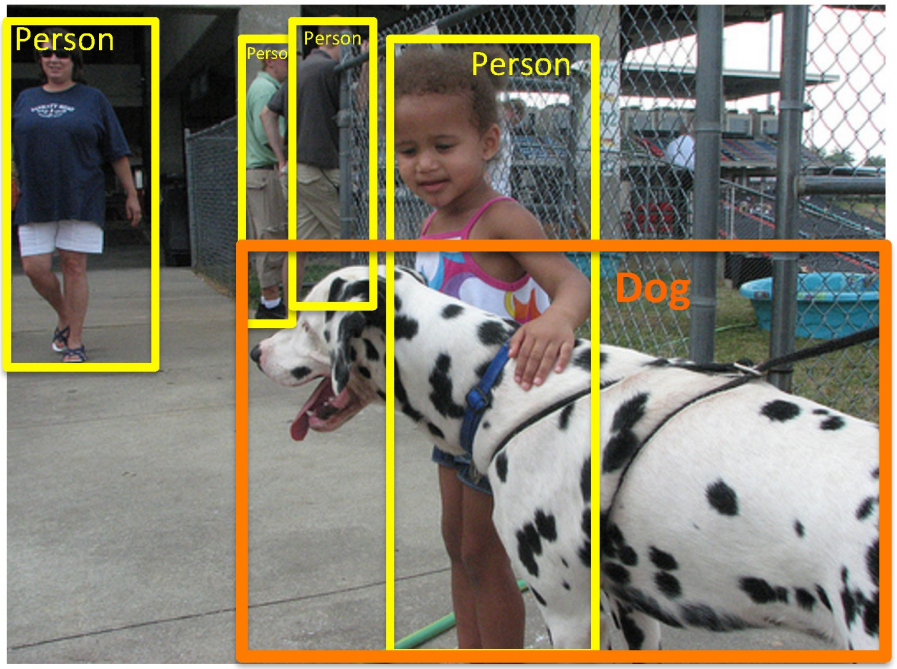
- 60k images
- 10 classes (left)
- 32x32 RGB
- subclasses

aquatic mammals	beaver, dolphin, otter, seal, whale
fish	aquarium fish, flatfish, ray, shark, trout
flowers	orchids, poppies, roses, sunflowers, tulips
food containers	bottles, bowls, cans, cups, plates

Pfft... weakling

IMAGENET Large Scale Visual Recognition Challenge (ILSVRC) 2010-2014

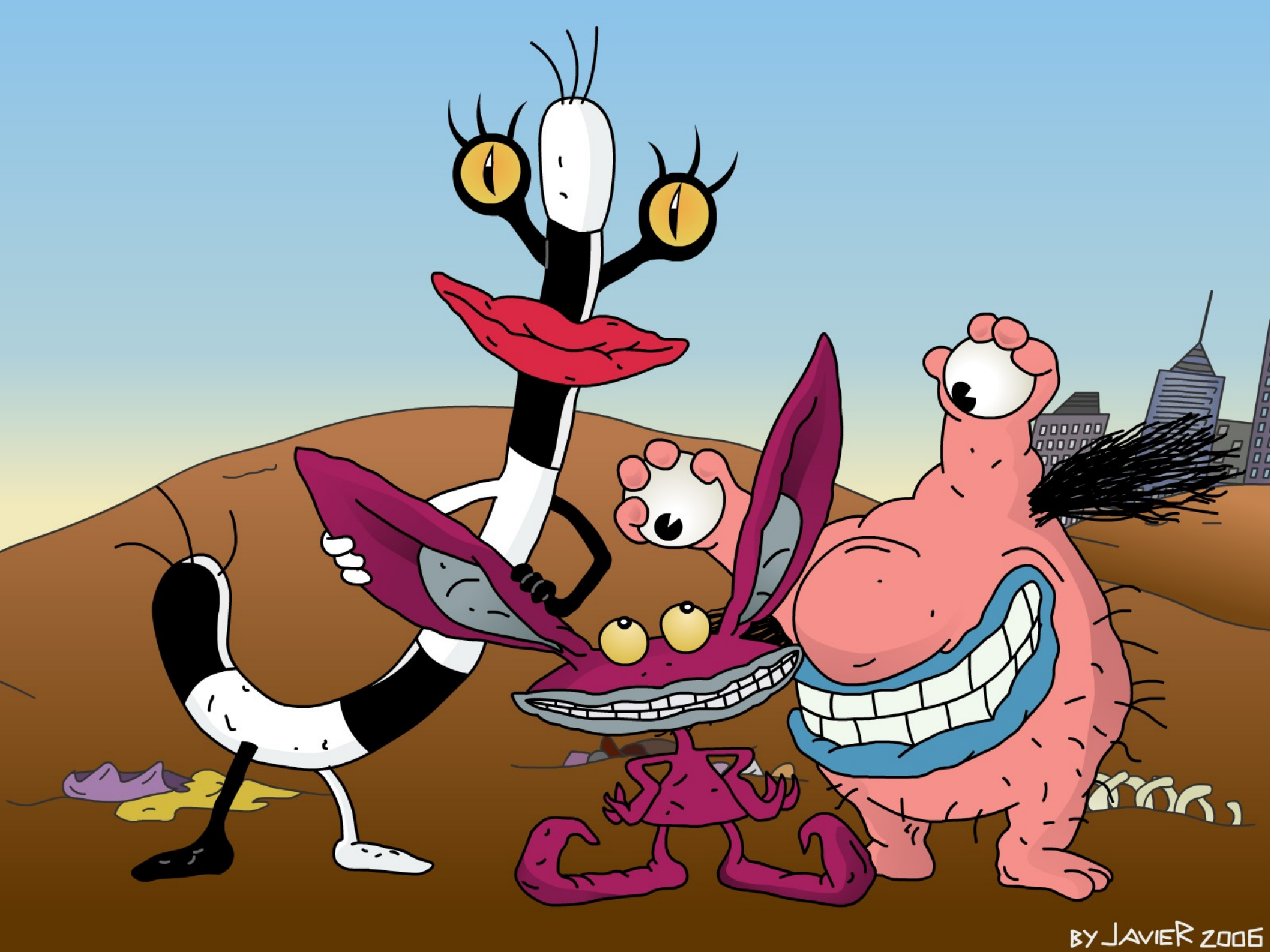
200 object classes	**456,567 images**	**DET**
1000 object classes	**1,431,167 images**	**CLS-LOC**



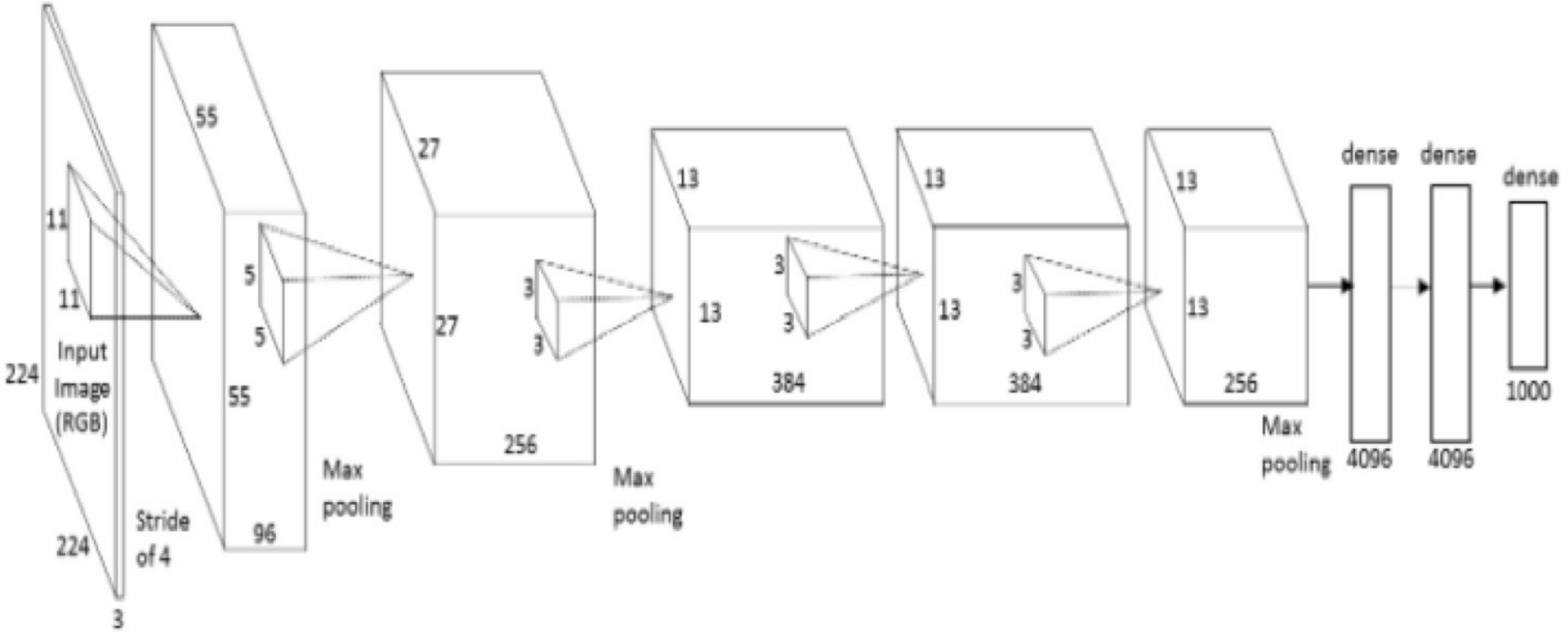
http://image-net.org/challenges/LSVRC/

Variety of object classes in ILSVRC

What kind of network will handle that?

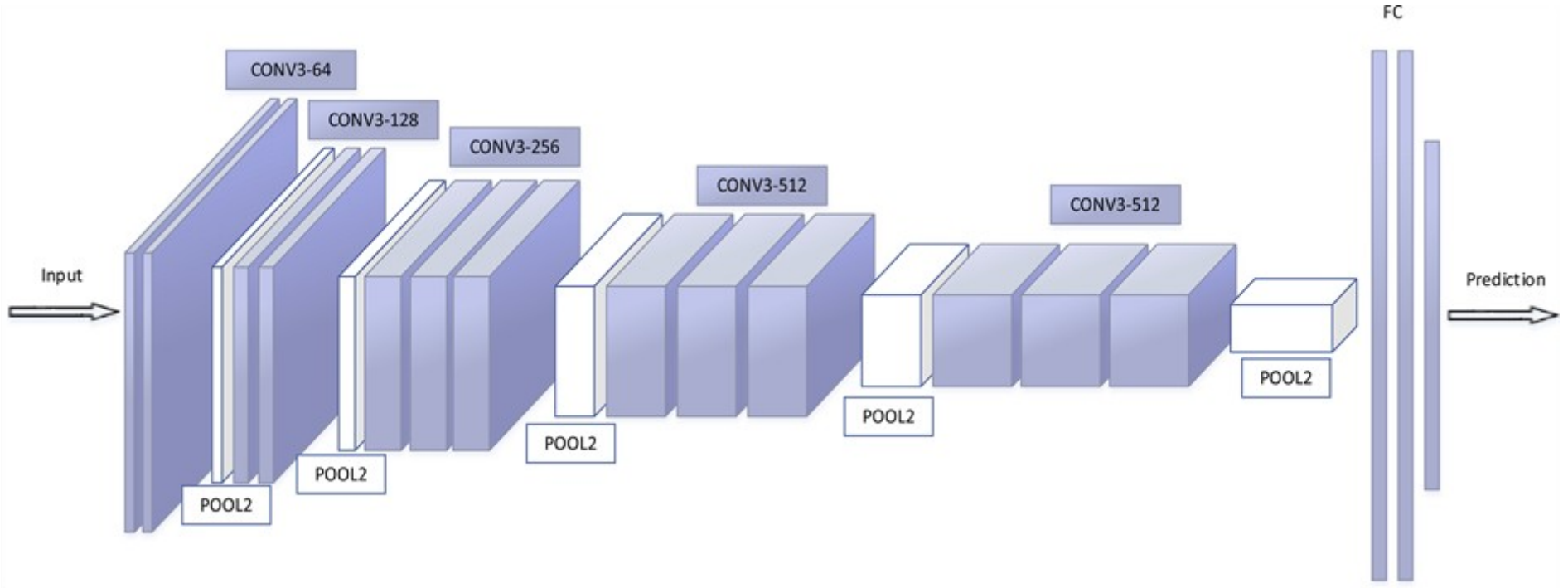

BY JAVIER 2006

Alexnet



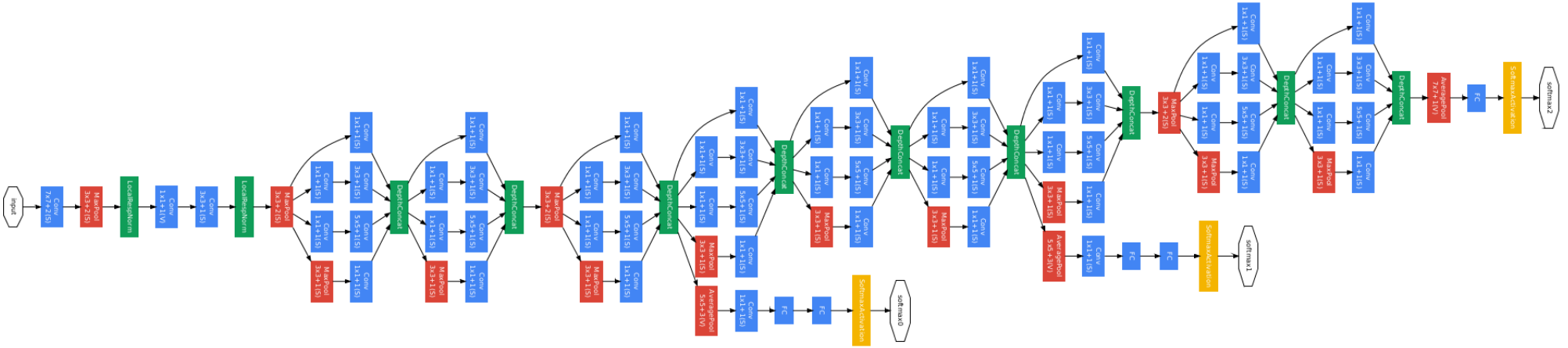
(Krizhevsky et al.)

VGG



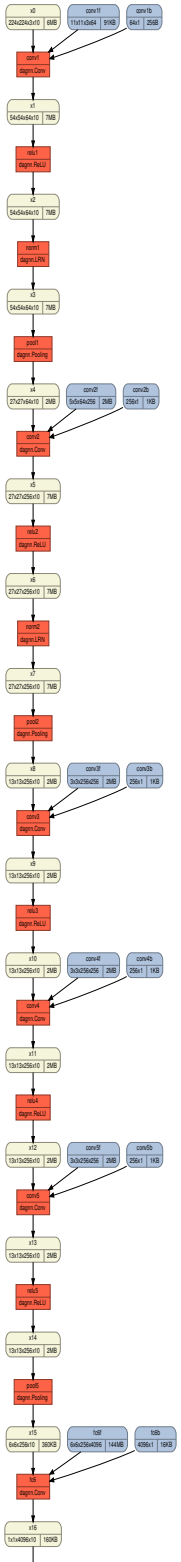
Karen Simonyan and Andrew Zisserman

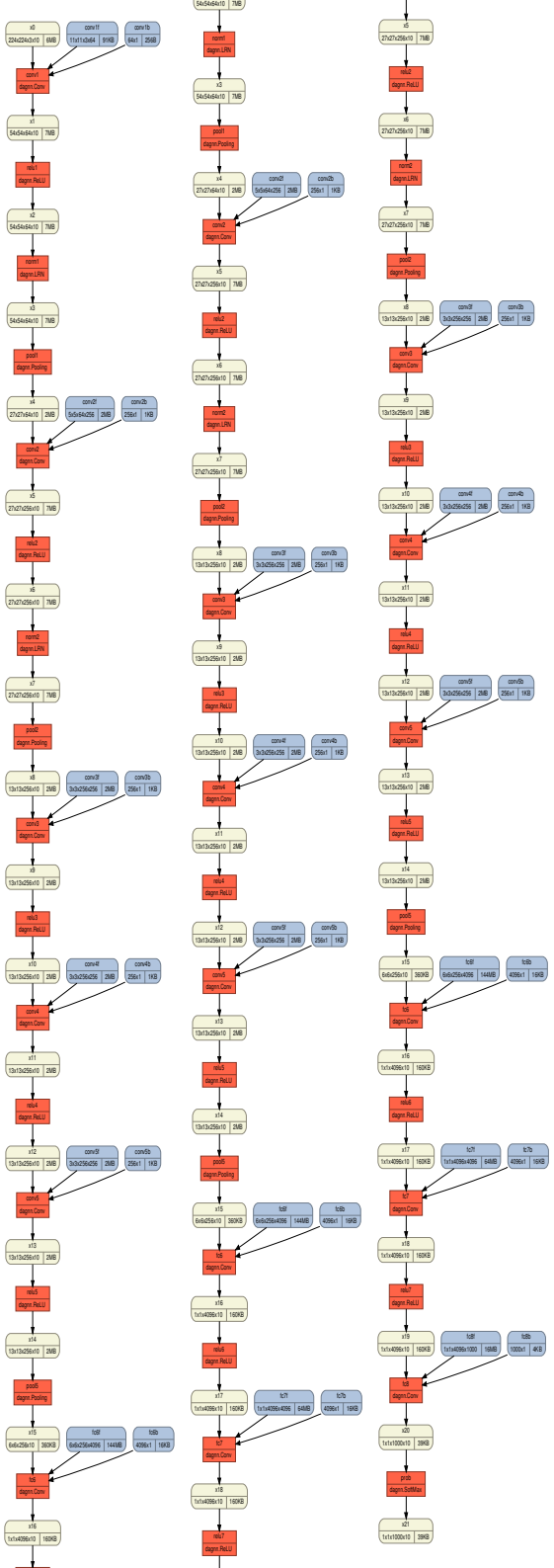
Googlenet (inception)



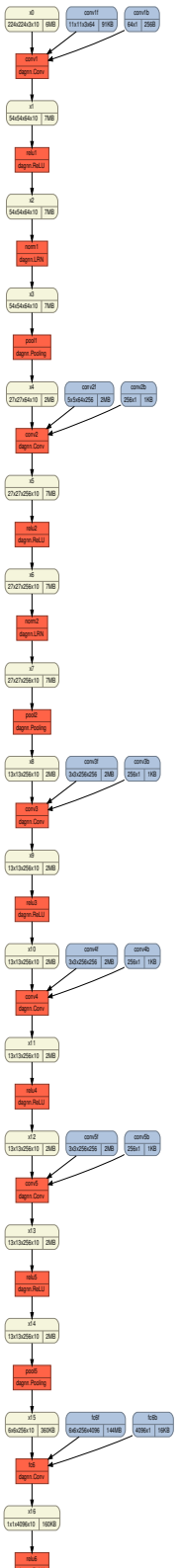
Szegedy et al

ResNet-152

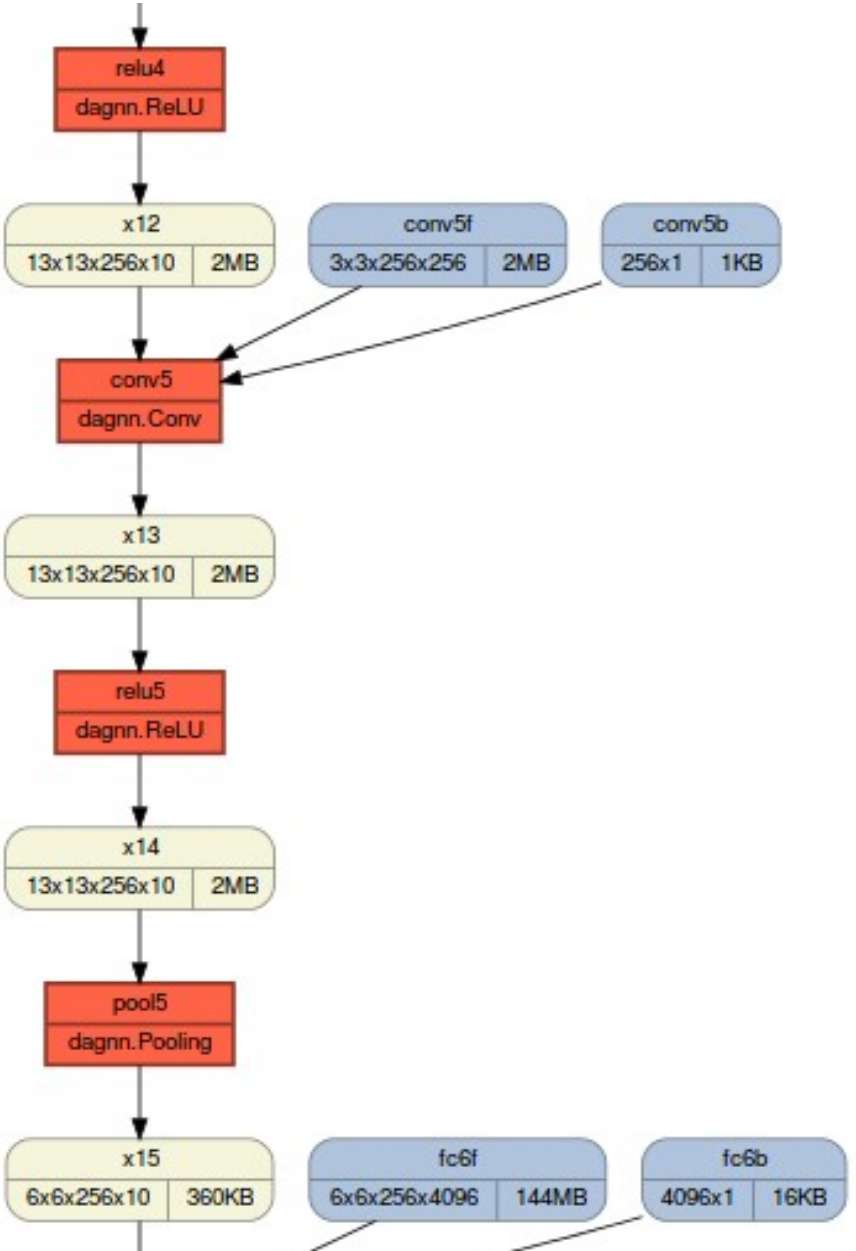




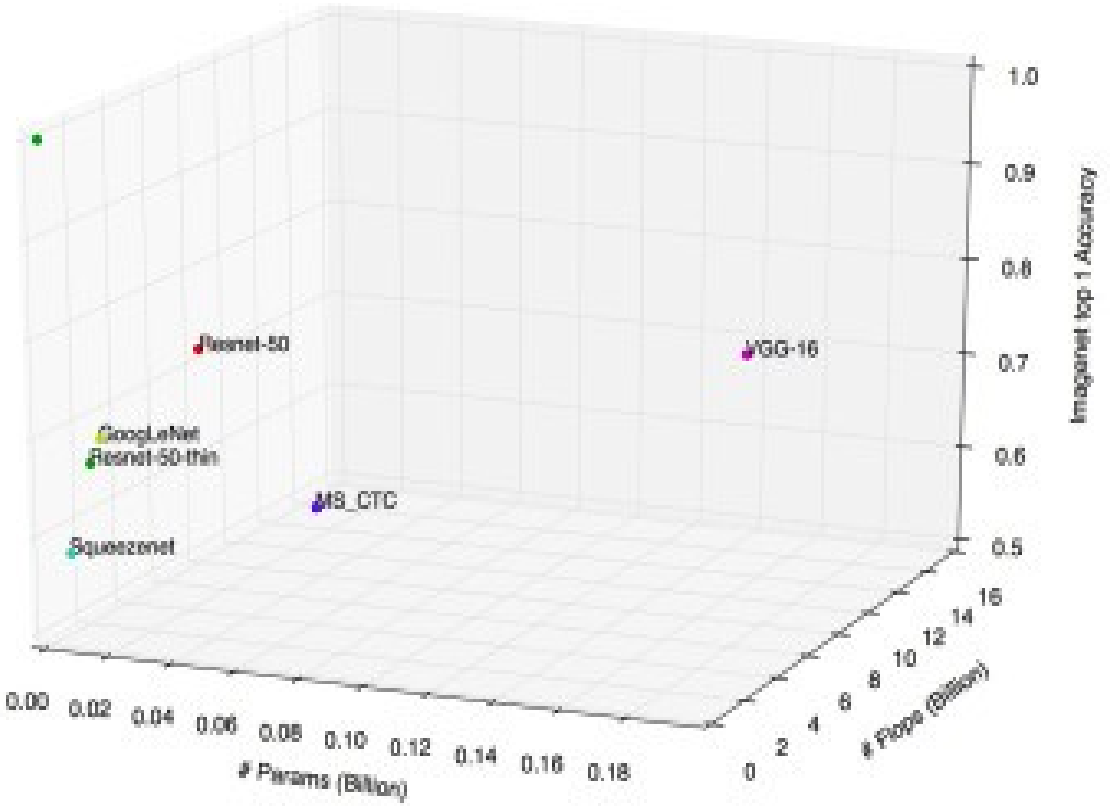
ResNet-152



ResNet-152


relu4
dagnn.ReLU
x12
13x13x256x10
2MB
conv5f
3x3x256x256
2MB
conv5b
256x1
1KB
conv5
dagnn.Conv
x13
13x13x256x10
2MB
relu5
dagnn.ReLU
x14
13x13x256x10
2MB
pool5
dagnn.Pooling
x15
6x6x256x10
360KB
fc6f
6x6x256x4096
144MB
fc6b
4096x1
16KB

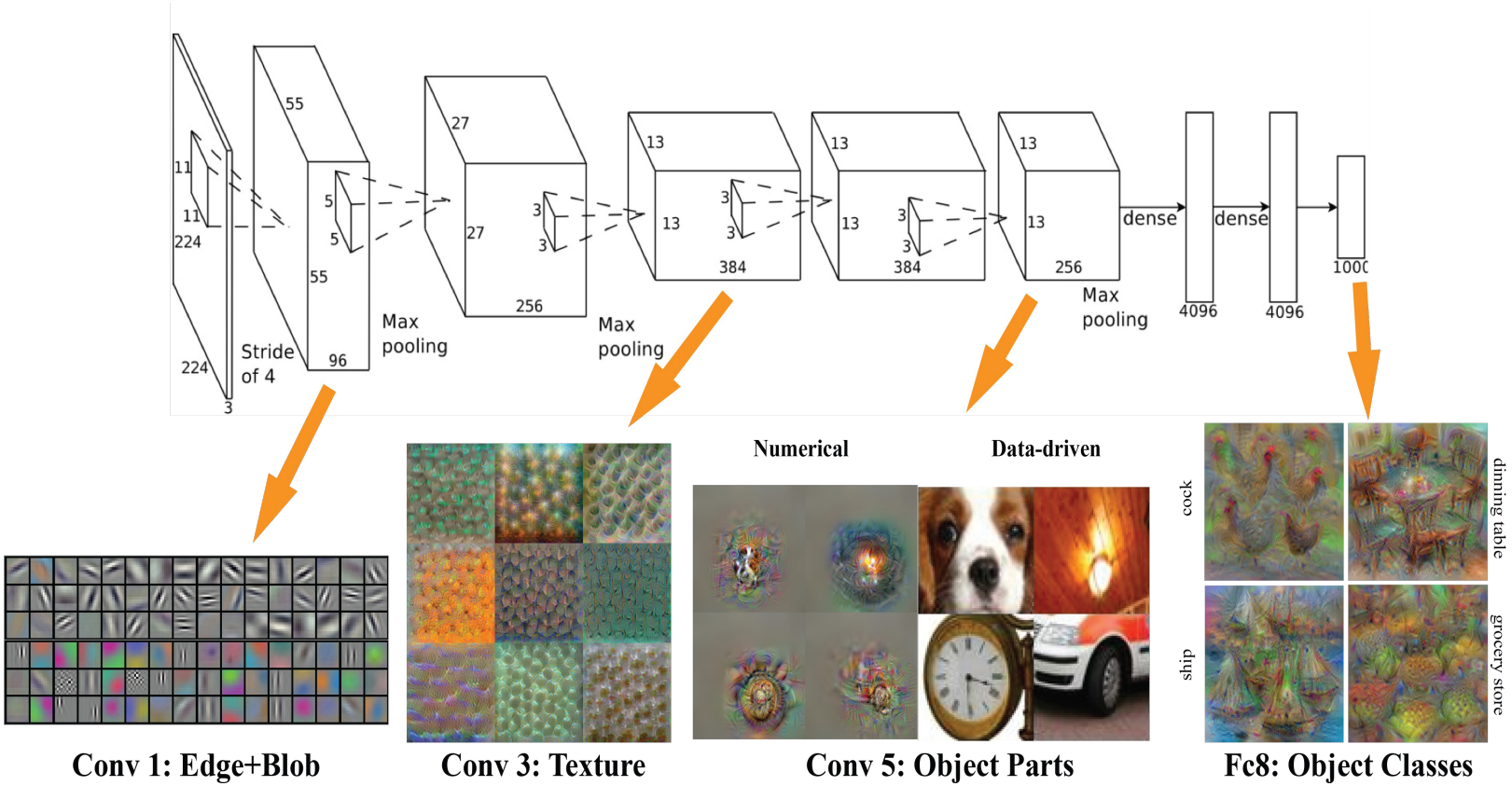
Monster stats



Can we use them?*

*for aircraft problem

Can we use them?



~ anywhere
even bran scans

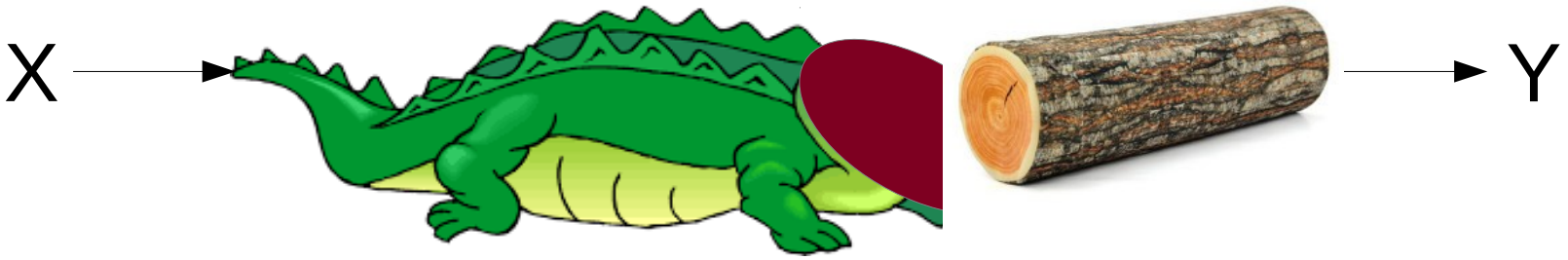
photos
any kinds

casual photos
i.e. not satelite maps

highly similar tasks

Reusing features

- Chop off “head”
- use “neck” as feature extractor
- Train ANY classifier
  - even random forest will do



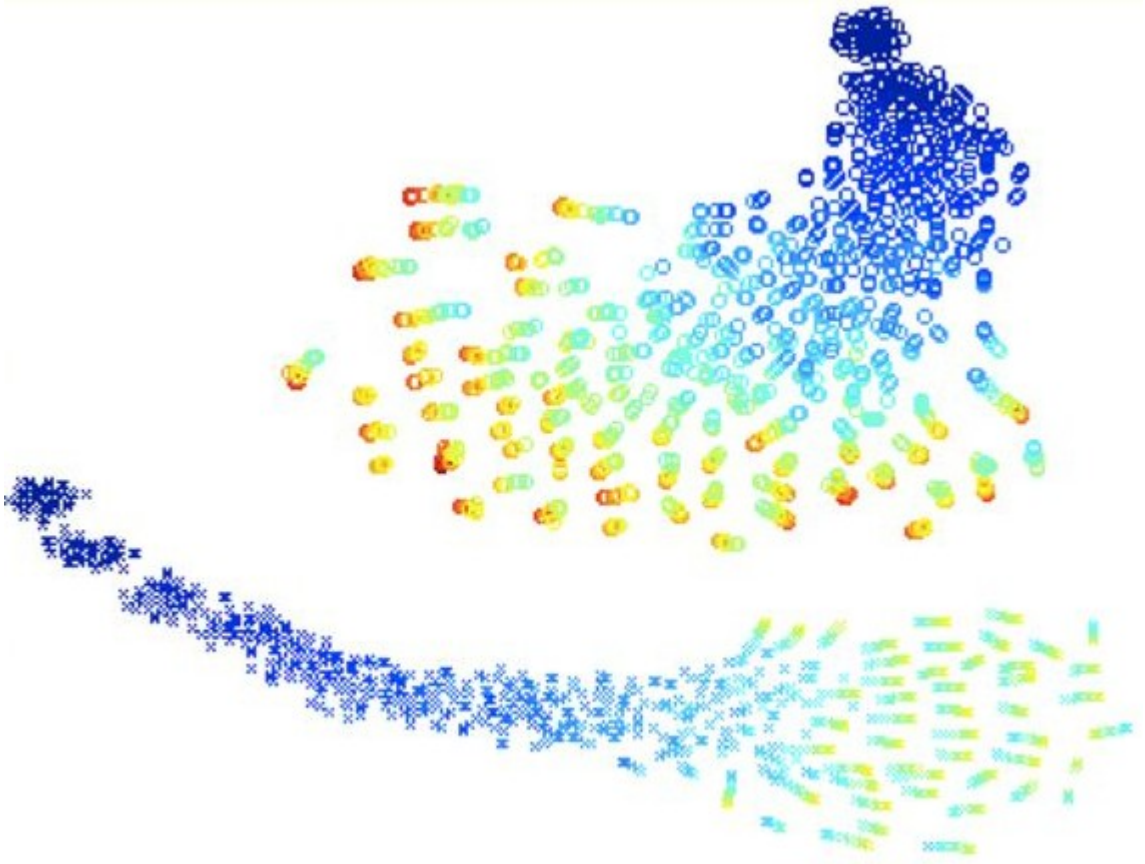
Large Neural Net

Fine-tuning

- Chop off “head”
- “freeze” body (consider constant)
- Build new neural network in it's place
- Train “head” only for several iteratons
- Un-freeze body and train full network

Large Neural Net

Comparison



- TSNE of network weights
- Crosses = pre-training, Rounds = no pre-training
- Color = epoch (hotter means higher)
- No overlap!

Other kinds of NNs

Fine-tuning not restricted to computer vision

- Using word2vec as pre-trained embedding
- Using skip-thought as pre-trained encoder-decoder
- Pre-training simple shallow networks on similar domains

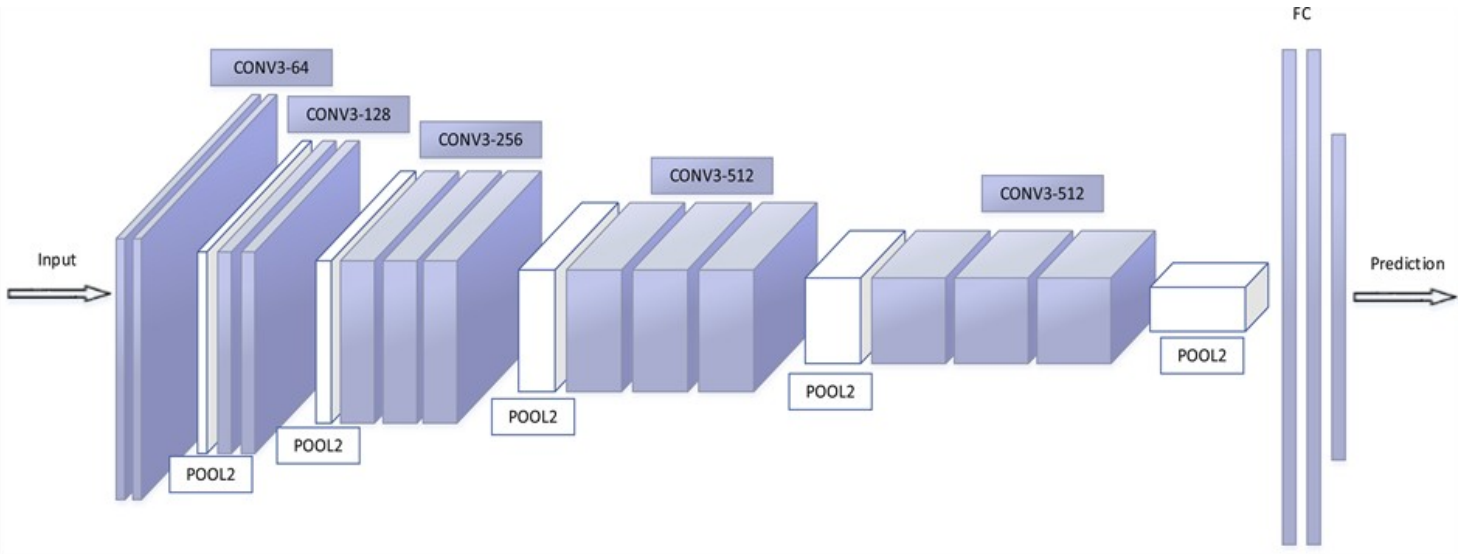
Alright, we did it!

But how do we fit large NN into an Iphone?

- That's simple!
  - We just train a smaller network
  - Or use a remote server with GPUs
  - Maybe use float16 instead of float32

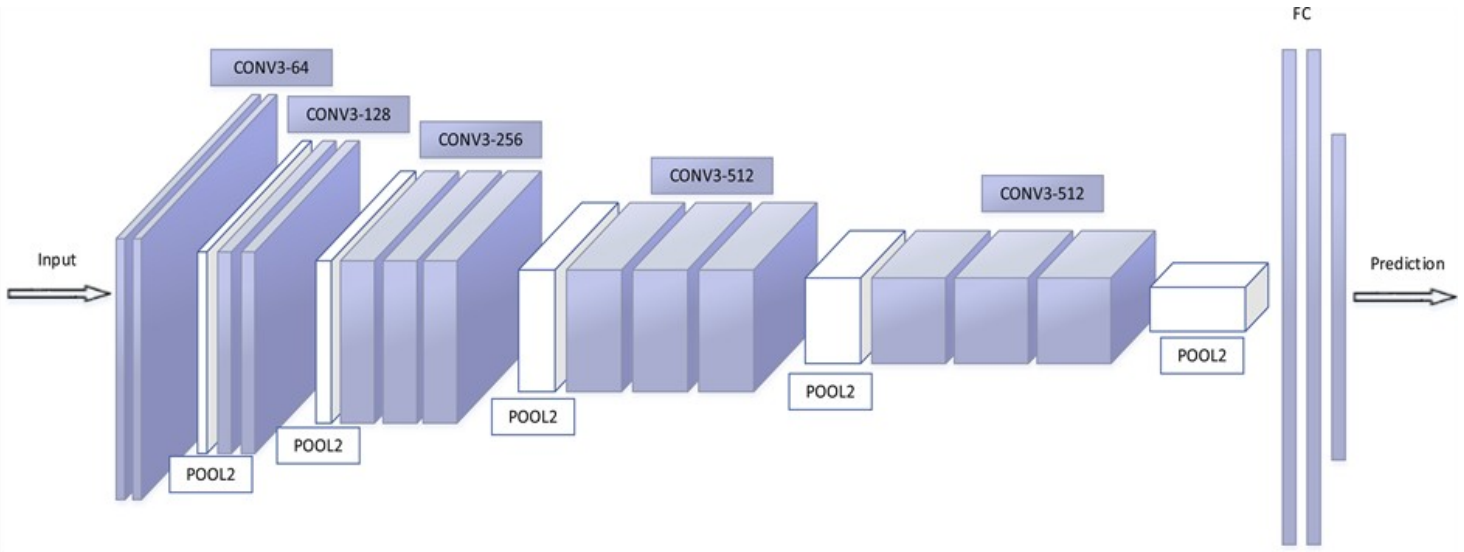
Can we do better?

Decomposition trick



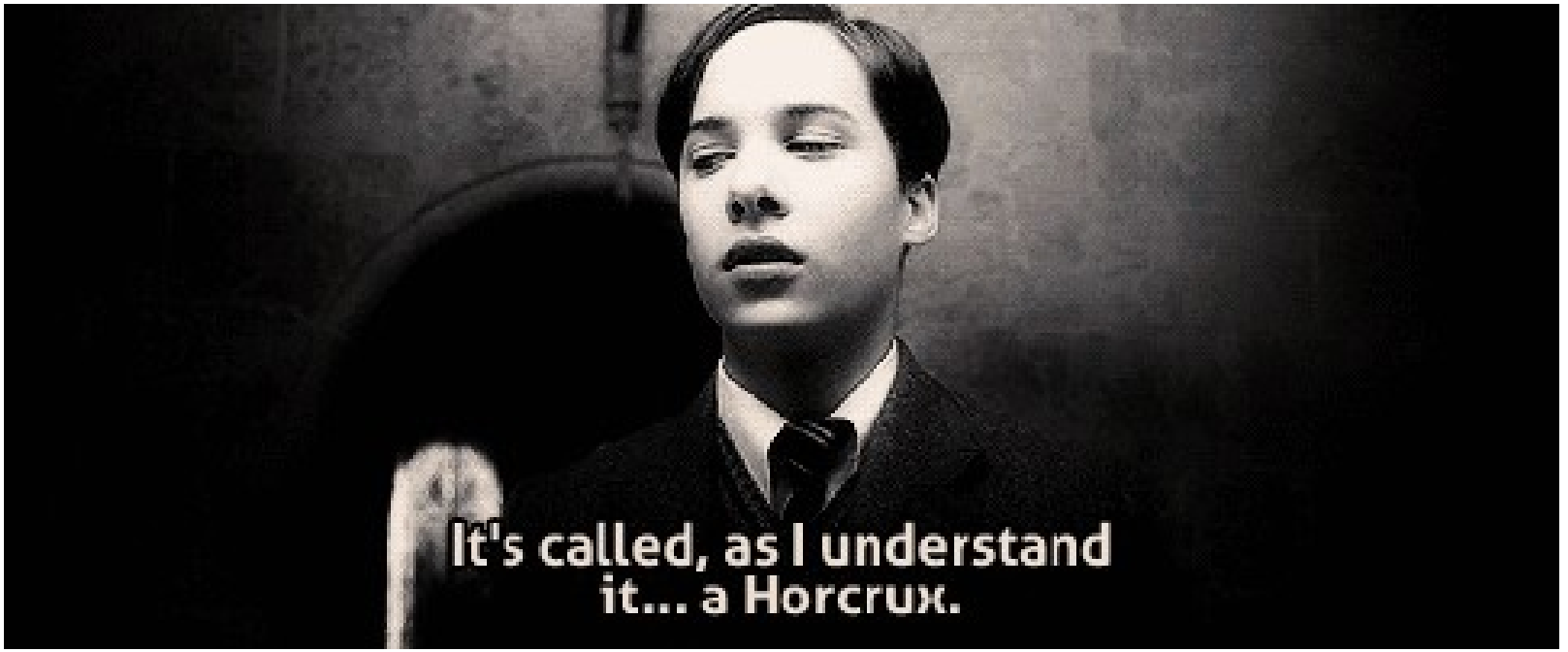
- Pre-final dense layer 4096x4096 matrix
- Filters 512x256x3x3
- Can we reduce the size/complexity?

Decomposition trick

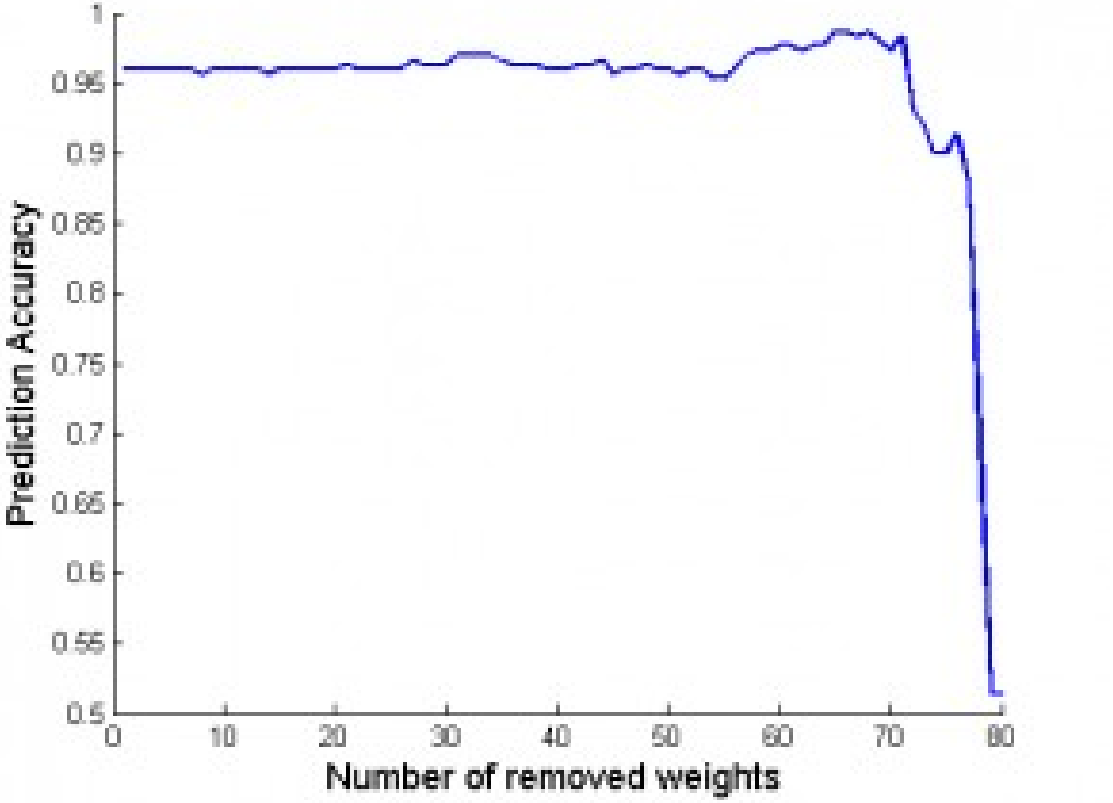


- Pre-final dense layer 4096x4096 matrix
- Filters 512x256x3x3
- Can we reduce the size/complexity?
  - Yes, we can! (c)
  - SVD (flatten to matrix), TensorTrain, ...

Reducing network size



Optimal Brain Damage



Optimal Brain Damage

- Define “neuron saliency” ~ how bad it is to remove it

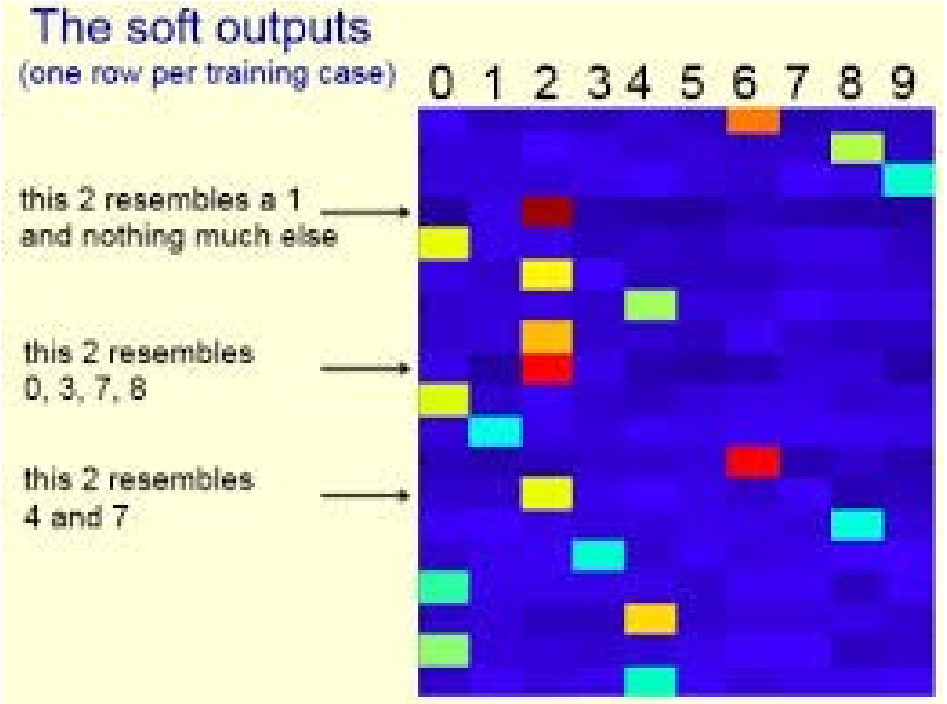
$$s_k = \frac{H_{kk} \cdot a_k^2}{2} = \frac{\delta^2 L}{\delta a_k} \cdot a_k^2 / 2$$

- Remove neurons with lowest saliency
- ???
- PROFIT!!!

- Similar idea: Regularize activations with L1 and prune zeros

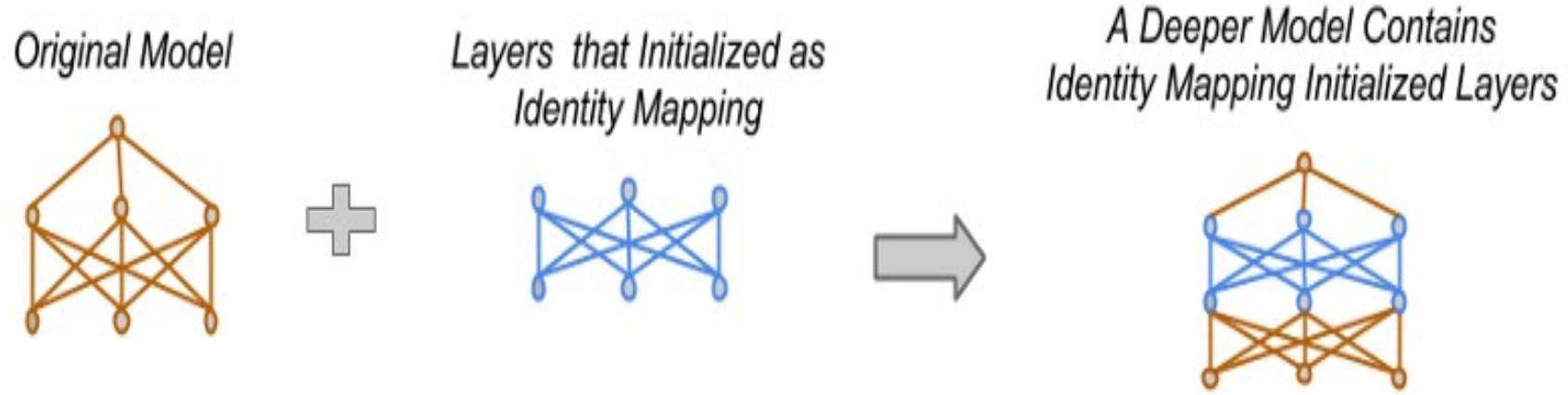
Soft Targets

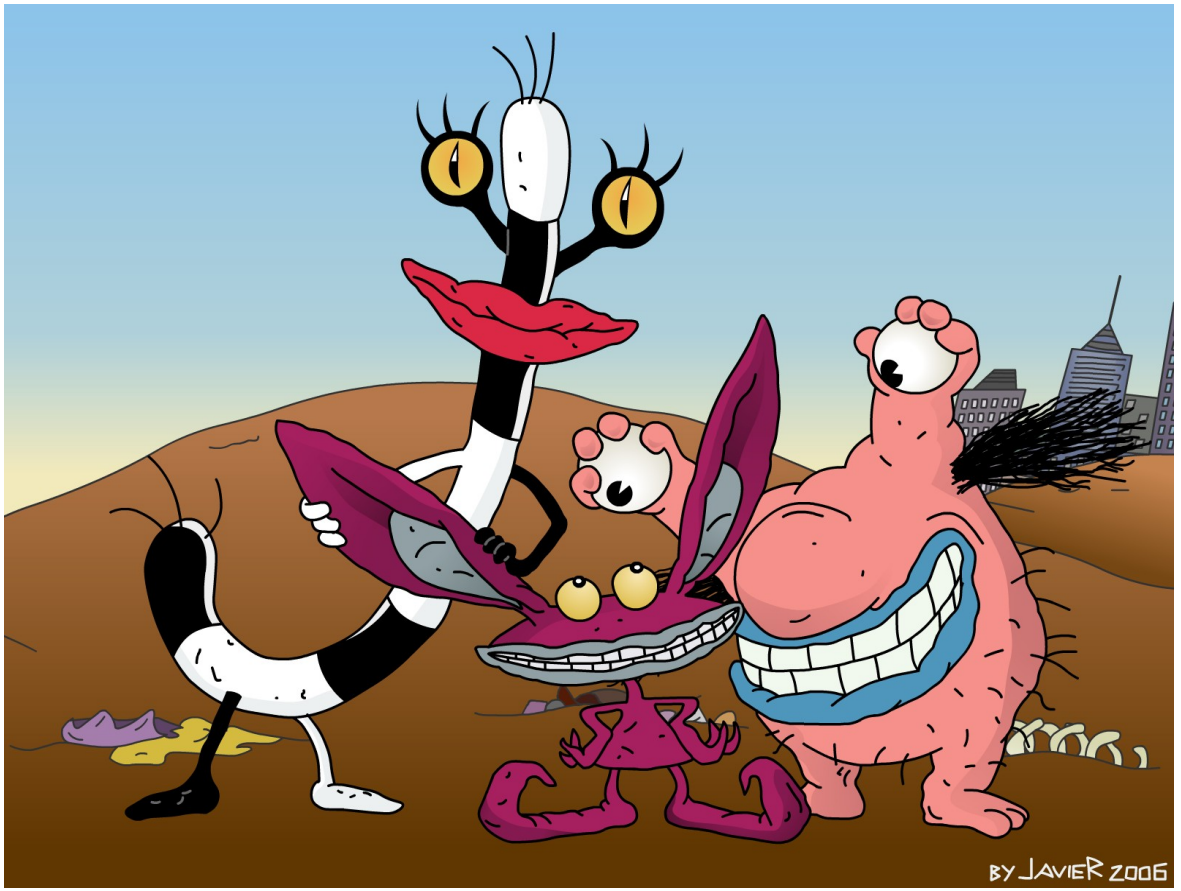
- Take the larger network (or any model / ensemble)
- Predict probabilities for training samples
- Use some function of probabilities as objective to train a new smaller network
- Some intuition: if an image is truly ambiguous, this way you will be able to handle that



Net2net

- Same, but expanding
- Start small, extend, fine-tune





Come at me bro!