Lab Meeting Presentation-3

June 14, 2017

Prerit Gupta
Summer Intern

Ravi Kiran Sarvadevabhatla
PhD Student

Objective

- Using Yosinki's Deep Visualization Toolbox to analyse the different layers of Alexnet & VGG-19 for top 160 classes of TU Berlin Sketch Dataset.
- Observing stroke density & class complexity through histogram plots.

Visualizing layers by finding the Maximum Activated Input Patch & perform Deconvolution

- Finding the top 9 patches from the input images in validation set which maximally activate a filter for a certain layer.
- To understand which kind of features are learned by filters in different layers and observe their sensitivity towards certain classes & localized patches with certain geometry.
- Perform deconvolution for the filters that are activated by maximally activating input patches.

Characterizing Visualizations in layers of Alexnet network

(Training of Alexnet uses mean file of sketch dataset)

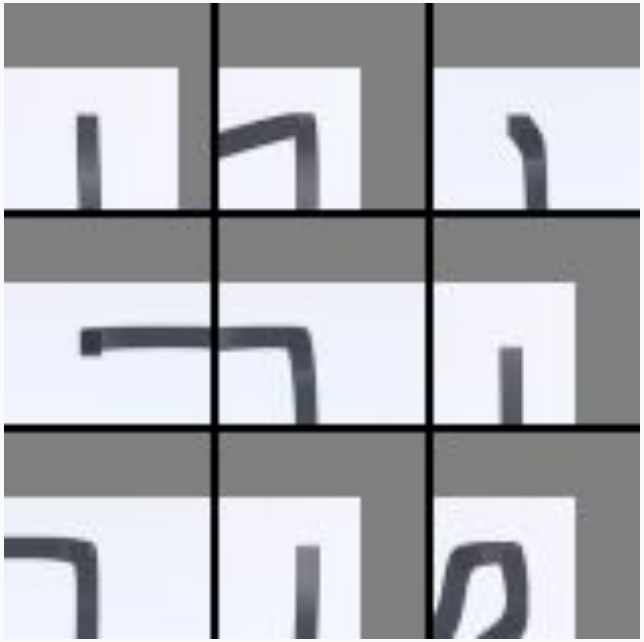
Convolution Layer 1

Input Patch for Maximum Activation

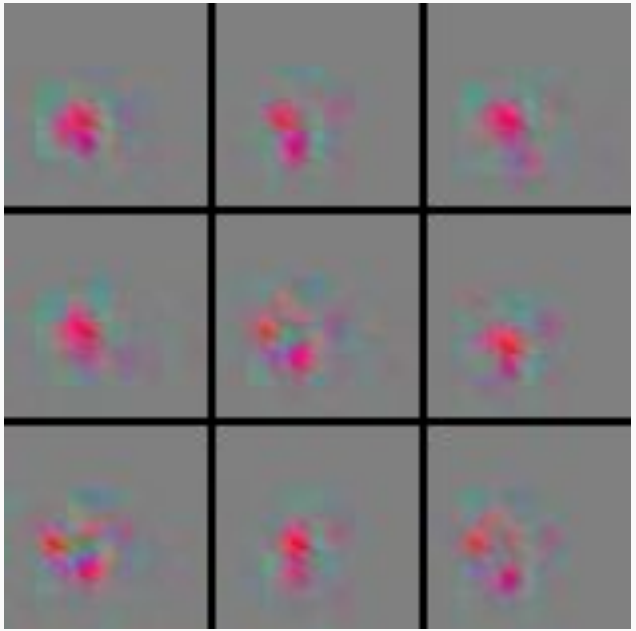
Deconvolution of the activated patch

- 11 x 11 Input Patch for 1st filter in 1st convolutional layer
- Filter sensitive to edge transition.

Convolution Layer 2



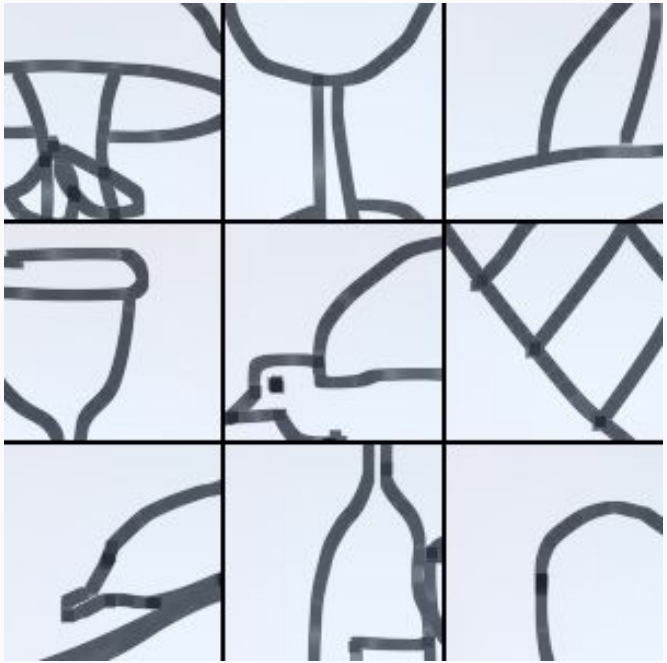
Input Patch for Maximum Activation



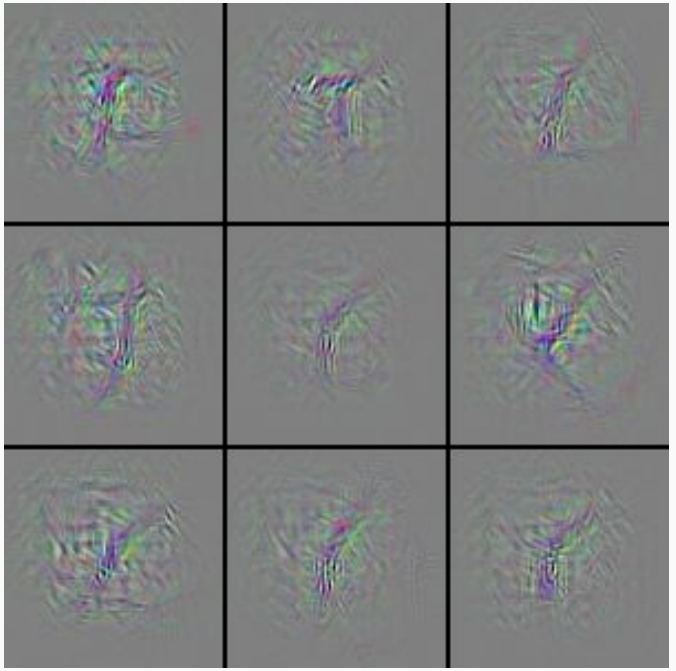
Deconvolution of the activated patch

- 51 x 51 Input Patch for 1st filter in 2nd convolutional layer
- Filter sensitive to straight lines.

Convolution Layer 3



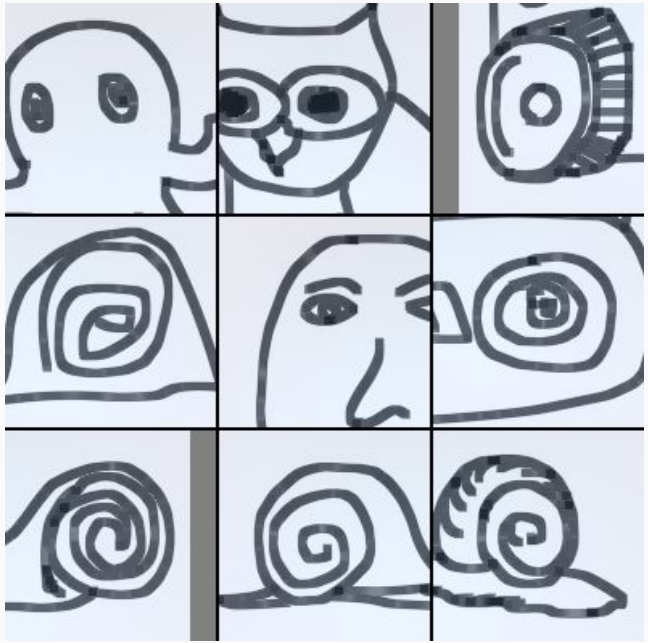
Input Patch for Maximum Activation



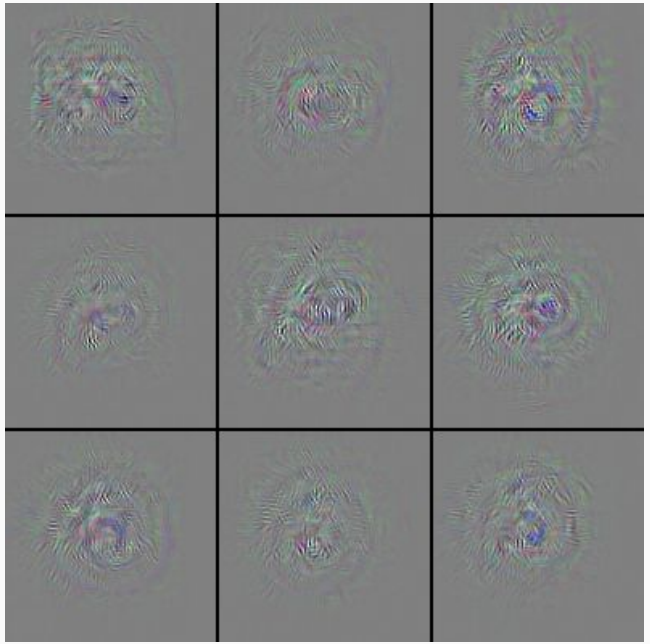
Deconvolution of the activated patch

- 99 x 99 Input Patch for 1st filter in 3rd convolutional layer
- Filter sensitive to specific parts of sketches like curved parts

Convolution Layer 4



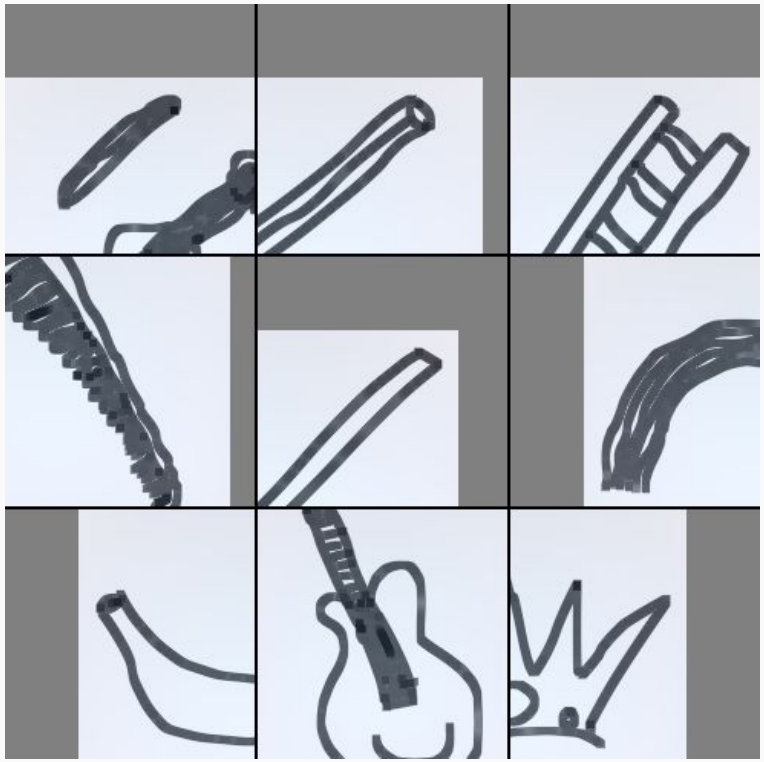
Input Patch for Maximum Activation



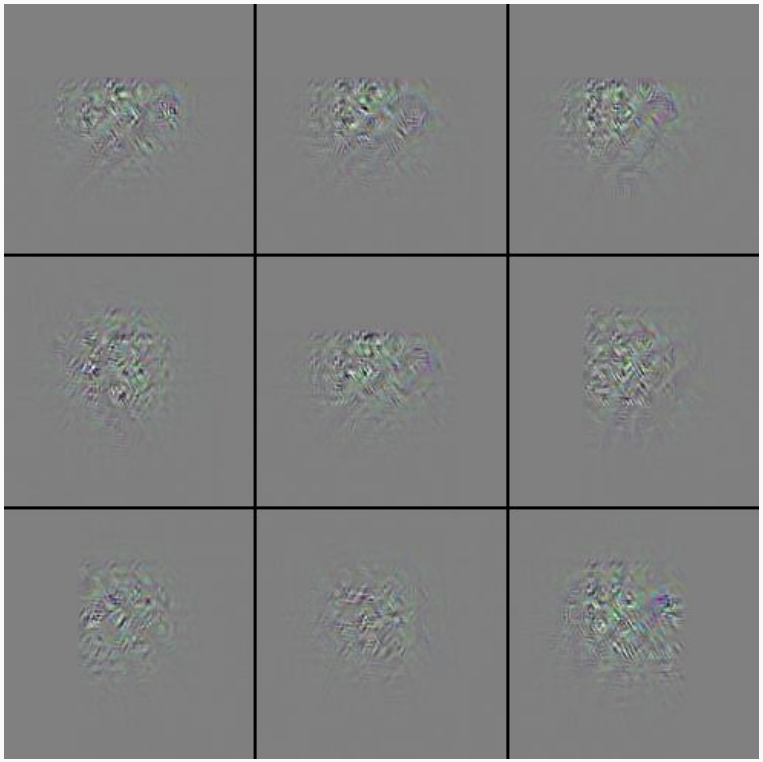
Deconvolution of the activated patch

- 131 x 131 Input Patch for 1st filter in 4th convolutional layer
- Filter sensitive to specific classes like snails & more towards round & spiral objects.

Convolution Layer 5



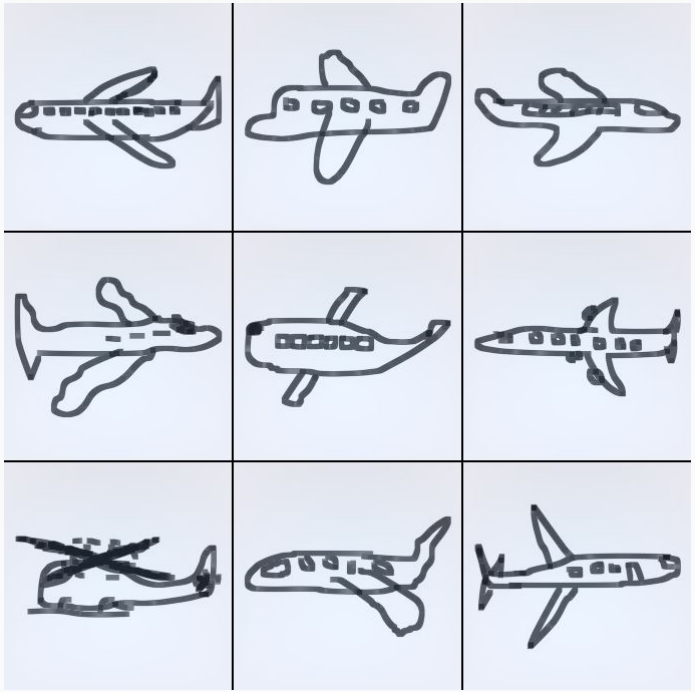
Input Match for Maximum Activation



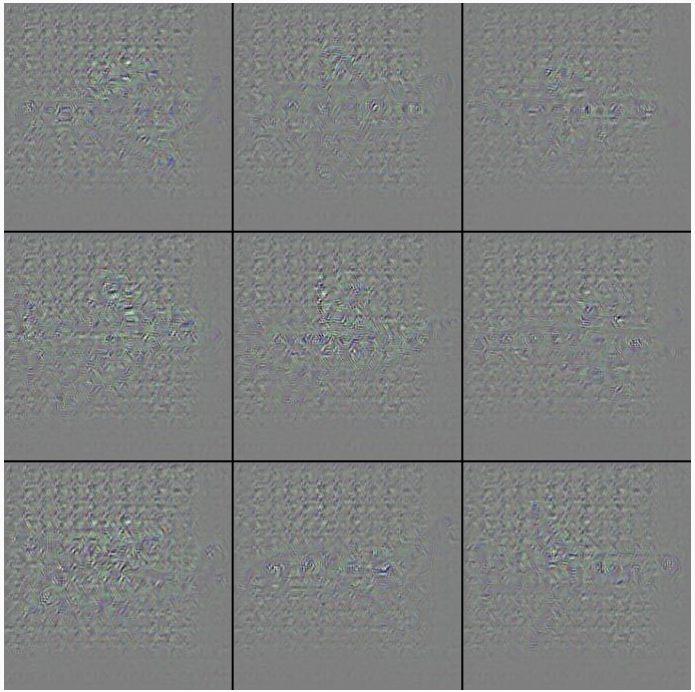
Deconvolution of the activated patch

- 163 x 163 Input Patch for 1st filter in 5th convolutional layer
- Filter sensitive to specific classes that have straight lines.

Fully Connected Layer 8



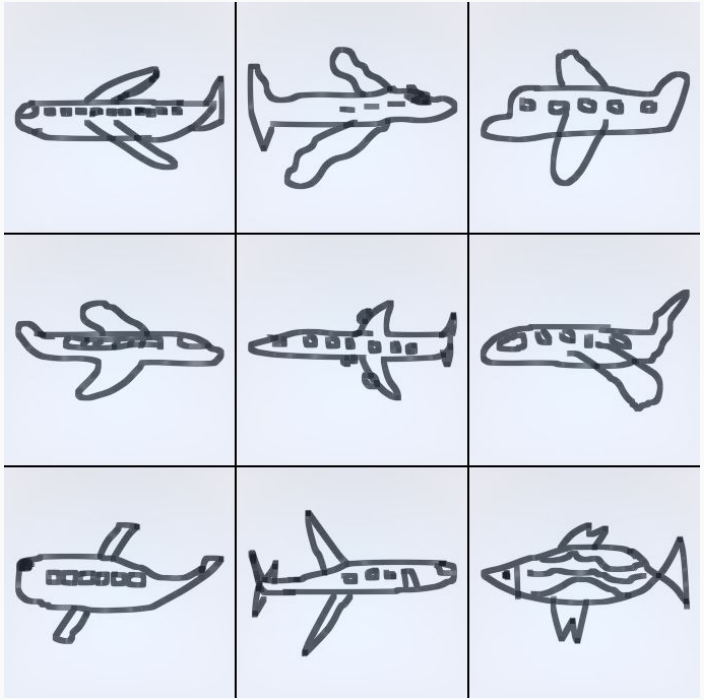
Input Patch for Maximum Activation



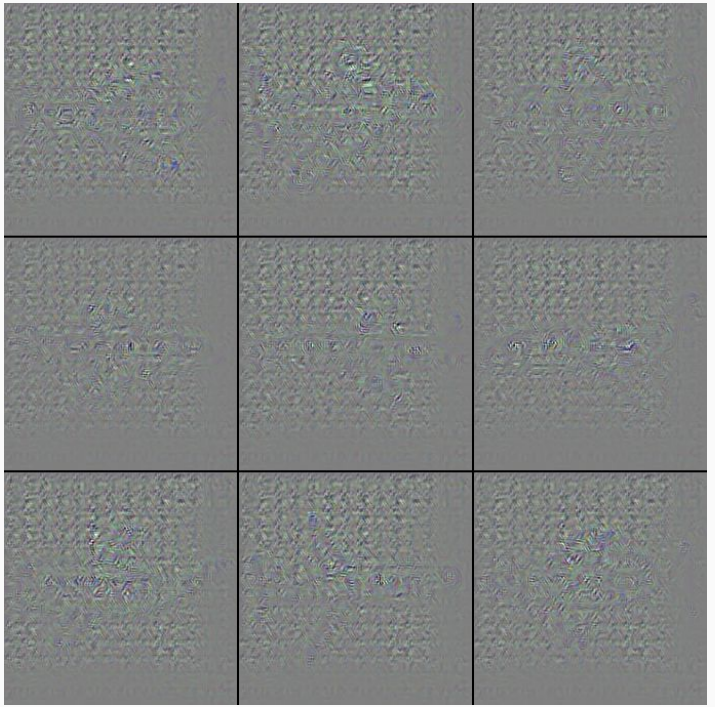
Deconvolution of the activated patch

- 227 x 227 Input Patch (complete image) for 1st filter in 8th layer which is fully connected.
- Filter sensitive to the airplane class. (which is also 1st class in target attribute)

Softmax Probability Layer



Input Patch for Maximum Activation



Deconvolution of the activated patch

- 227 x 227 Input Patch for 1st filter in the final softmax layer.
- The final layer is sensitive to only airplane class.

Plotting histograms to understand class preference in each layer

- The top 9 maximally activated images for every filter in a layer are traced back to the classes which they belong.
- For every rank of activation across all filters in a layer, a histogram is plotted to represent the count of class a filter prefers.
- Hence, for every layer 9 histograms are obtained on whose weighted summation a single histogram is obtained for every layer.
- Weight of rank 1 class -> 9 , Rank 2 -> 8 & so on…. Rank 9 -> 1

Inference

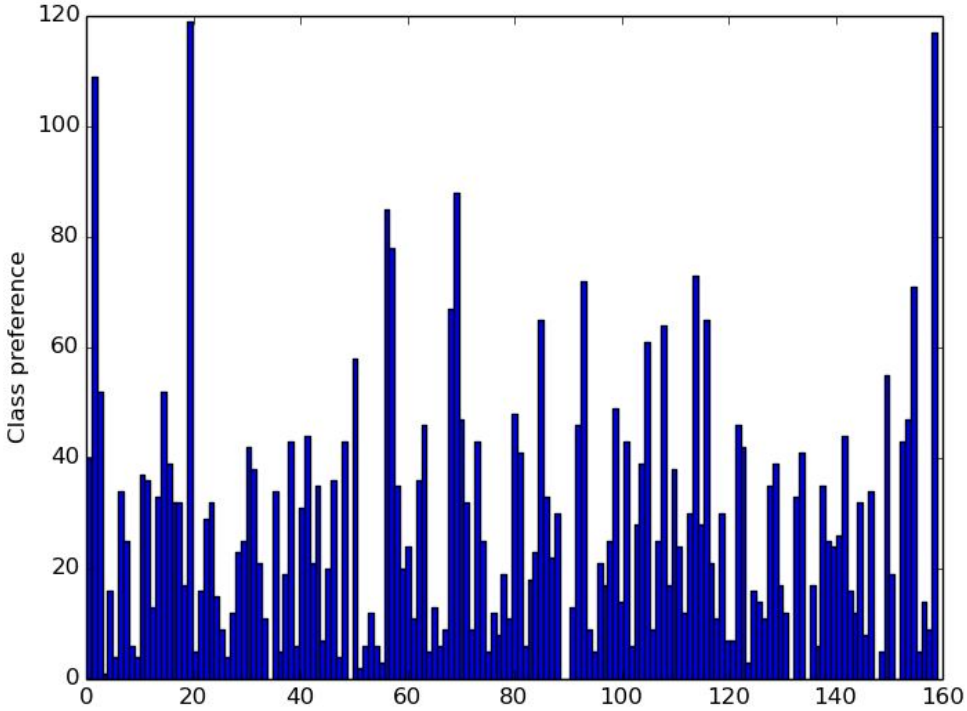
Top 5 prefered classes :

- Butterfly
- Zebra
- Alarm clock
- Hedgehog
- Windmill

Least 5 prefered classes:

- Ant
- Bus
- Cloud
- Flying bird
- Foot

Convolution Layer 1



Inference

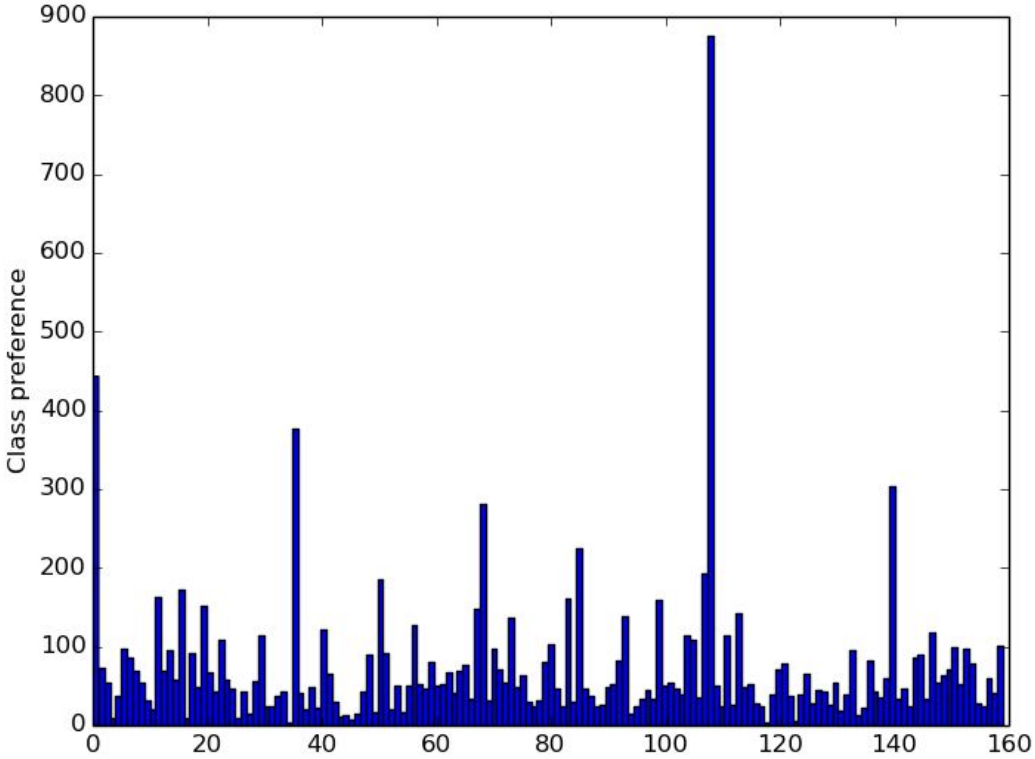
Top 5 prefered classes :

- Rainbow
- Comb
- Zebra
- Mermaid
- Radio

Least 5 prefered classes:

- Candle
- Bowl
- Flying bird
- Bed
- Ant

Convolution Layer 2



Inference

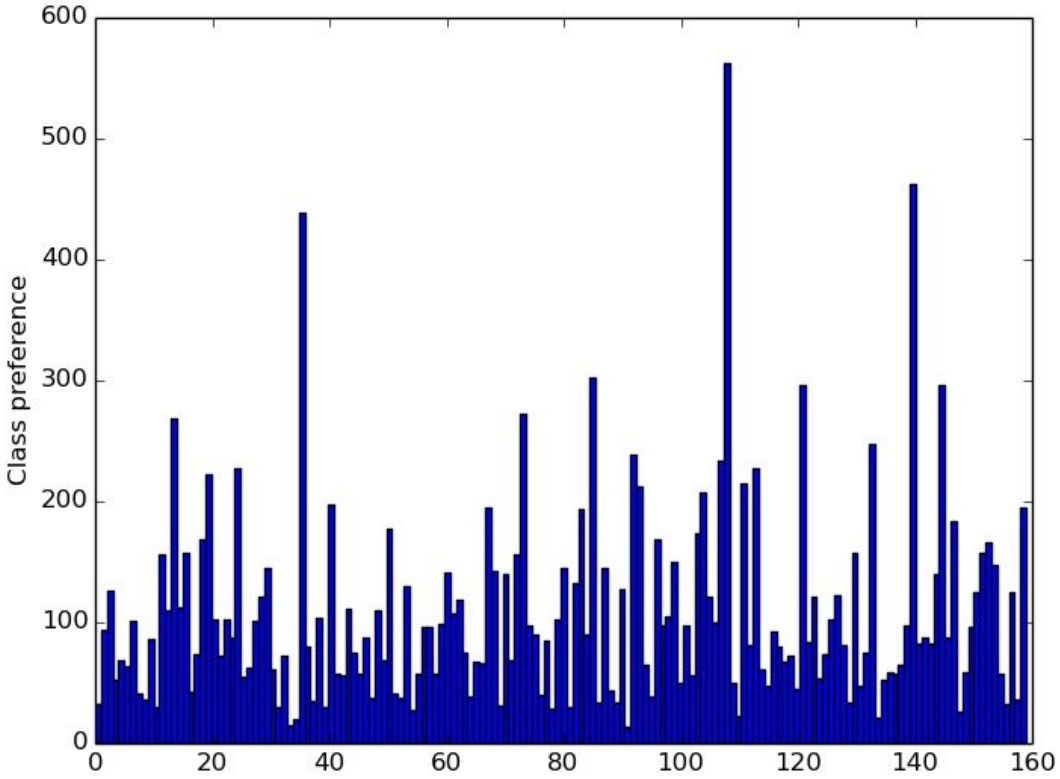
Top 5 prefered classes :

- Octopus
- Snail
- Sun
- Pizza
- Tractor

Least 5 prefered classes:

- Pen
- Cigarette
- Nose
- Rifle
- Bed

Convolution Layer 3



Inference

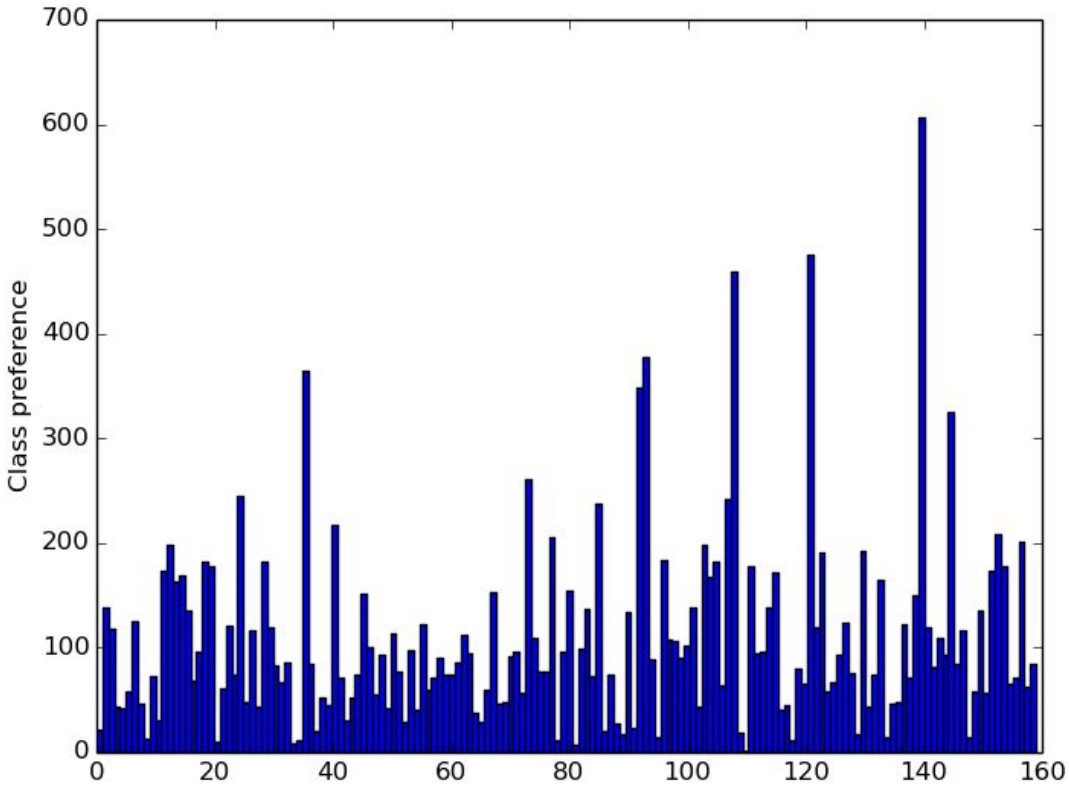
Top 5 prefered classes :

- Snail
- Octopus
- Key
- Tractor
- Telephone

Least 5 prefered classes:

- Door
- Rifle
- Pen
- Revolver
- Fish

Convolution Layer 4



Inference

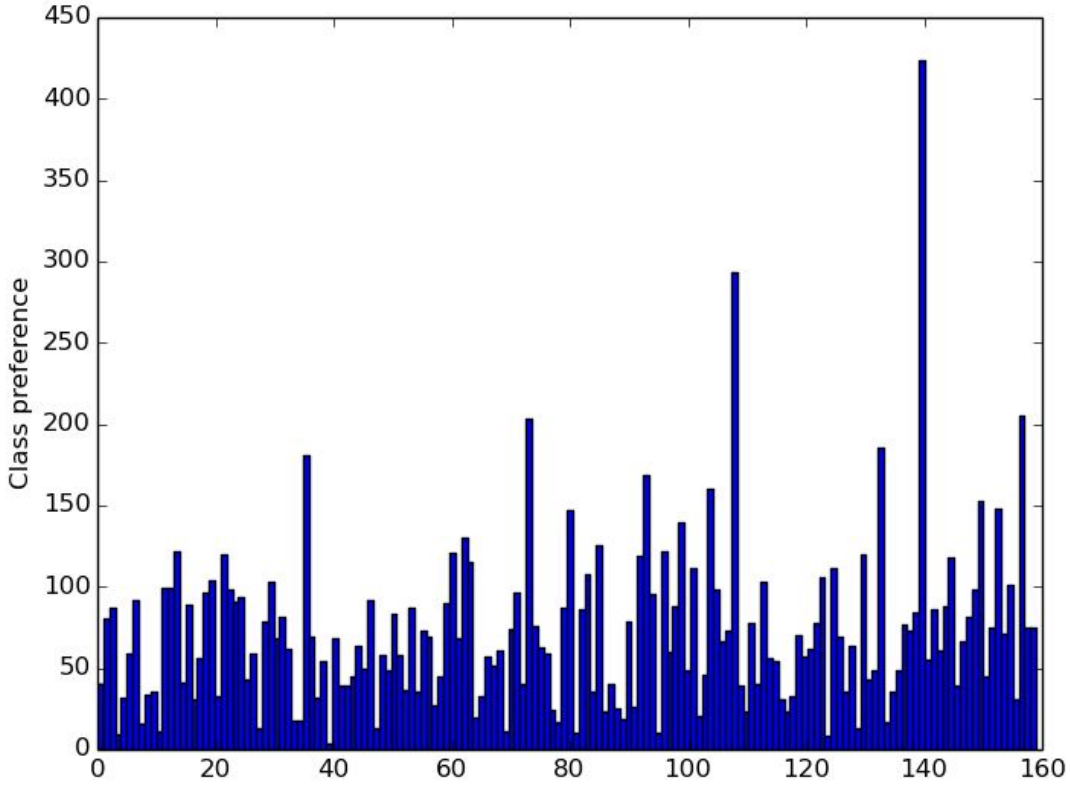
Top 5 prefered classes :

- Tennis-racket
- Sun
- Telephone
- Cake
- Rainbow

Least 5 prefered classes:

- Baseball bat
- Nose
- Rifle
- Bed
- Bowl

Convolution Layer 5



Inference

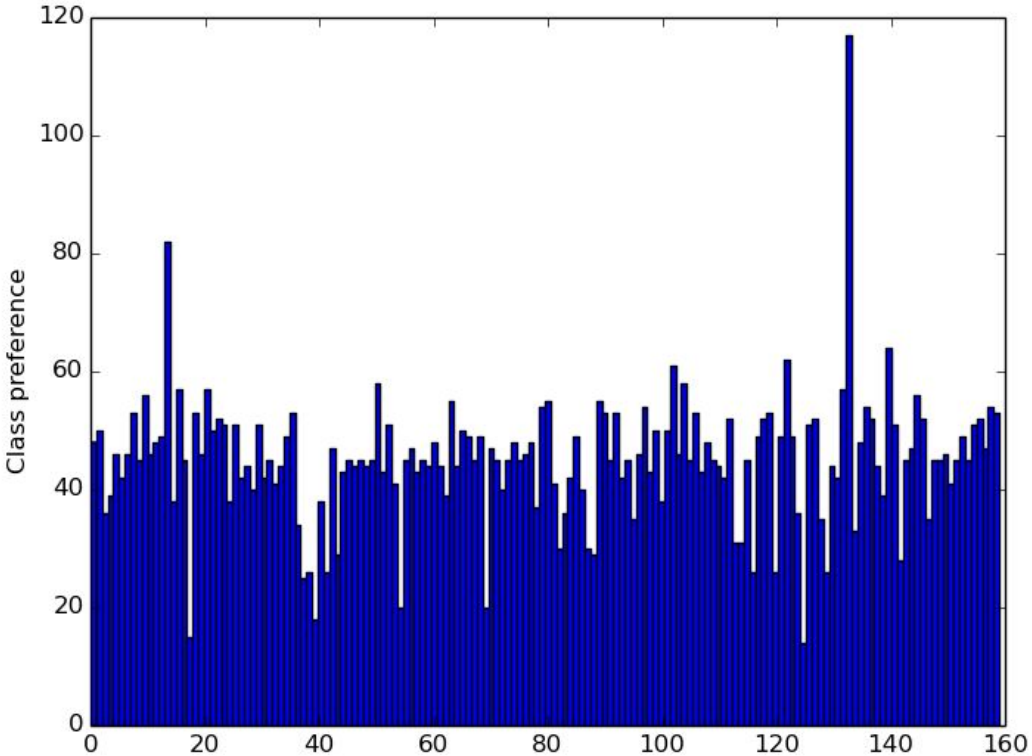
Top 5 prefered classes :

- Sun
- Book
- Kangaroo
- Bicycle
- Octopus

Least 5 prefered classes:

- Dragon
- Bread
- Scorpion
- Cat
- Elephant

Fully Connected Layer 8



Inference

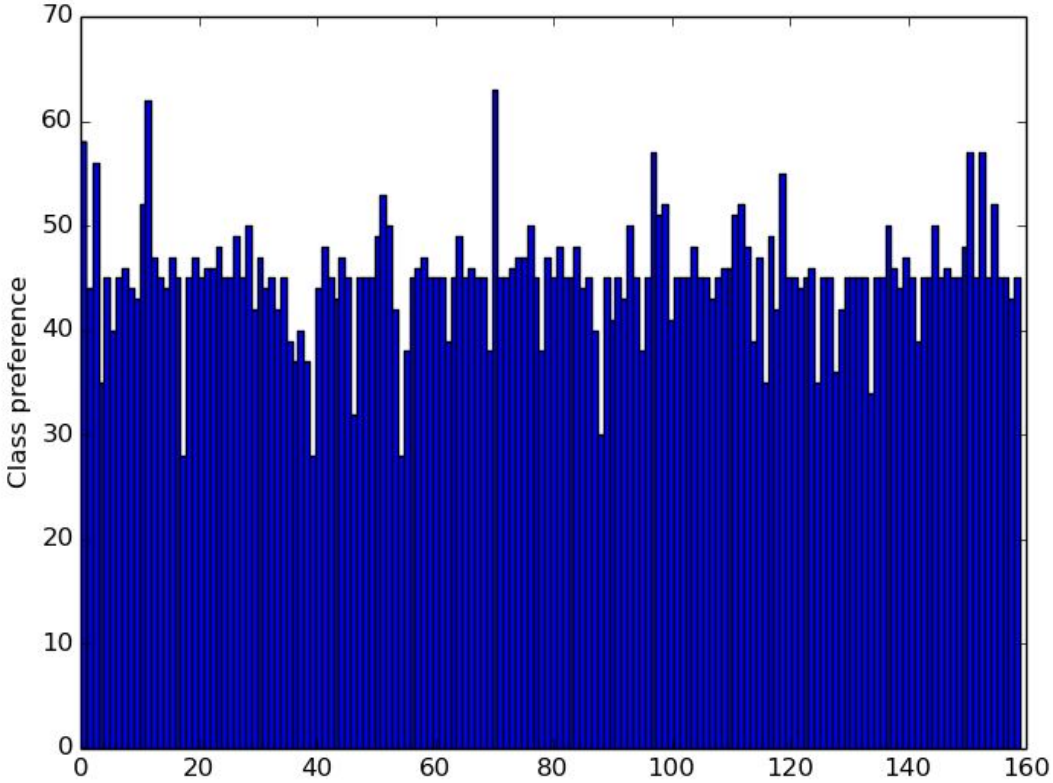
Top 5 prefered classes :

- Pineapple
- Tablelamp
- Crab
- Shoe
- Tv

Least 5 prefered classes:

- Horse
- Dragon
- Strawberry
- Elephant
- Computer-mouse

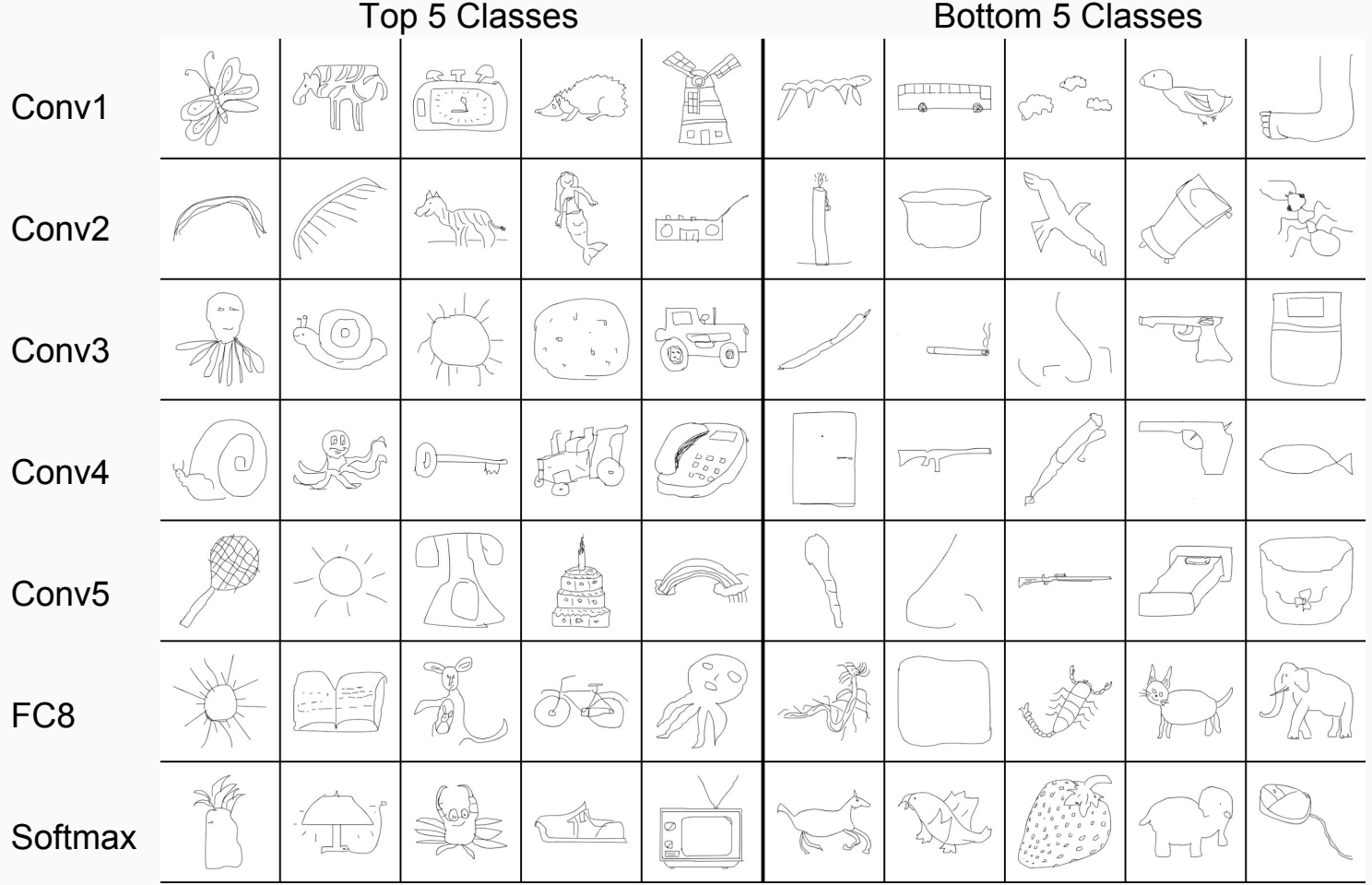
Softmax Probability Layer



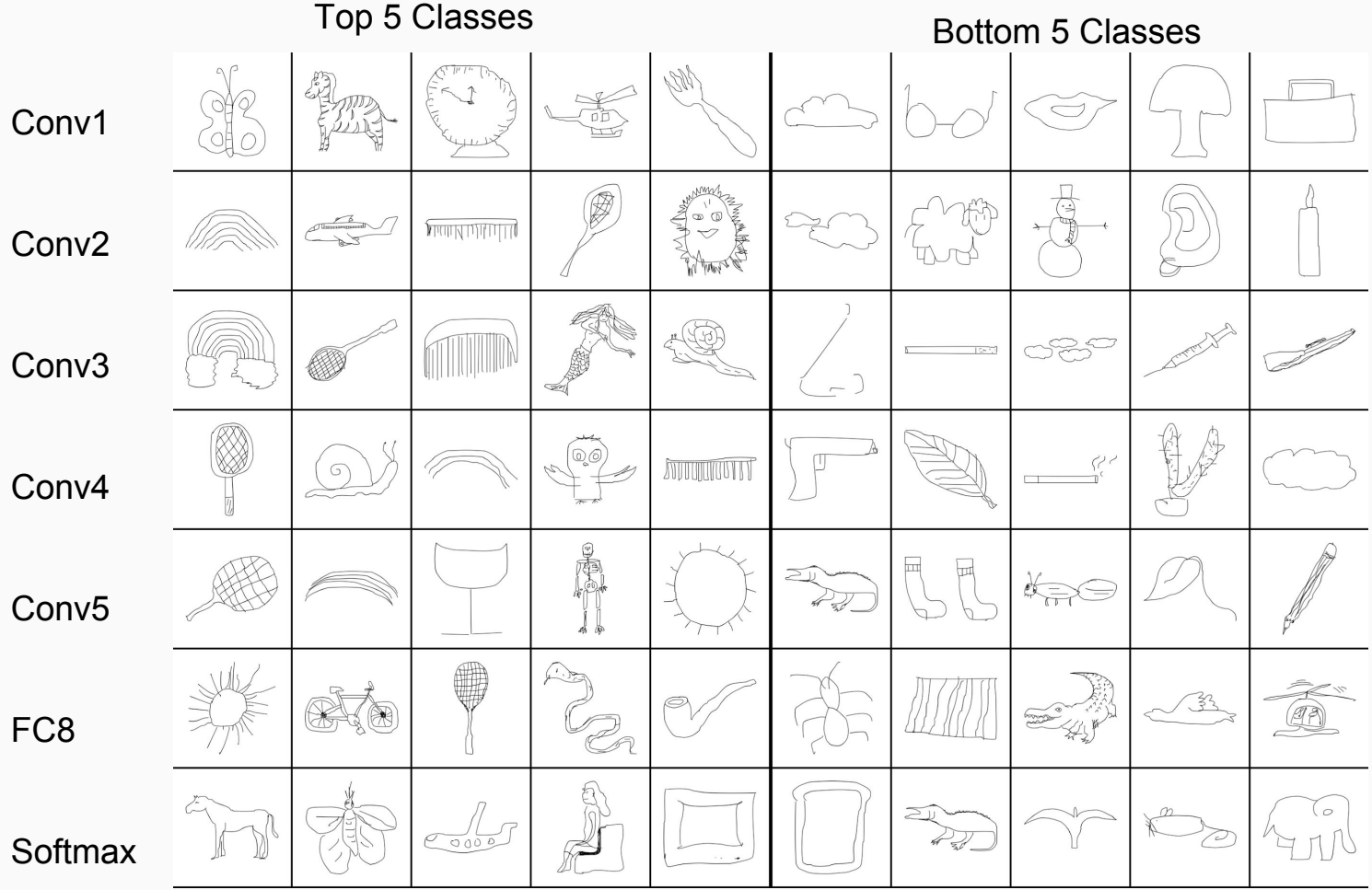
Layer Biaseness towards specific classes

Layers	Top 5 Classes	Bottom 5 Classes
conv1:	butterfly zebra alarm clock hedgehog windmill	ant bus cloud flying bird foot
conv2:	rainbow comb zebra mermaid radio	candle bowl flying bird bed ant
conv3:	octopus snail sun pizza tractor	pen cigarette nose rifle bed
conv4:	snail octopus key tractor telephone	door rifle pen revolver fish
conv5:	tennis-racket sun telephone cake rainbow	baseball bat nose rifle bed bowl
Fc8 :	sun book kangaroo bicycle octopus	dragon bread scorpion cat elephant
Prob :	pineapple tablelamp crab shoe tv	horse dragon strawberry elephant computer-mouse
Overall:	octopus sun rainbow snail telephone	bed rifle ant bowl dragon

Summary Plot (Alexnet with Imagenet Mean)

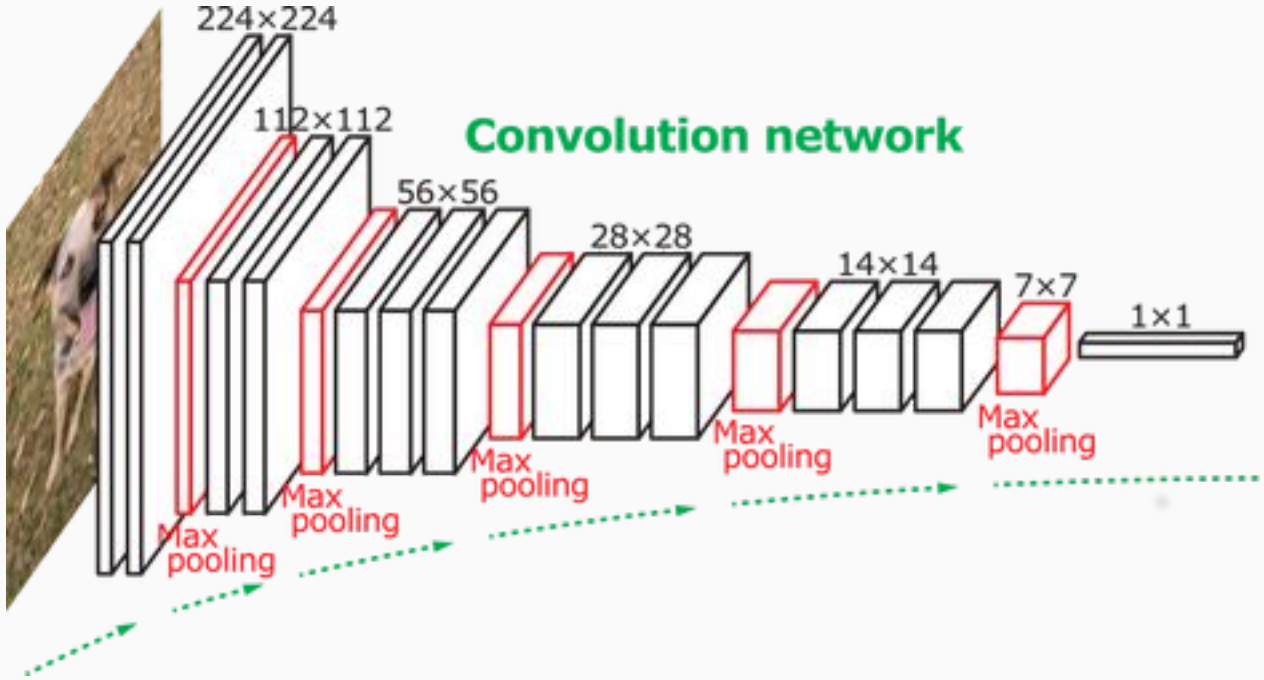


Summary Plot (Alexnet with Sketches Mean)



Characterizing Visualizations in layers of VGG-19 network

VGG-16 architecture



Adding extra single layers in stage 3, 4, 5 results in VGG-19 architecture.

Conv1_1 Layer

Input Patch for Maximum Activation

Deconvolution of the activated patch

- 3 x 3 Input Patch for 1st filter in Conv1_1 layer
- Filter sensitive to edge transition.

Conv1_2 Layer

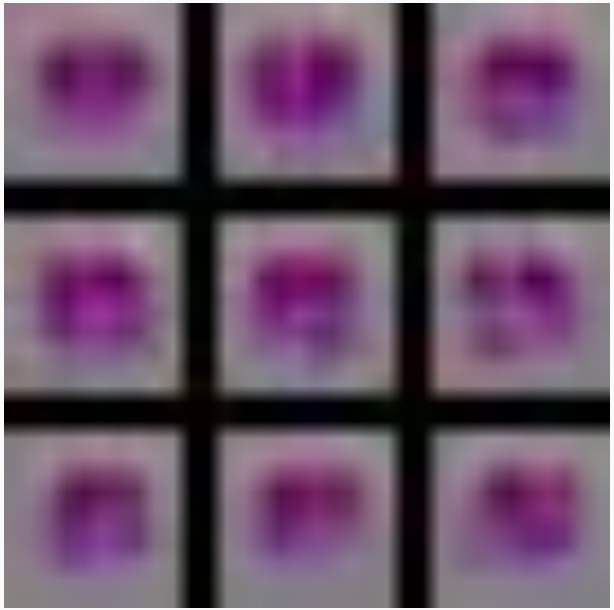
Input Patch for Maximum Activation

Deconvolution of the activated patch

- 5 x 5 Input Patch for 1st filter in Conv1_2 layer
- Filter sensitive to edge transition.

Conv2_1 Layer

Input Patch for Maximum Activation



Deconvolution of the activated patch

- 10 x 10 Input Patch for 1st filter in Conv2_1layer
- Filter sensitive to edge transition.

Conv2_2 Layer

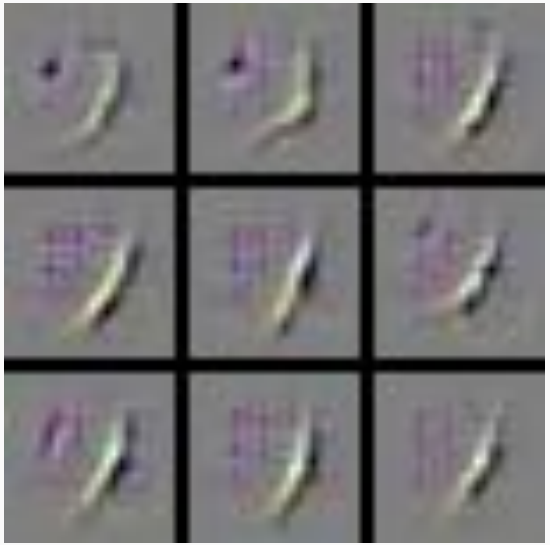
Input Patch for Maximum Activation

Deconvolution of the activated patch

- 14 x 14 Input Patch for 1st filter in Conv2_2 layer
- Filter sensitive to diagonal stroke transition.

Conv3_1 Layer

Input Patch for Maximum Activation

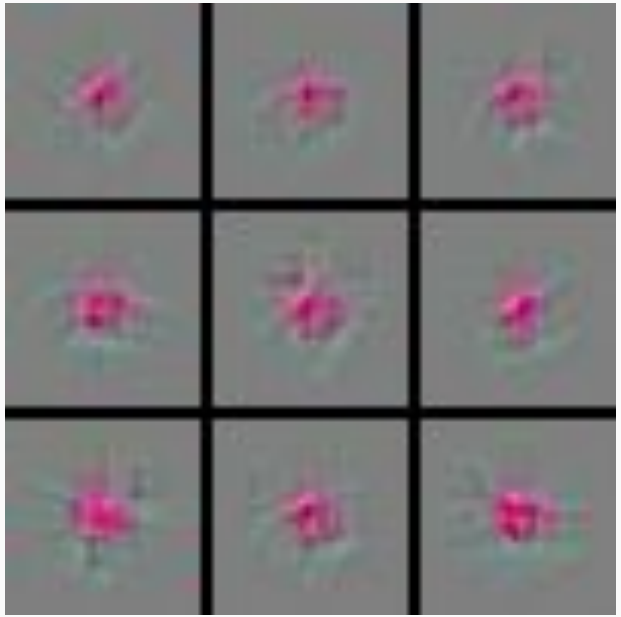


Deconvolution of the activated patch

- 24 x 24 Input Patch for 1st filter in Conv3_1 layer
- Filter sensitive to curved straight lines.

Conv3_2 Layer

Input Patch for Maximum Activation

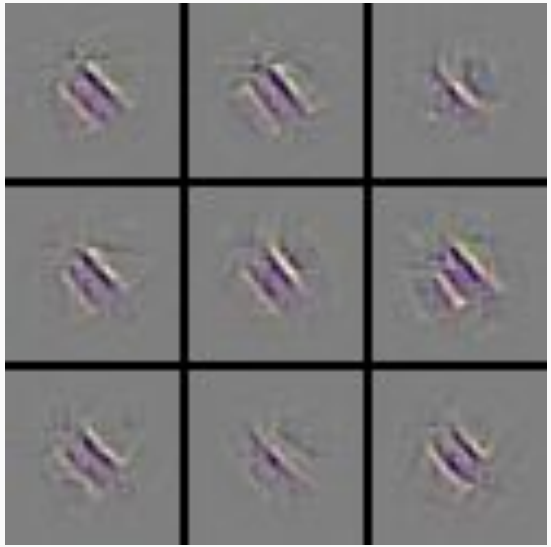


Deconvolution of the activated patch

- 32 x 32 Input Patch for 1st filter in Conv3_2 layer
- Filter sensitive to random strokes & lines.

Conv3_3 Layer

Input Patch for Maximum Activation

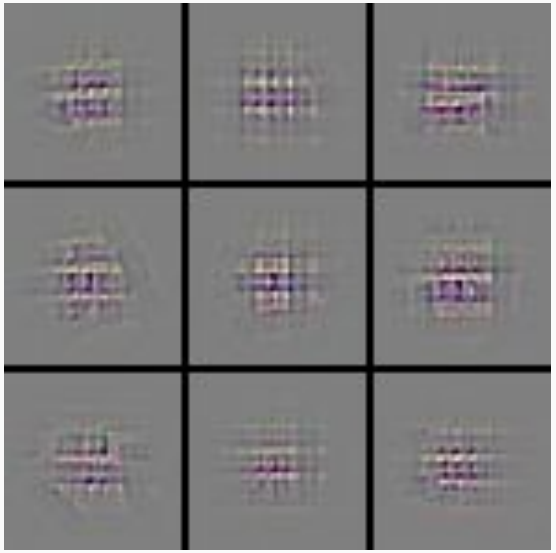


Deconvolution of the activated patch

- 40 x 40 Input Patch for 1st filter in Conv3_3 layer
- Filter sensitive to high stroke density patterns like nets, spokes in a wheel.

Conv3_4 Layer

Input Patch for Maximum Activation

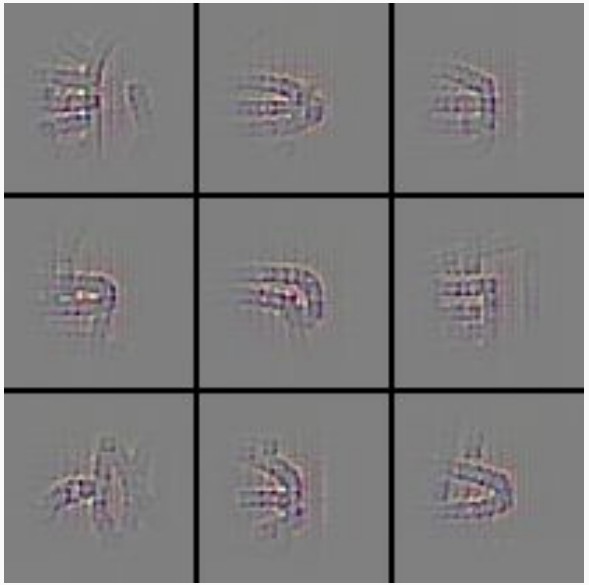


Deconvolution of the activated patch

- 48 x 48 Input Patch for 1st filter in Conv3_4 layer
- Filter sensitive to cross net patterns.

Conv4_1 Layer

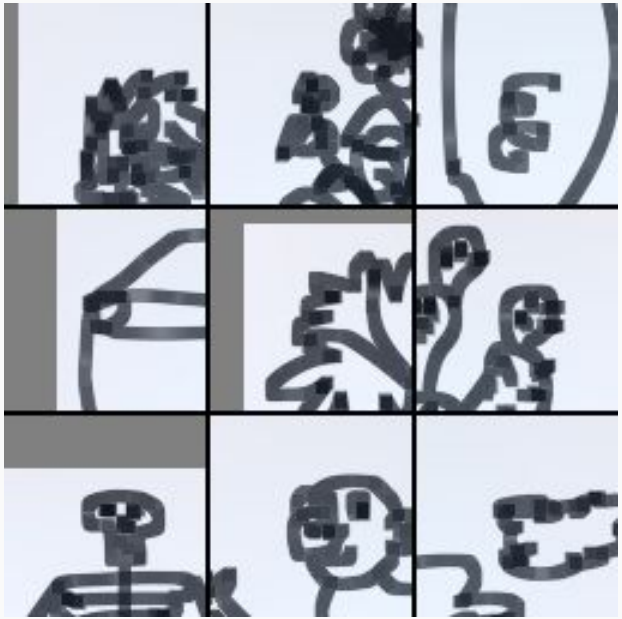
Input Patch for Maximum Activation



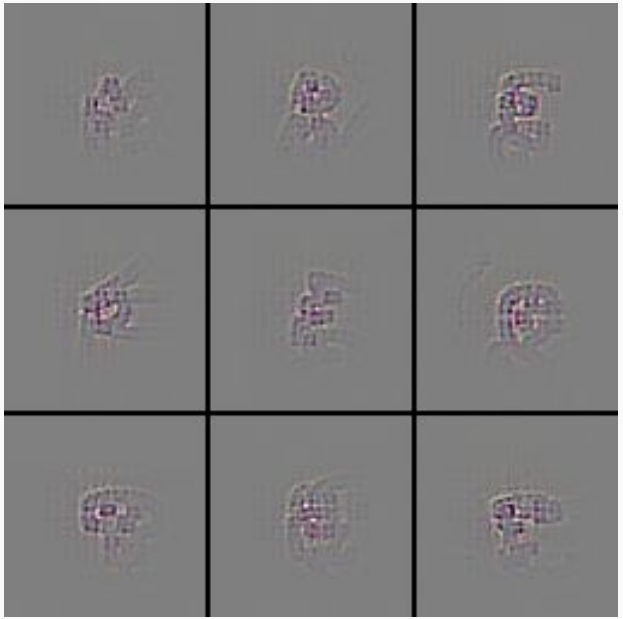
Deconvolution of the activated patch

- 68 x 68 Input Patch for 1st filter in Conv4_1 layer
- Filter sensitive to specific patterns.

Conv4_2 Layer



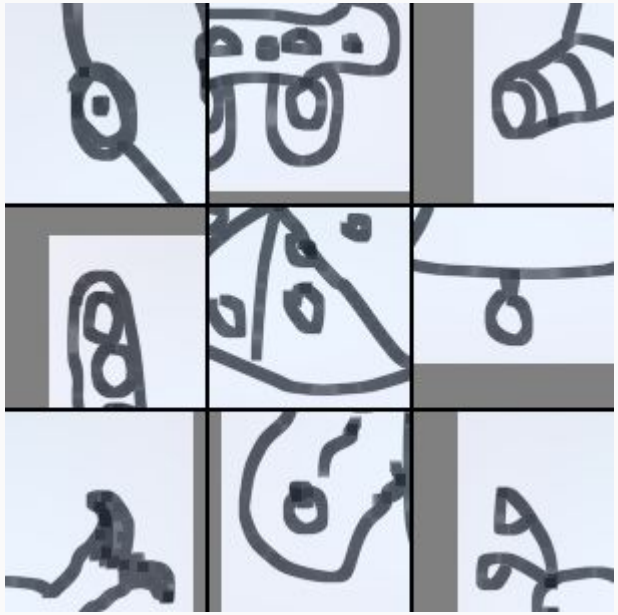
Input Patch for Maximum Activation



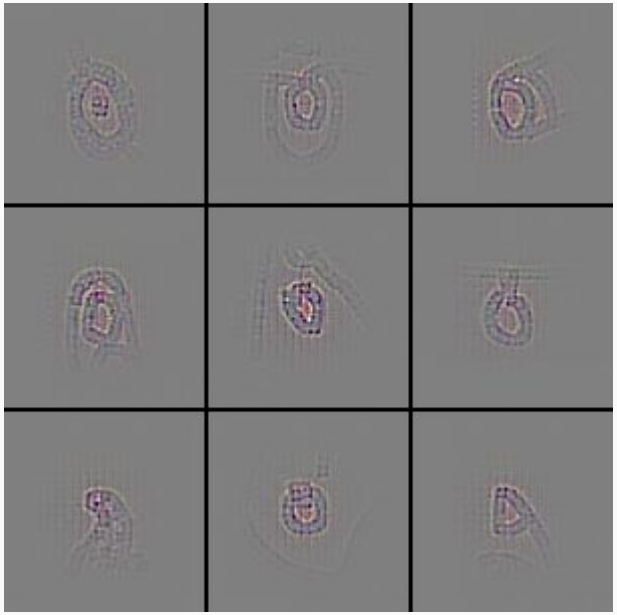
Deconvolution of the activated patch

- 84 x 84 Input Patch for 1st filter in Conv4_2 layer
- Filter sensitive to faces, leaves etc.

Conv4_3 Layer



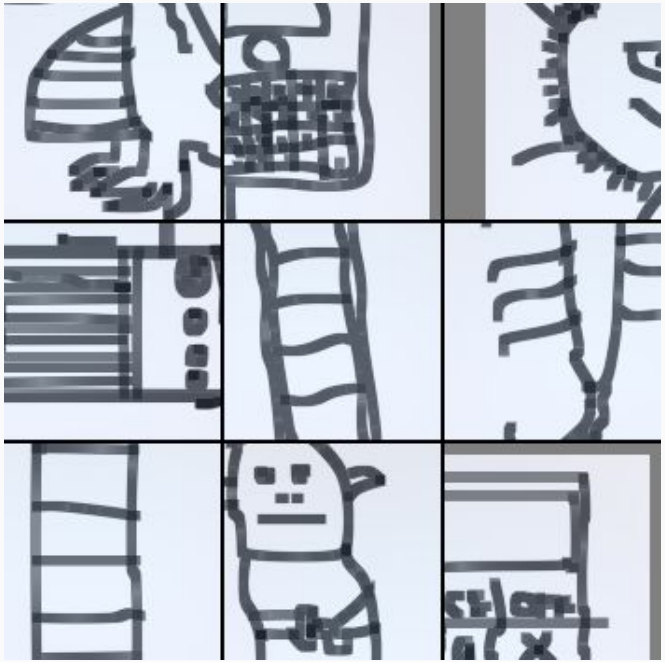
Input Patch for Maximum Activation



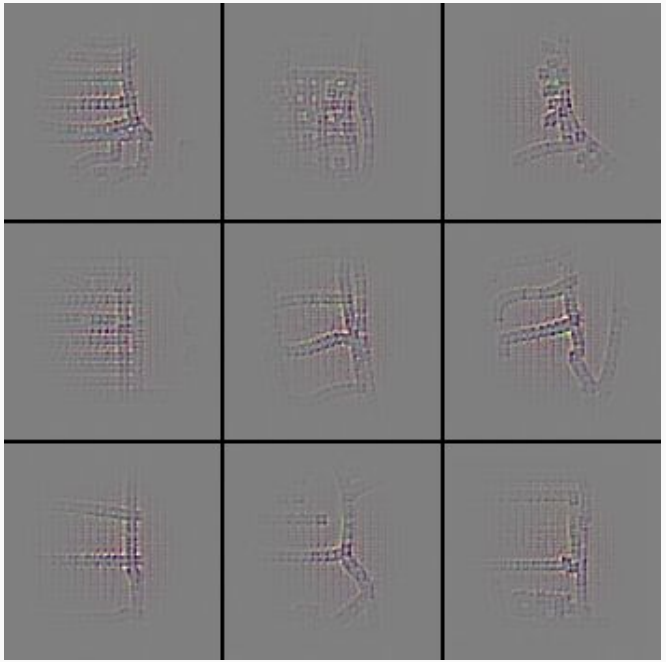
Deconvolution of the activated patch

- 100 x 100 Input Patch for 1st filter in Conv4_3 layer
- Filter sensitive to random patterns.

Conv4_4 Layer



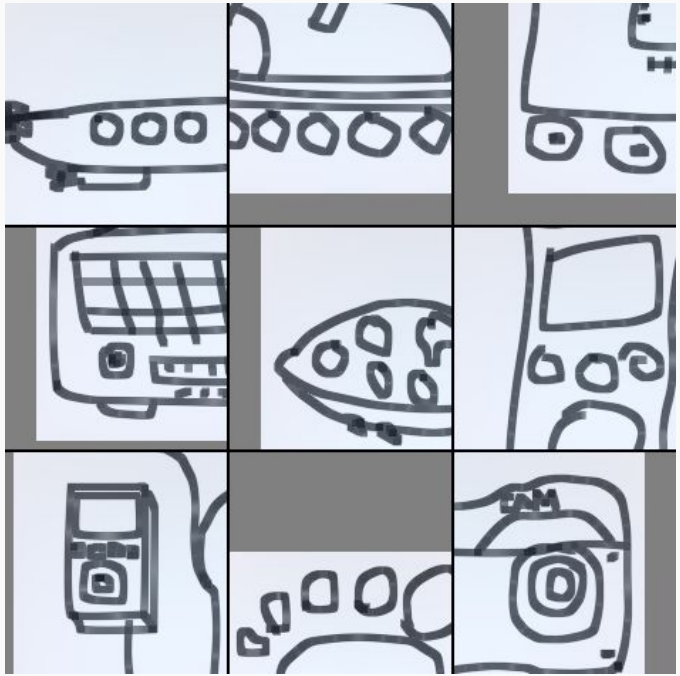
Input Patch for Maximum Activation



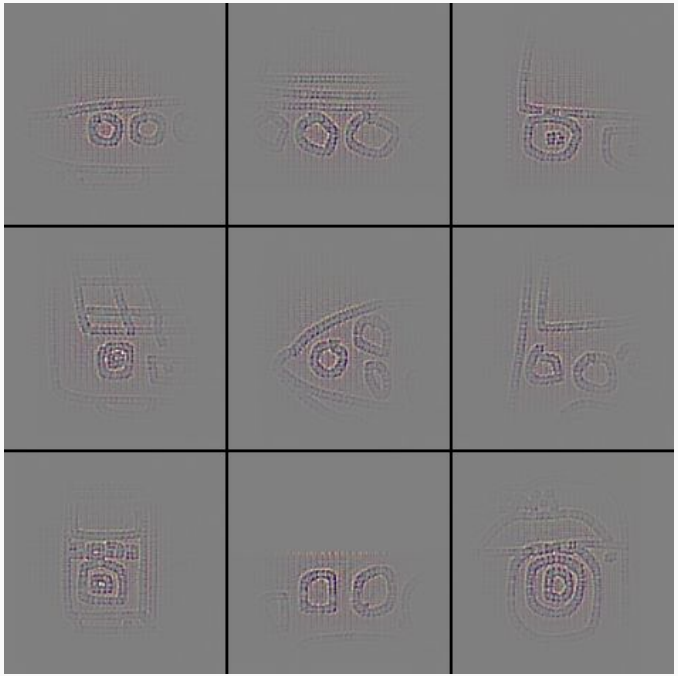
Deconvolution of the activated patch

- 116 x 116 Input Patch for 1st filter in Conv4_4 layer
- Filter sensitive to horizontal strokes pattern.

Conv5_1 Layer



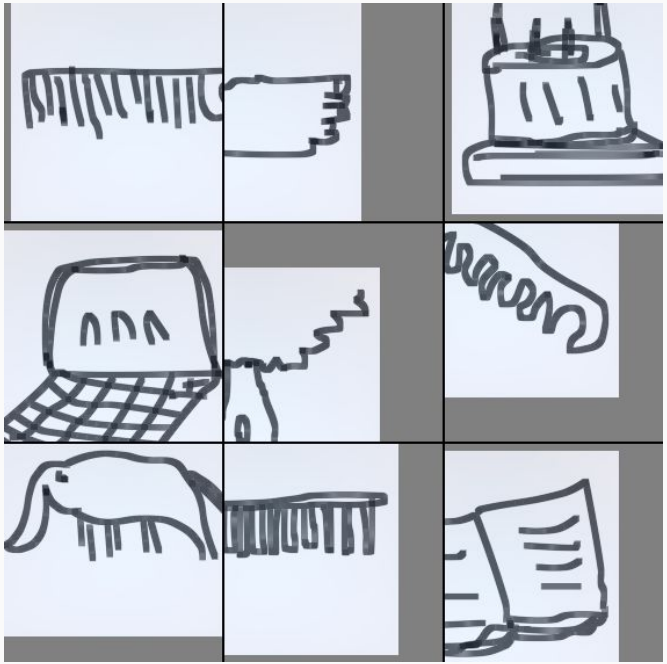
Input Patch for Maximum Activation



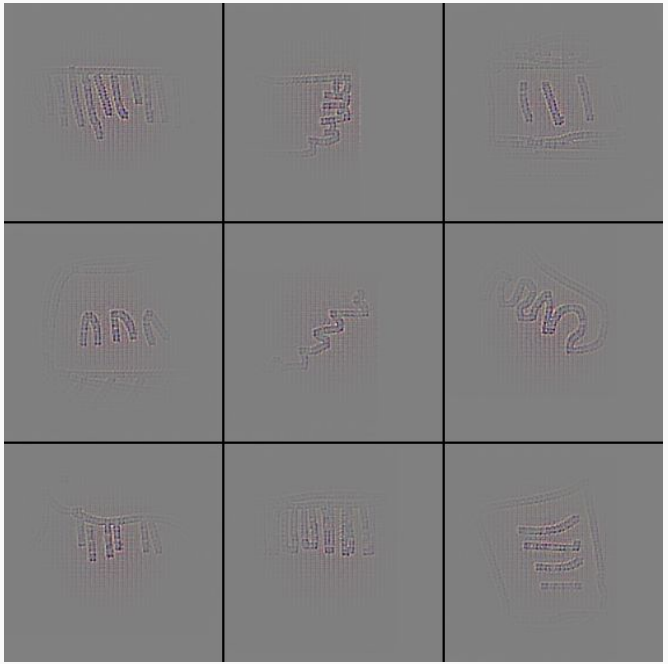
Deconvolution of the activated patch

- 156 x 156 Input Patch for 1st filter in Conv5_1 layer
- Filter sensitive to patterns with small many small circles.

Conv5_2 Layer



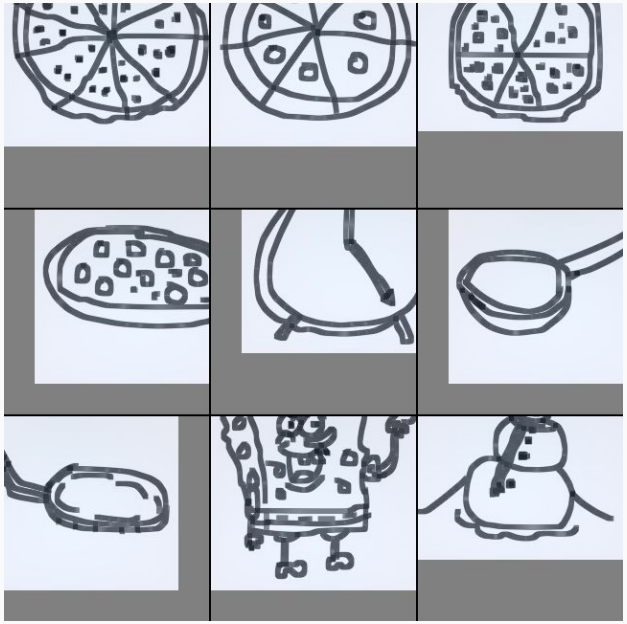
Input Patch for Maximum Activation



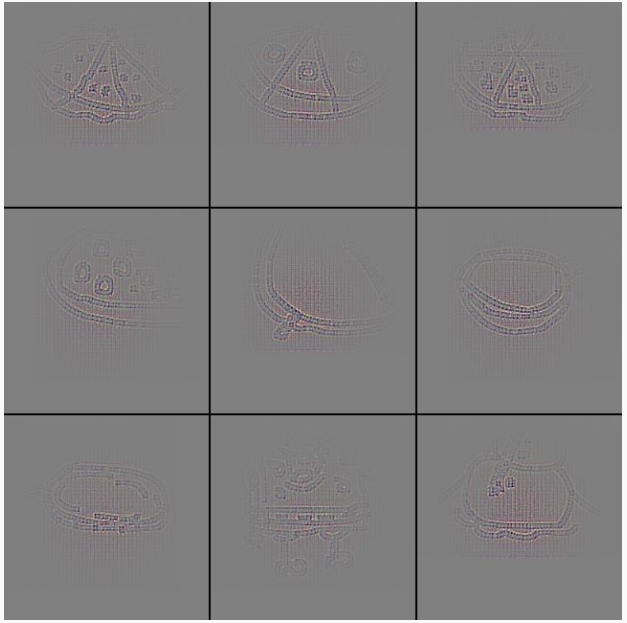
Deconvolution of the activated patch

- 188 x 188 Input Patch for 1st filter in conv5_2 layer
- Filter sensitive to patterns with vertical strokes.

Conv5_3 Layer



Input Patch for Maximum Activation

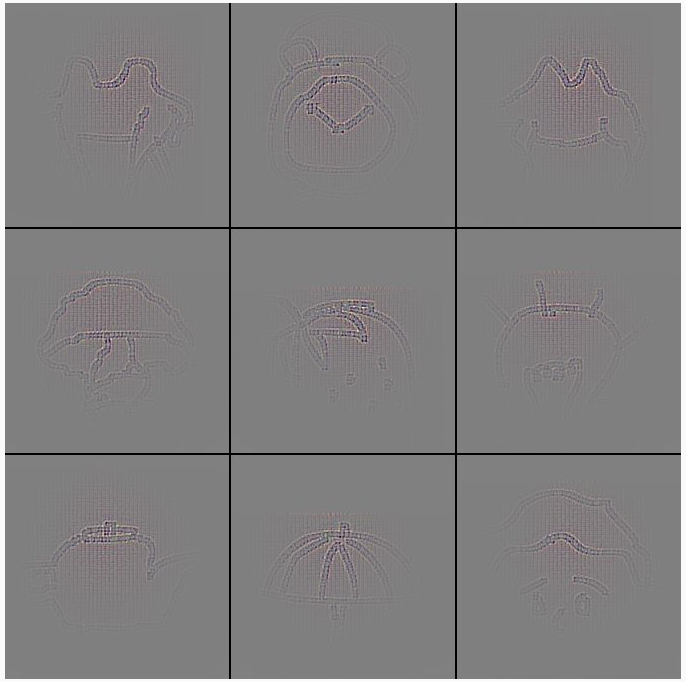


Deconvolution of the activated patch

- 214 x 214 Input Patch for 1st filter in Conv5_3 layer
- Filter sensitive to specific round classes like pizza, clock, snowman .

Conv5_4 Layer

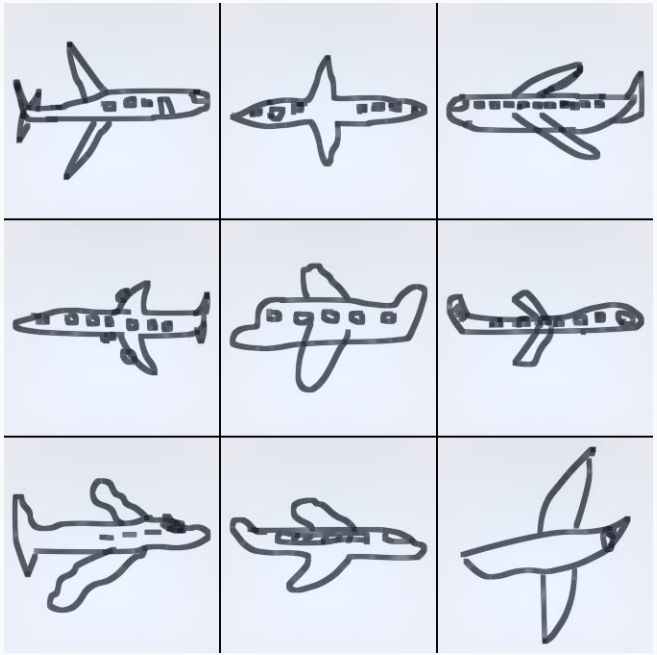
Input Patch for Maximum Activation



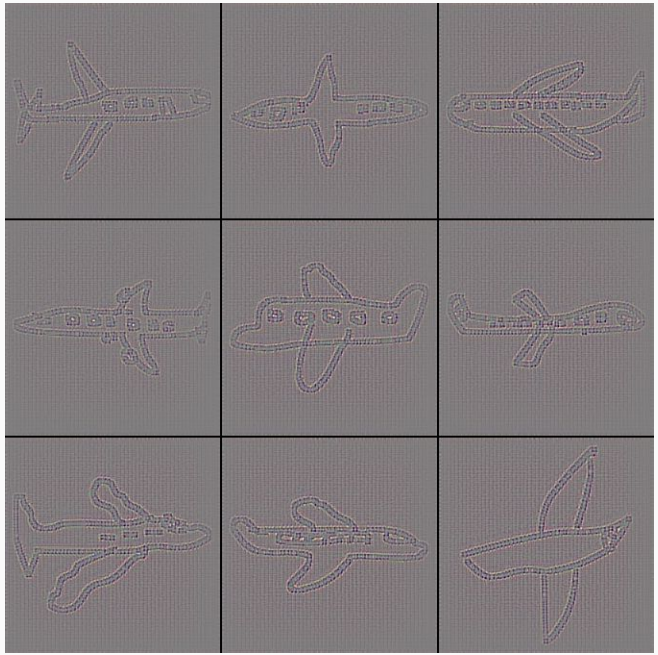
Deconvolution of the activated patch

- 224 x 224 Input Patch for 1st filter in Conv5_4 layer
- Filter sensitive to random classes with certain features.

Fully Connected layer



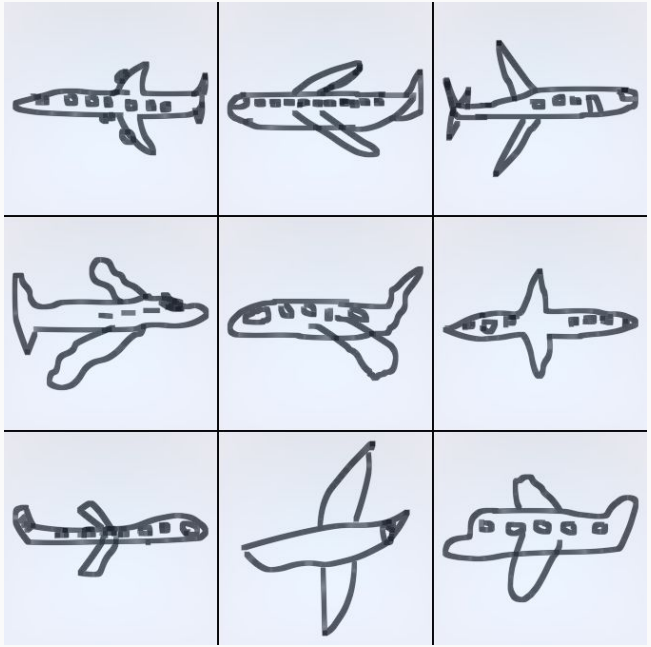
Input Patch for Maximum Activation



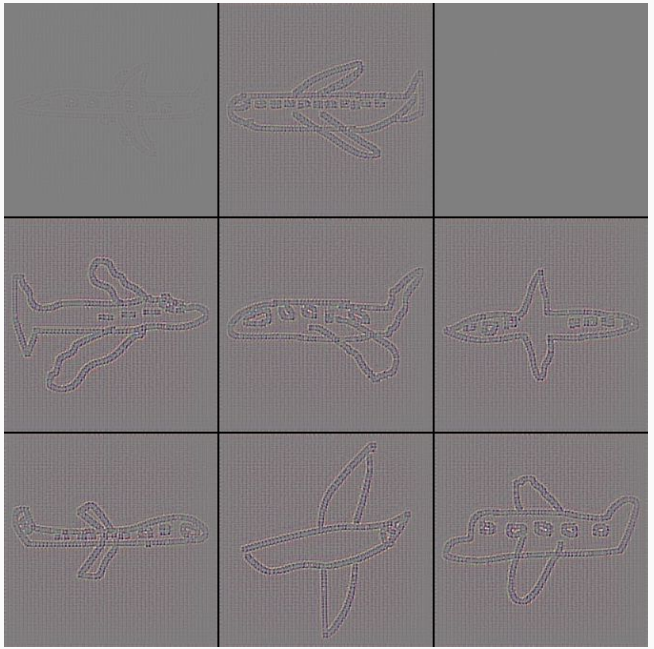
Deconvolution of the activated patch

- 224 x 224 Input Patch (complete image) for 1st filter in fully connected layer.
- Filter sensitive to the airplane class. (which is also 1st class in target attribute)

Softmax layer



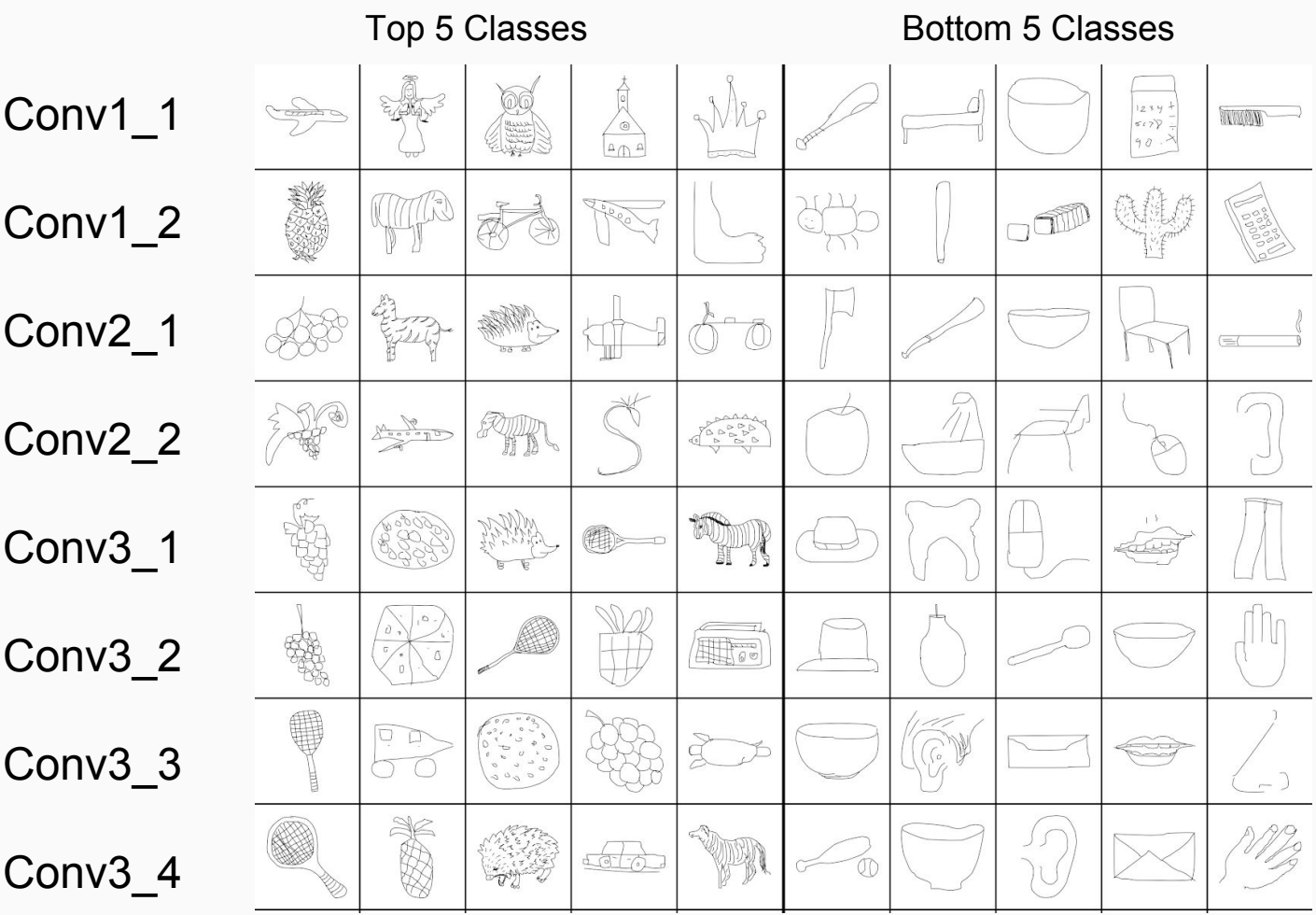
Input Patch for Maximum Activation

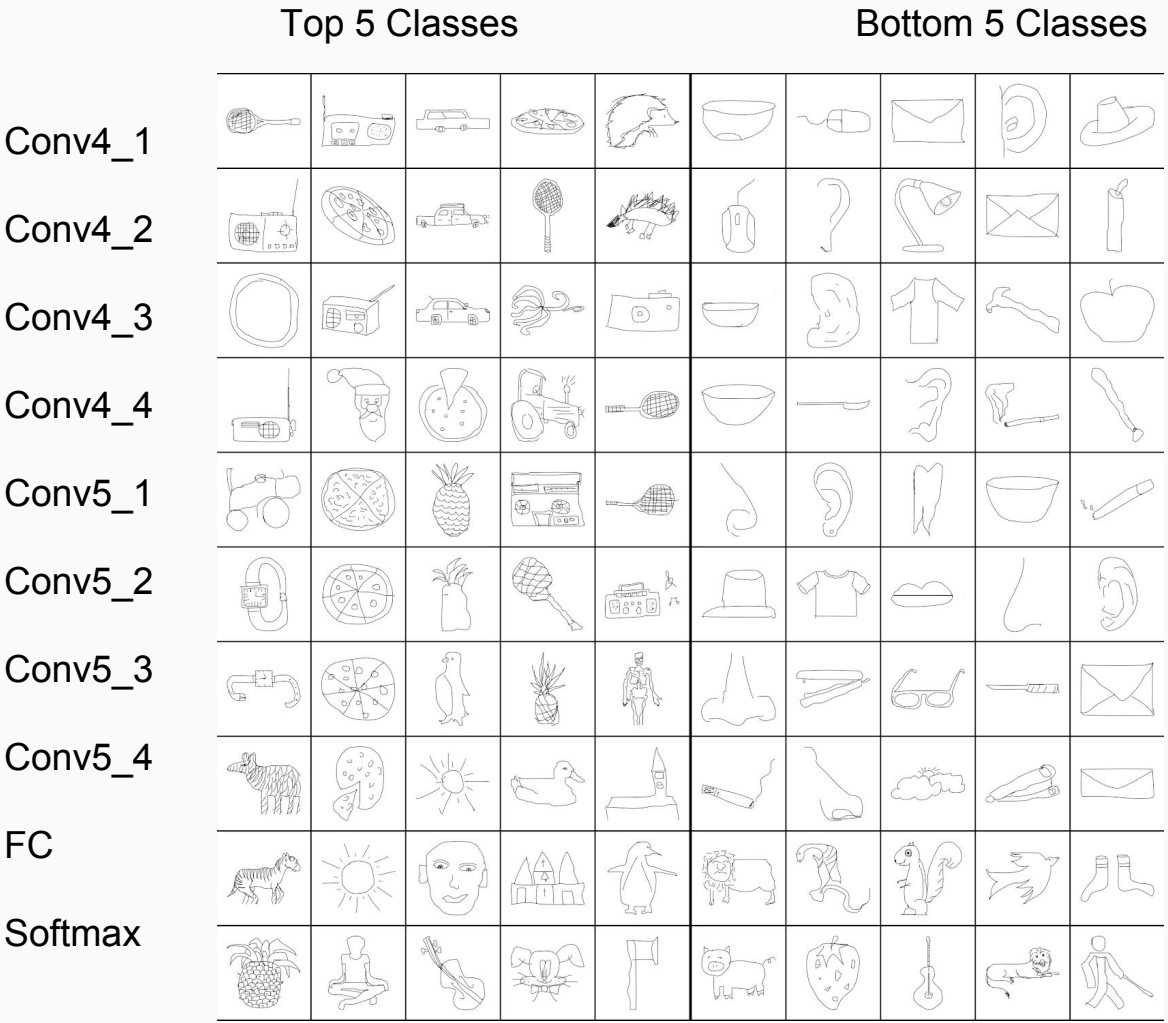


Deconvolution of the activated patch

- 227 x 227 Input Patch (complete image) for 1st filter in softmax layer.
- Filter sensitive to the airplane class. (which is also 1st class in target attribute)

Summary Plot (VGG-19 with Imagenet Mean)




Top 5 Classes
Bottom 5 Classes
Conv4_1
Conv4_2
Conv4_3
Conv4_4
Conv5_1
Conv5_2
Conv5_3
Conv5_4
FC
Softmax

Conclusion

- Filters in lower level layers are sensitive to edges, patterns, corners, lines and certain localized geometries present in sketches & in higher level layers are sensitive to certain classes & some specific objects in sketches.
- Deconvolution of the filters in VGG-19 shows more sharp input space activations than filters in the Alexnet.
- This comes in coherence with difference in performance where Alexnet gives 82% accuracy & VGG-19 gives 87% on validation dataset.
- Convolutional layers are selective to classes with high stroke density.

For next week ...

- Extending the usage of Yosinski's Deep Visualization Toolbox to analyze sketch CNNs fine-tuned for sketches (GoogLeNet, ResNet)
- Visualizing & Understanding RNN to know what different states capture.
- Use LSTMVis toolbox for sketches.

Thank You