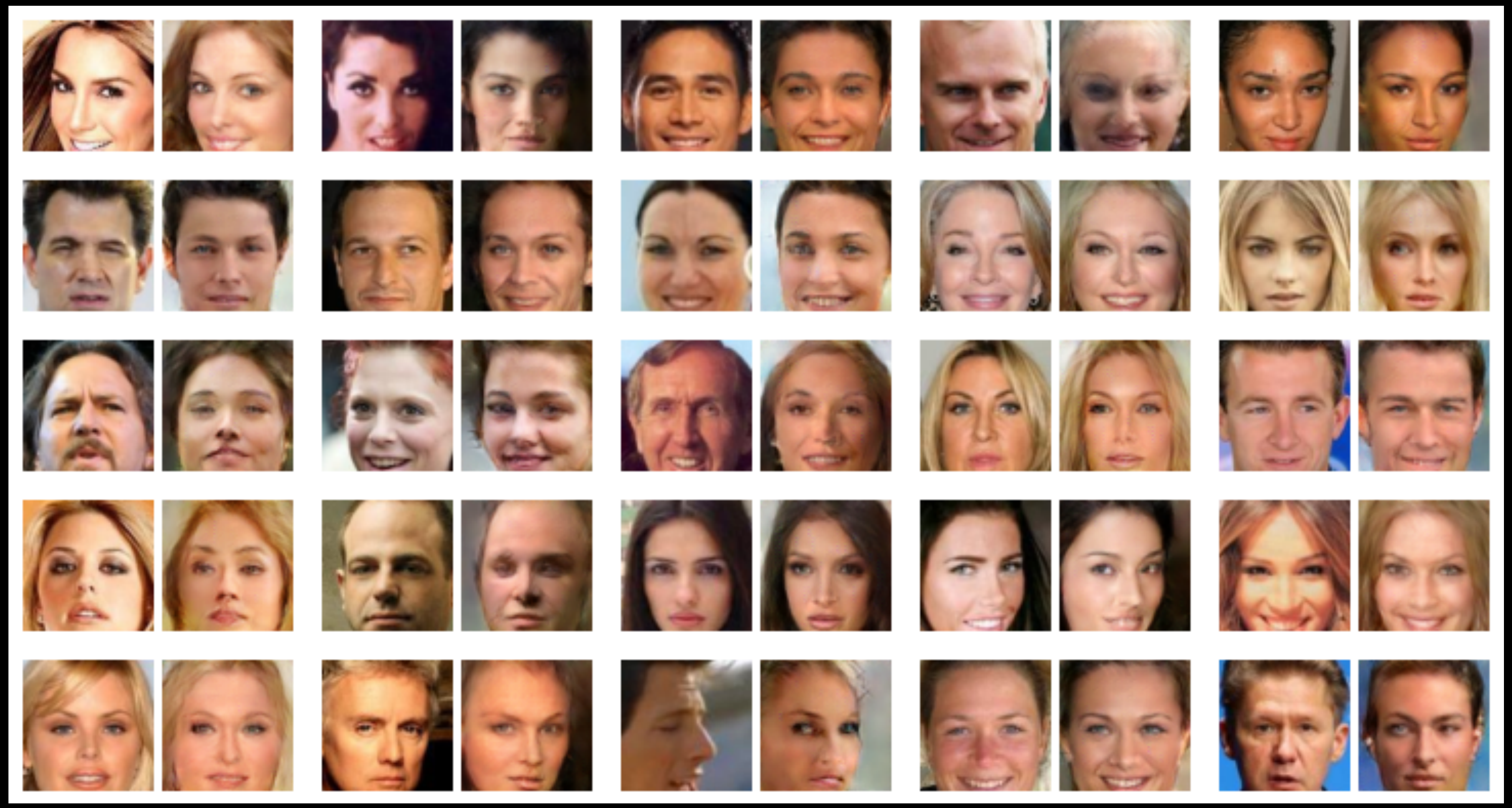
DL Dev Course: Week 05
Advanced CNNs

RED DRAGON AI

f() = kitten

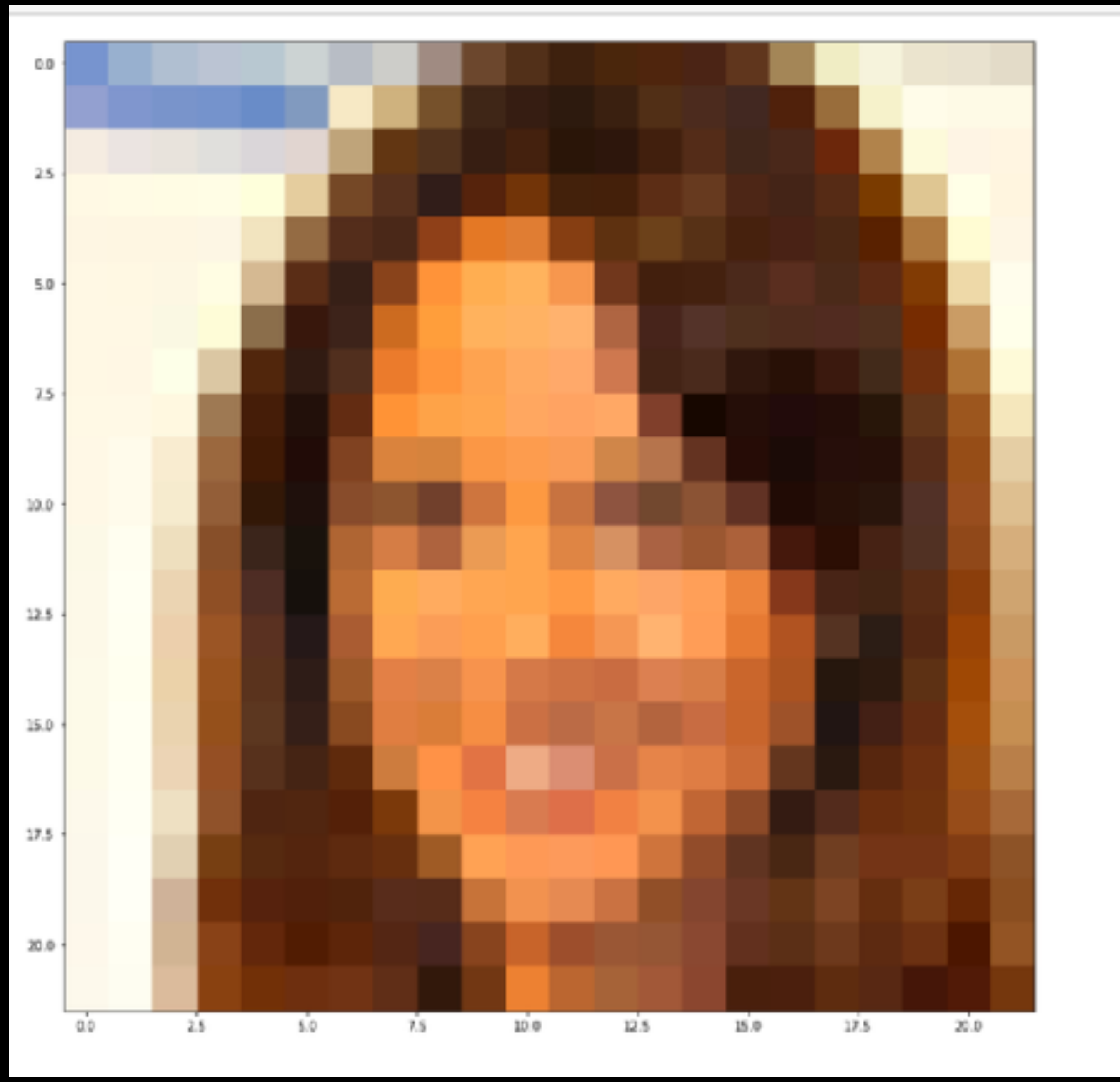
Classification

f() = 

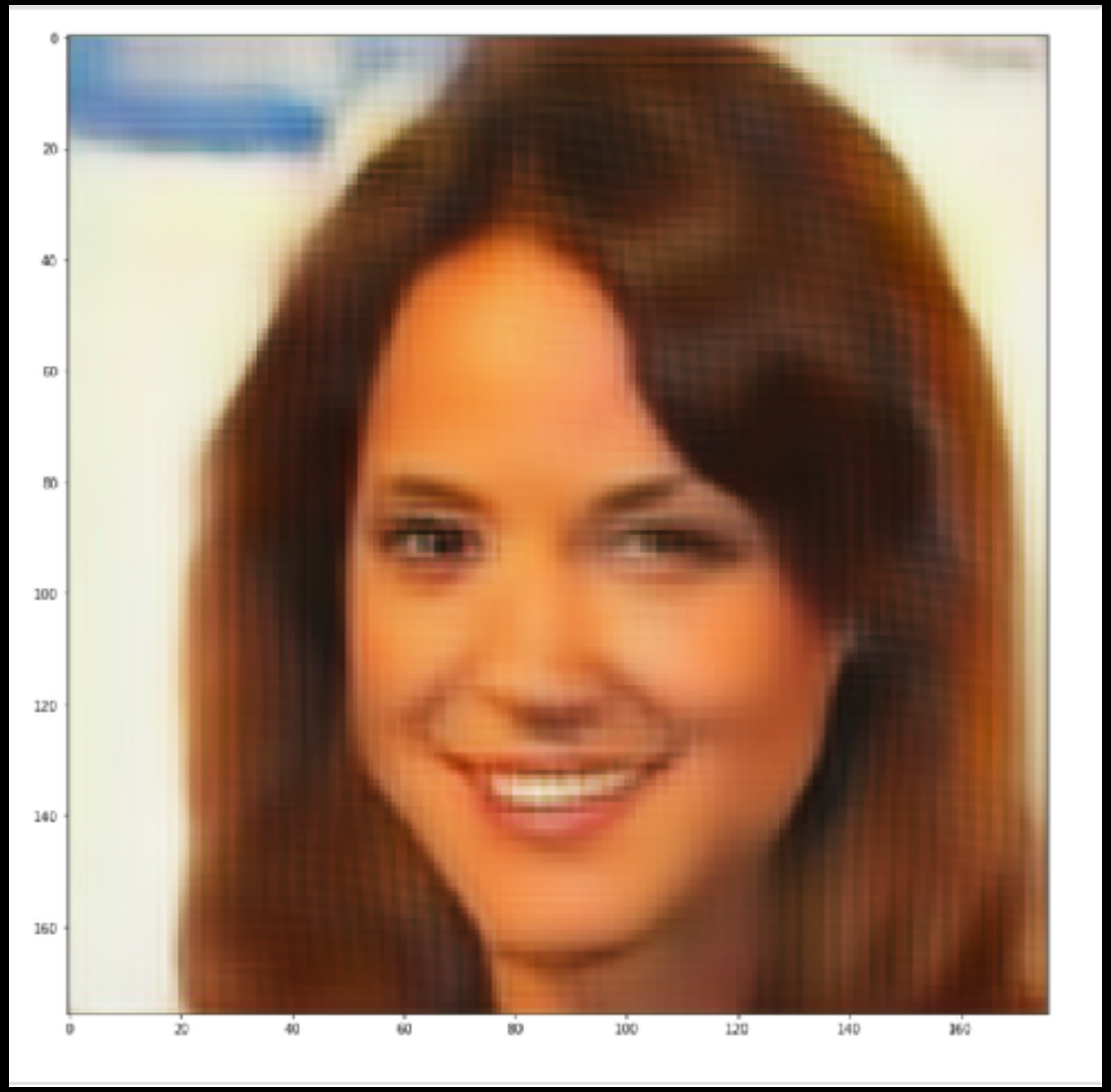
Generative

8x Super Resolution

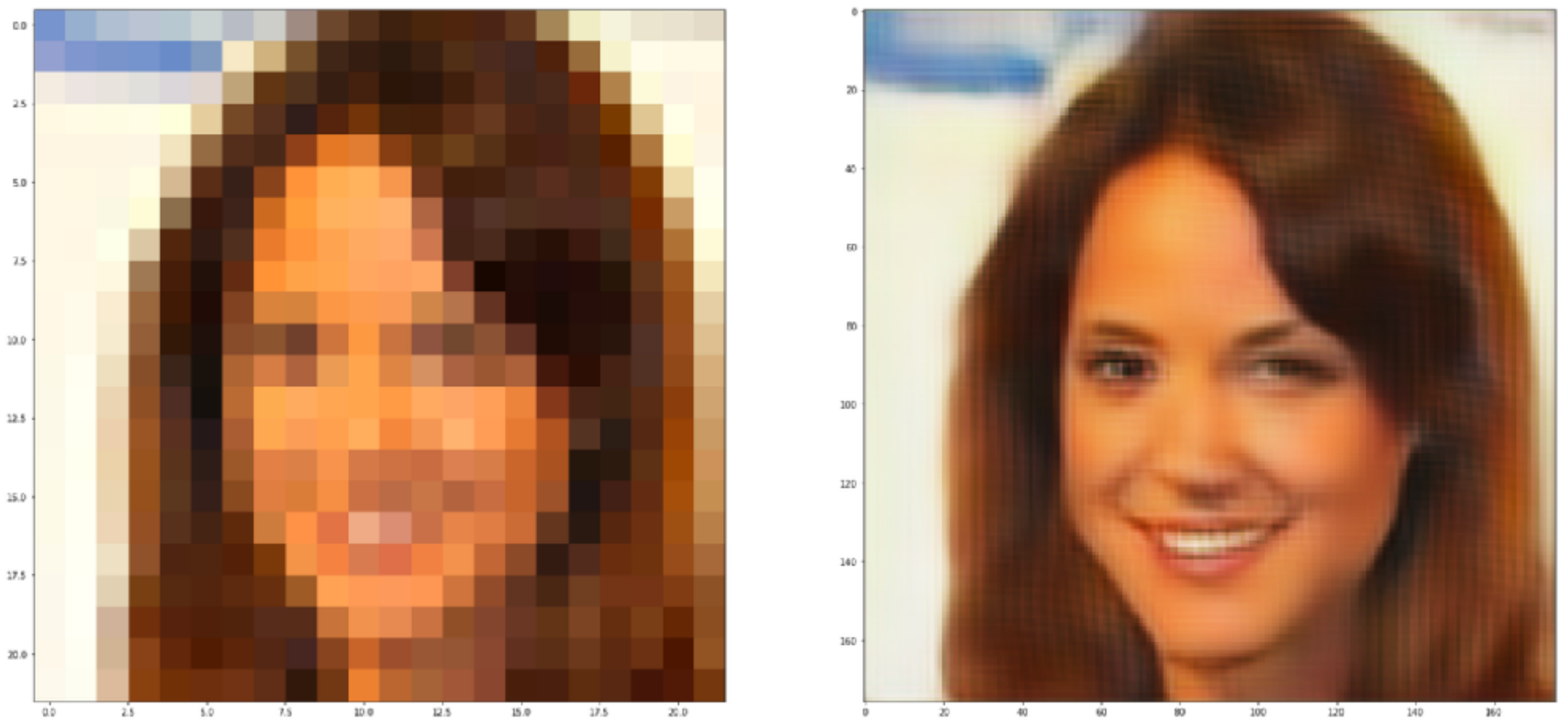
f(



) =

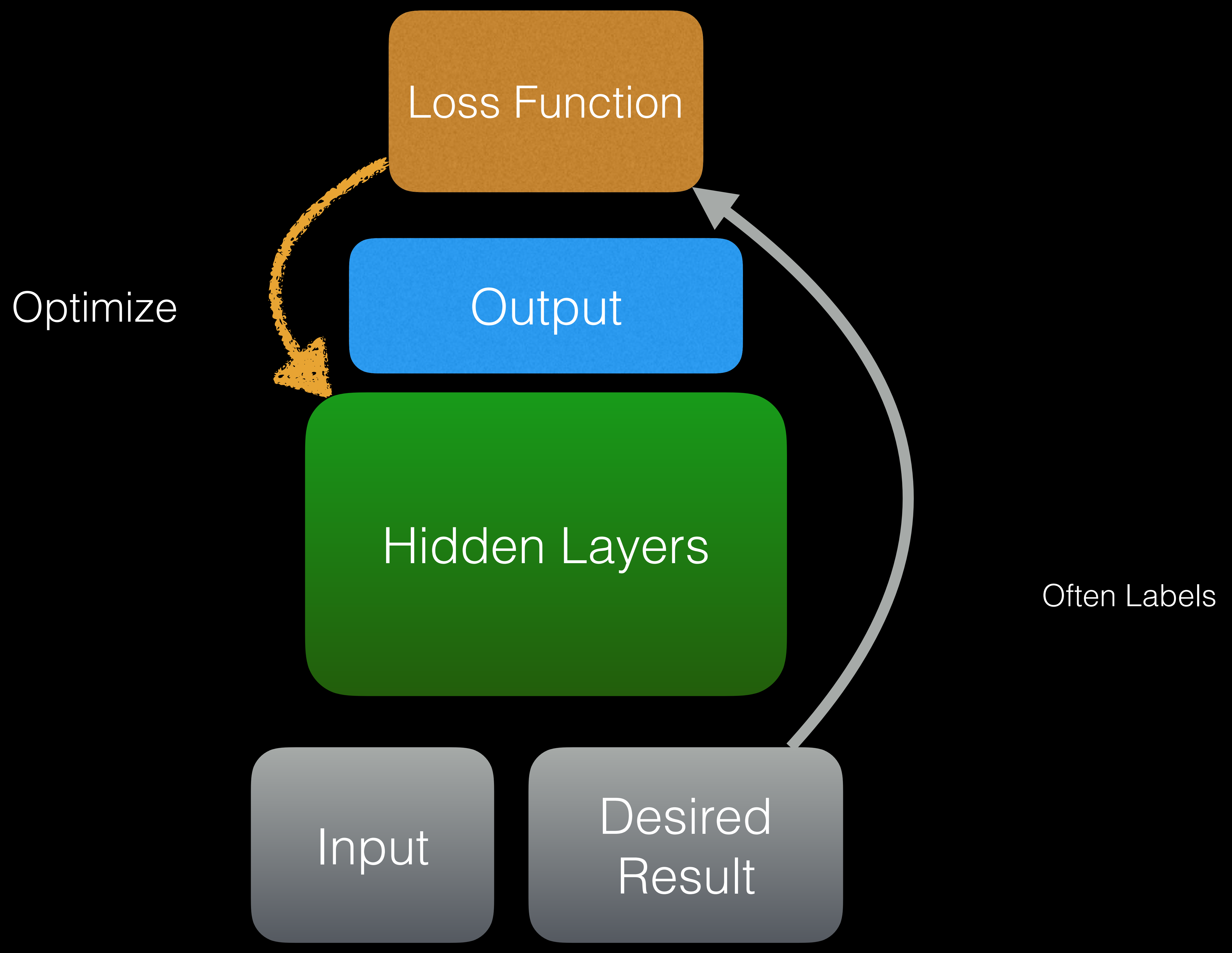


In [35]:
```
compare_pics(arr_test_elr[24].astype('uint8'),p[24].astype('uint8'))
```

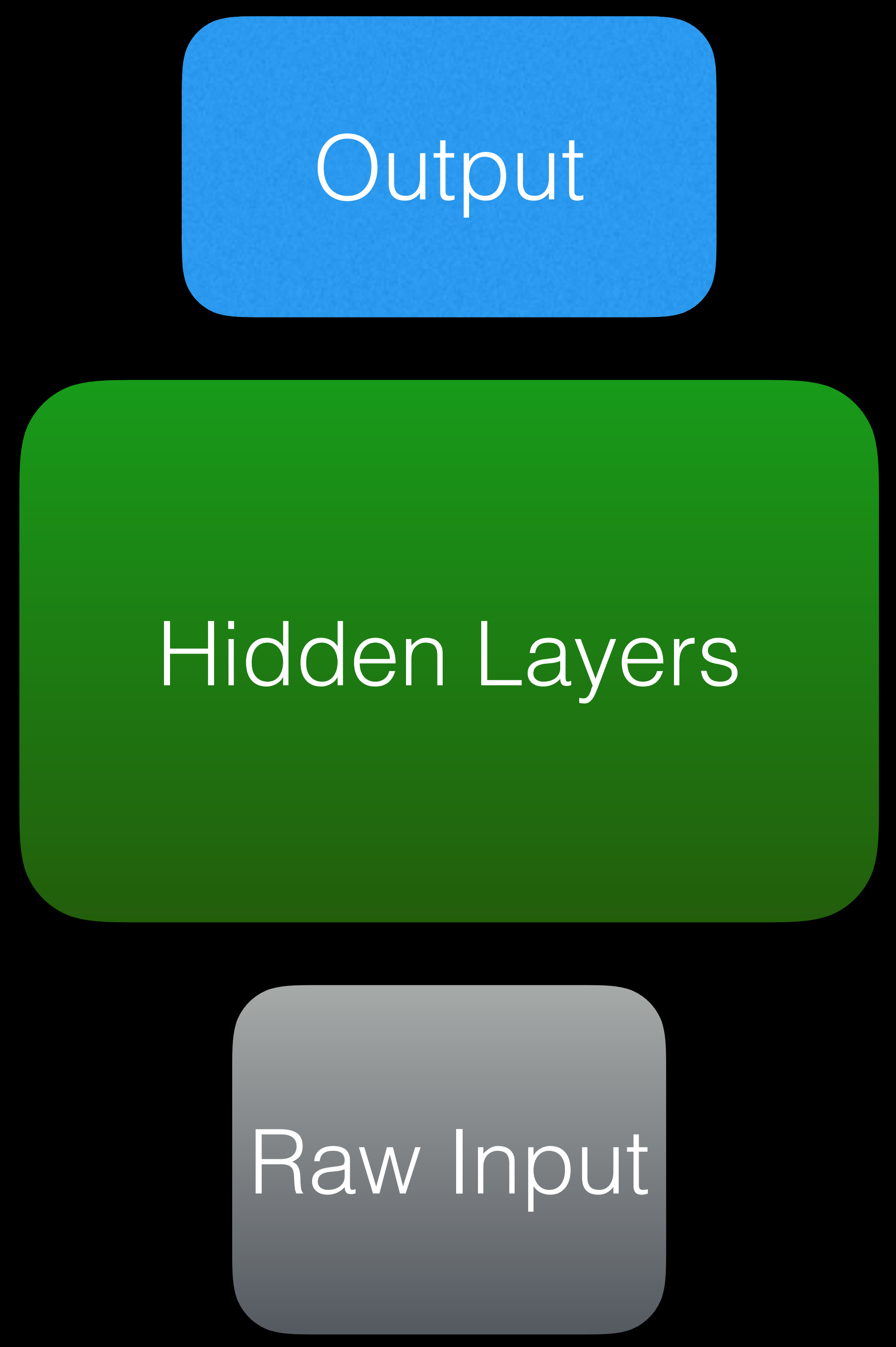


In [45]:
```
compare_pics(arr_test_hr[24].astype('uint8'),p[24].astype('uint8'))
```

Neural Network



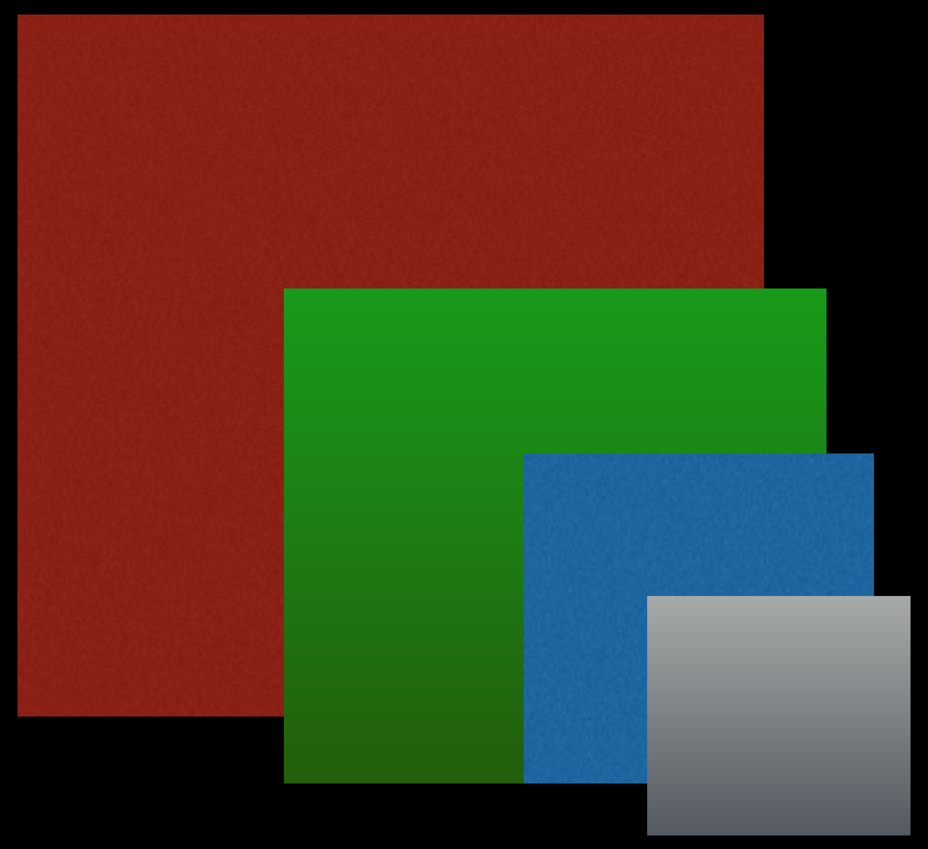
Neural Network Inference



No More Loss or Optimize

No More Labels

Classification CNN



Conv Blocks(Conv+Pooling) Relu

Reduction of HxW
increased number of filters(K)

FCN layers

Prediction

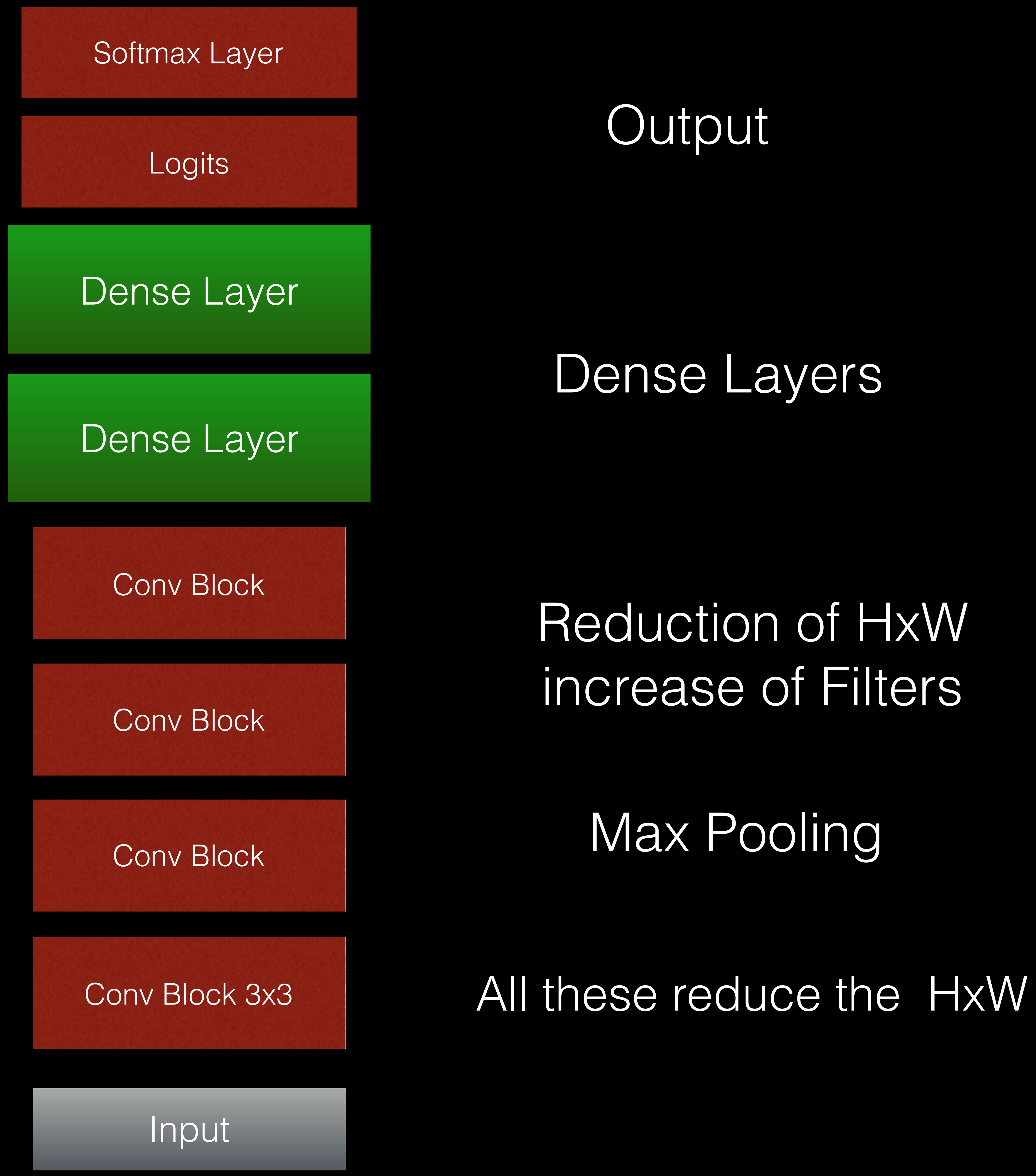
The Magic of CNNs

- CNNs are not just for classification

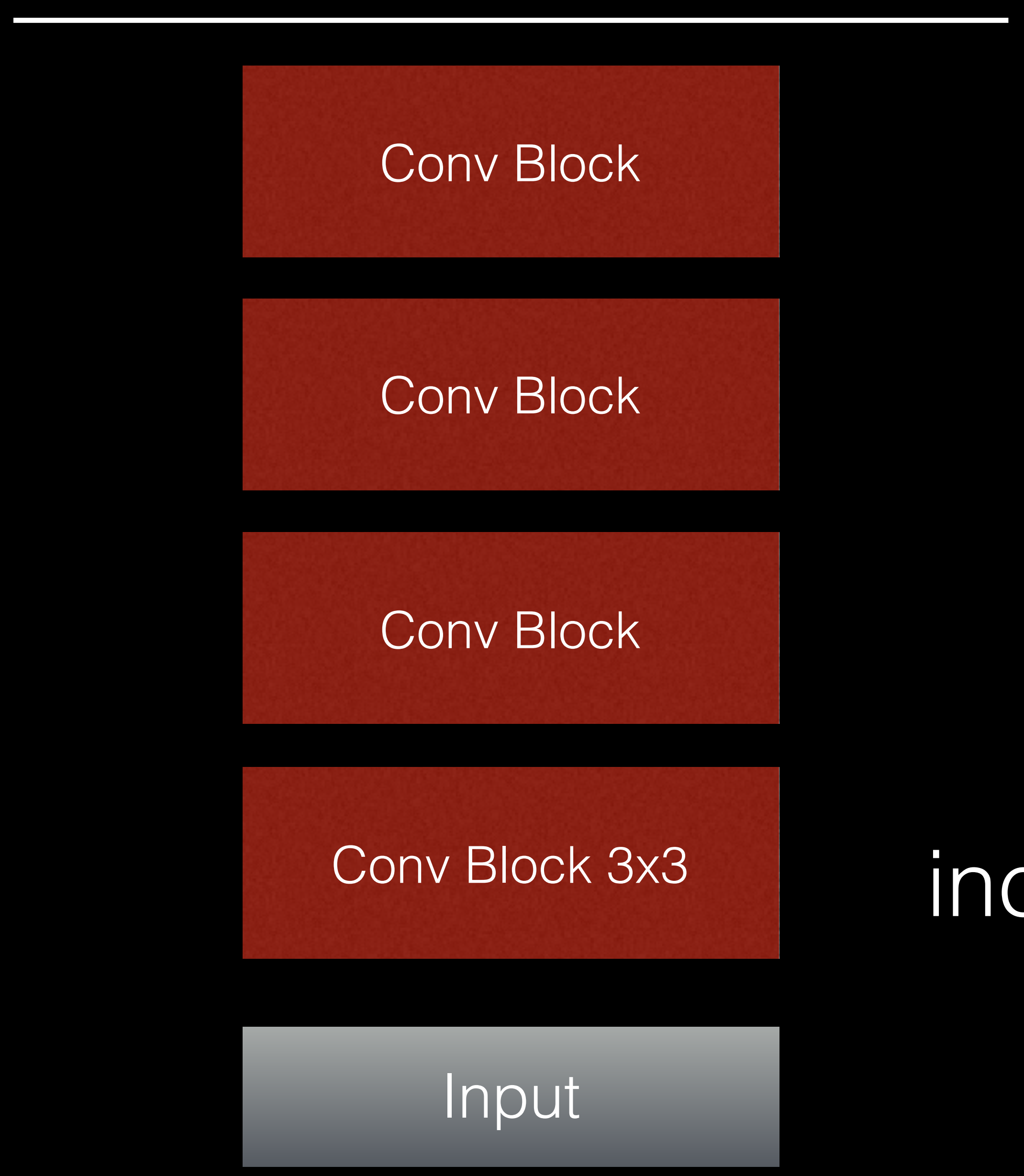
- Don’t fall into the trap of thinking CNNs are just a precursor to a set of dense layers, logits and classes

- CNNs give us rich feature representations of what we put into them, if we cut off the dense layers and we use convolution blocks as our outputs

Old Style CNN



Modified CNN



Output

very deep (k) output
eg. lots of filters

Reduction of HxW
increased number of filters

The paper

Perceptual Losses for Real-Time Style Transfer and Super-Resolution

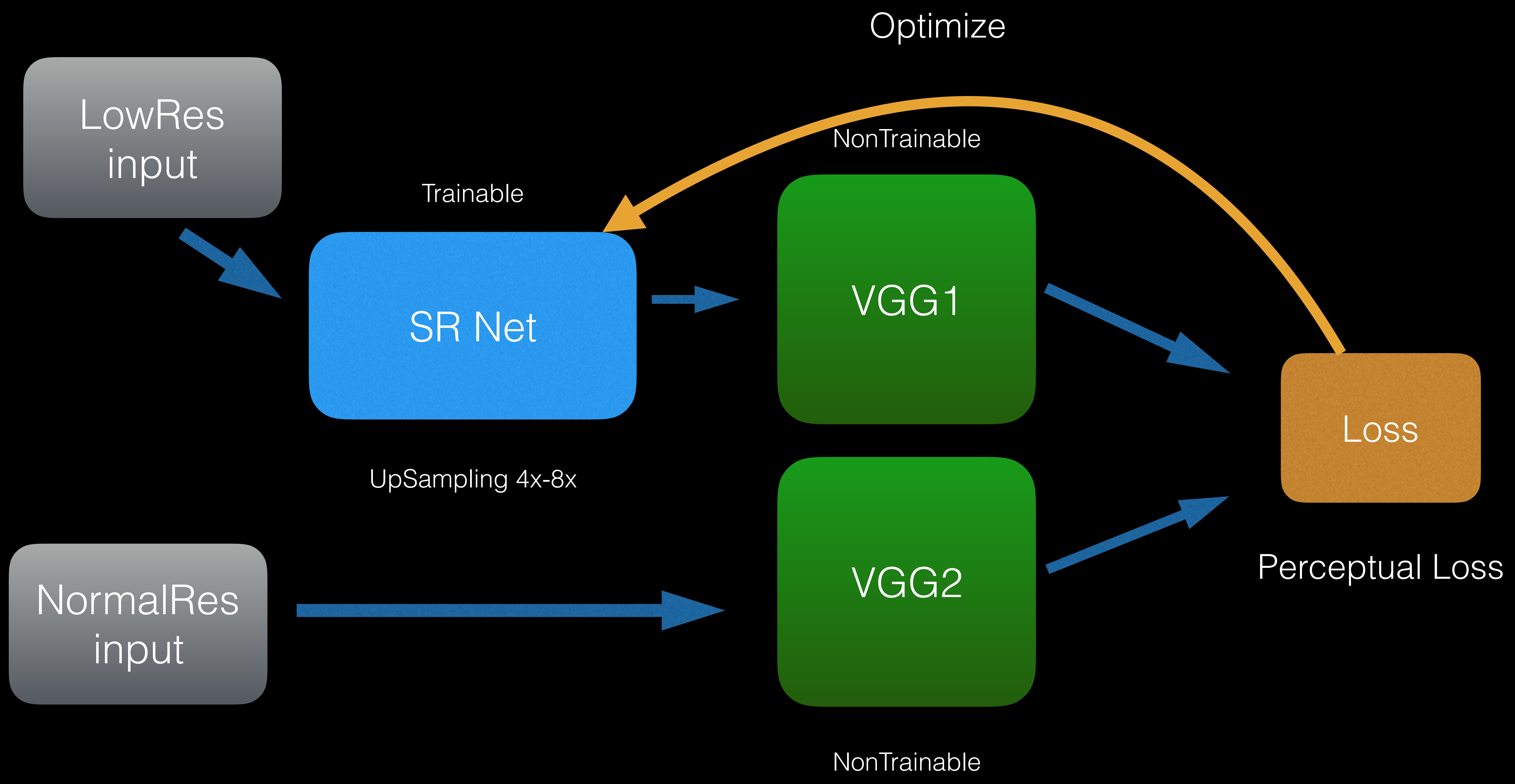
Justin Johnson, Alexandre Alahi, Li Fei-Fei
{jcjohns, alahi, feifeili}@cs.stanford.edu

Department of Computer Science, Stanford University

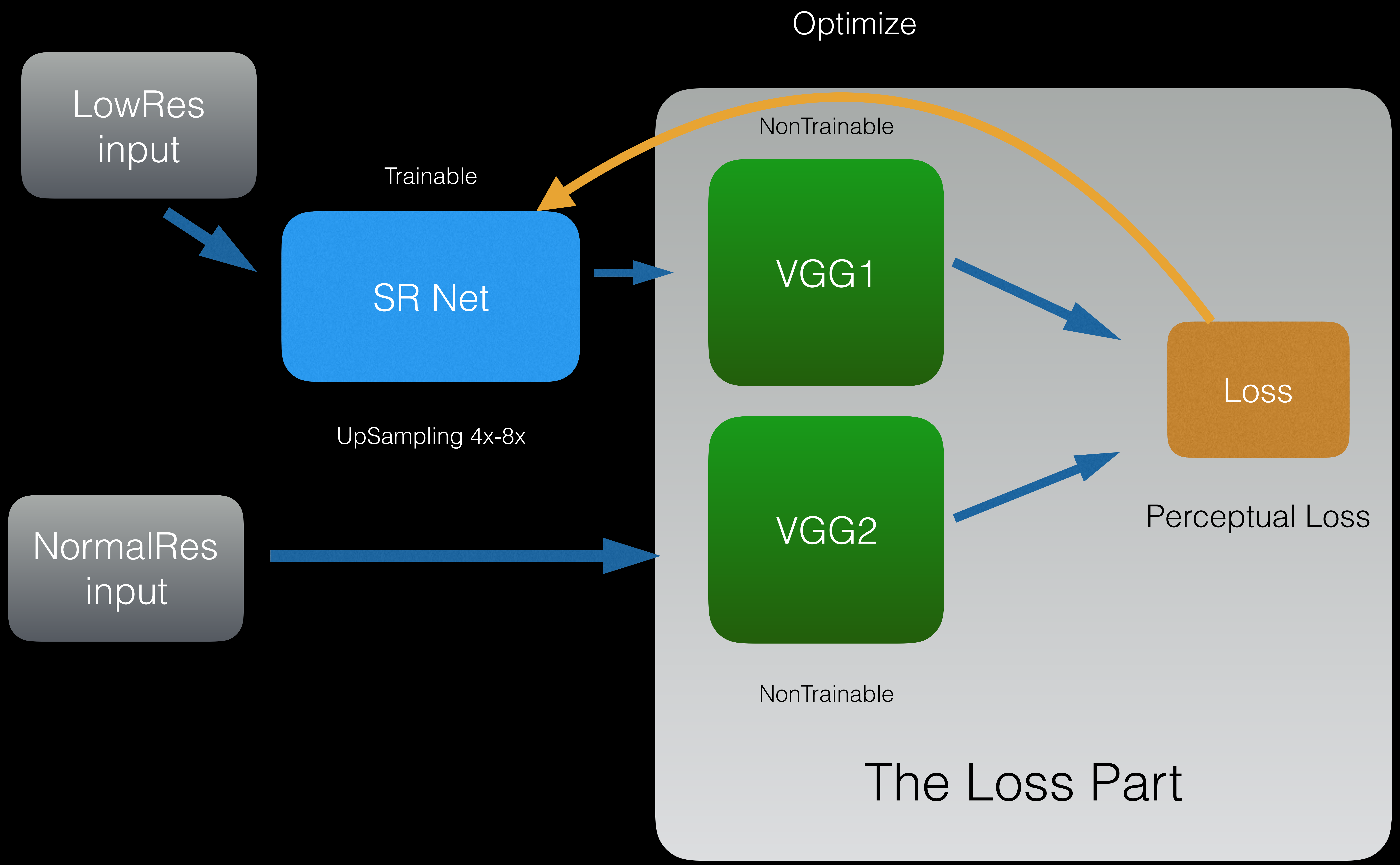
Abstract. We consider image transformation problems, where an input image is transformed into an output image. Recent methods for such problems typically train feed-forward convolutional neural networks using a *per-pixel* loss between the output and ground-truth images. Parallel work has shown that high-quality images can be generated by defining and optimizing *perceptual* loss functions based on high-level features extracted from pretrained networks. We combine the benefits of both ap-

"Perceptual Losses for Real-Time Style Transfer and Super-Resolution" by Johnson et.al
http://arxiv.org/abs/1603.08155

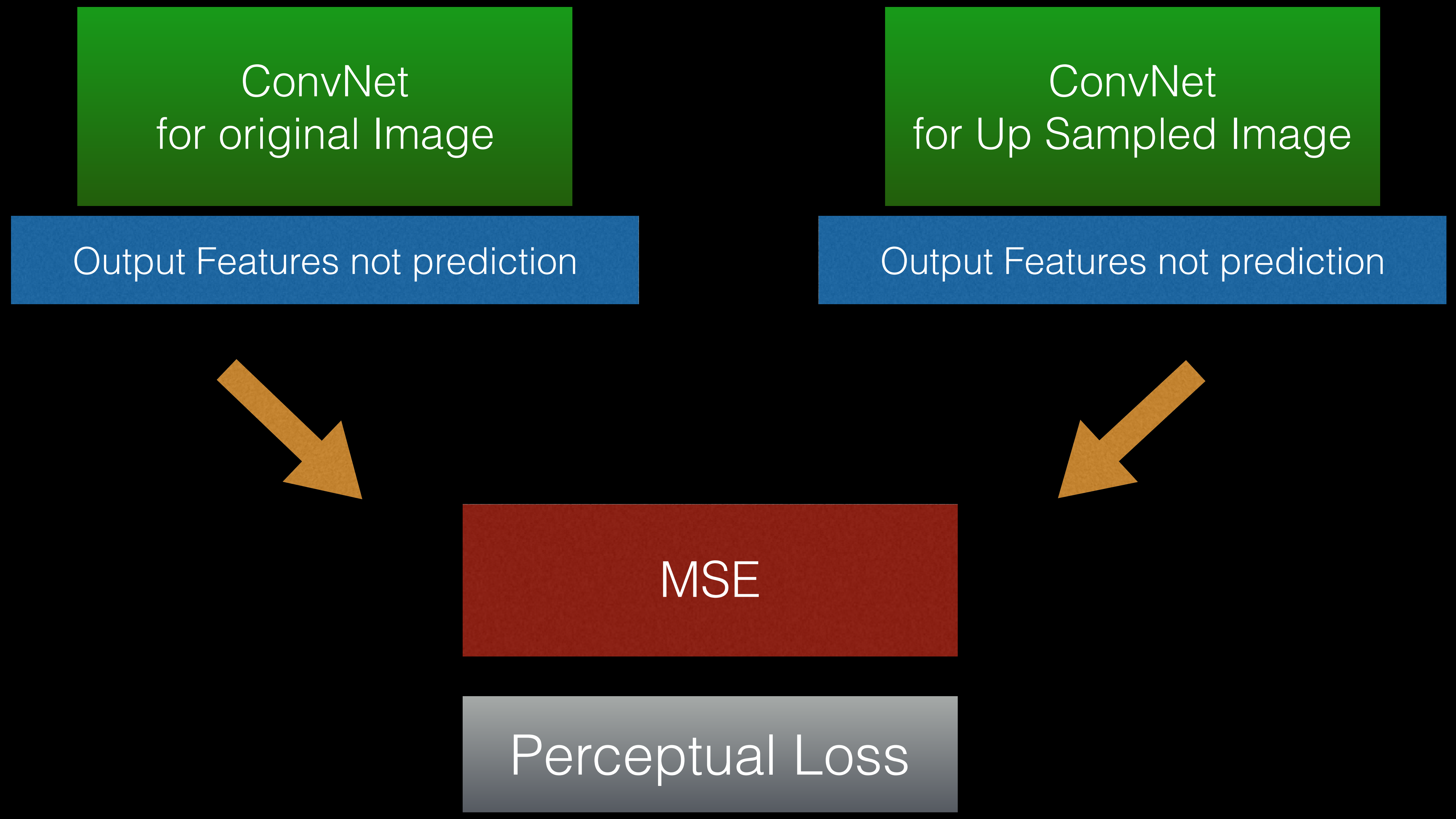
The SR Network



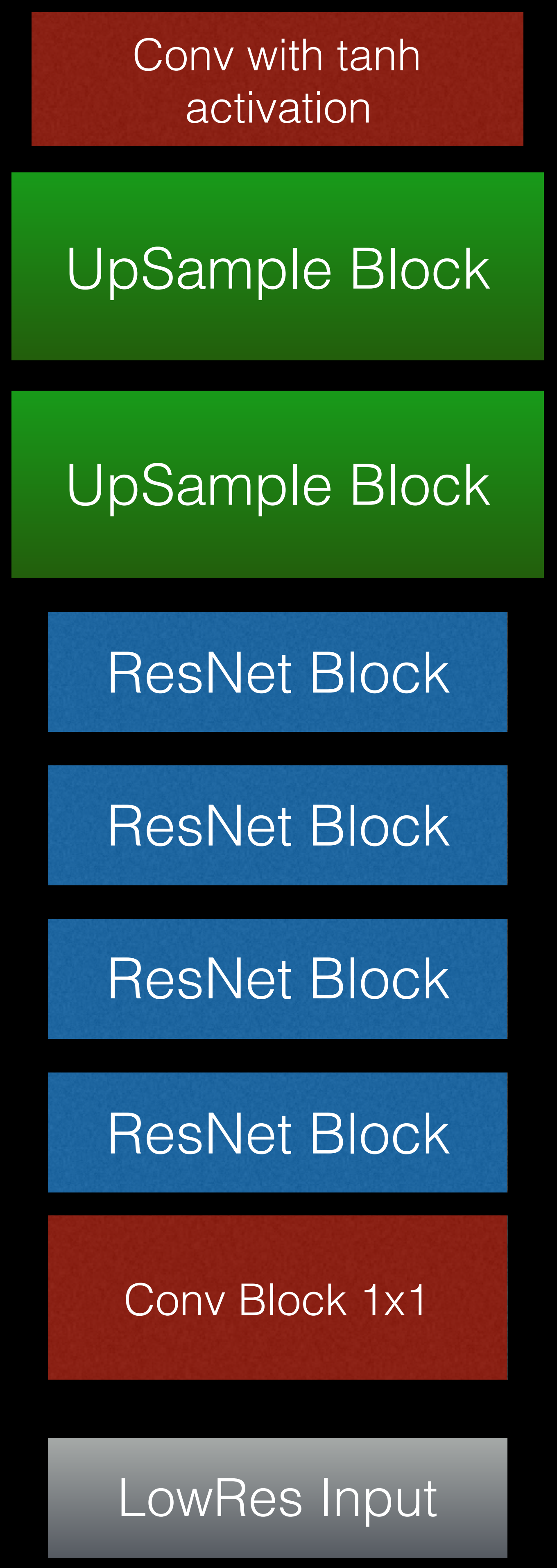
The SR Network



Perceptual Loss



Up Sampling CNN



Output

Increasing Pixels

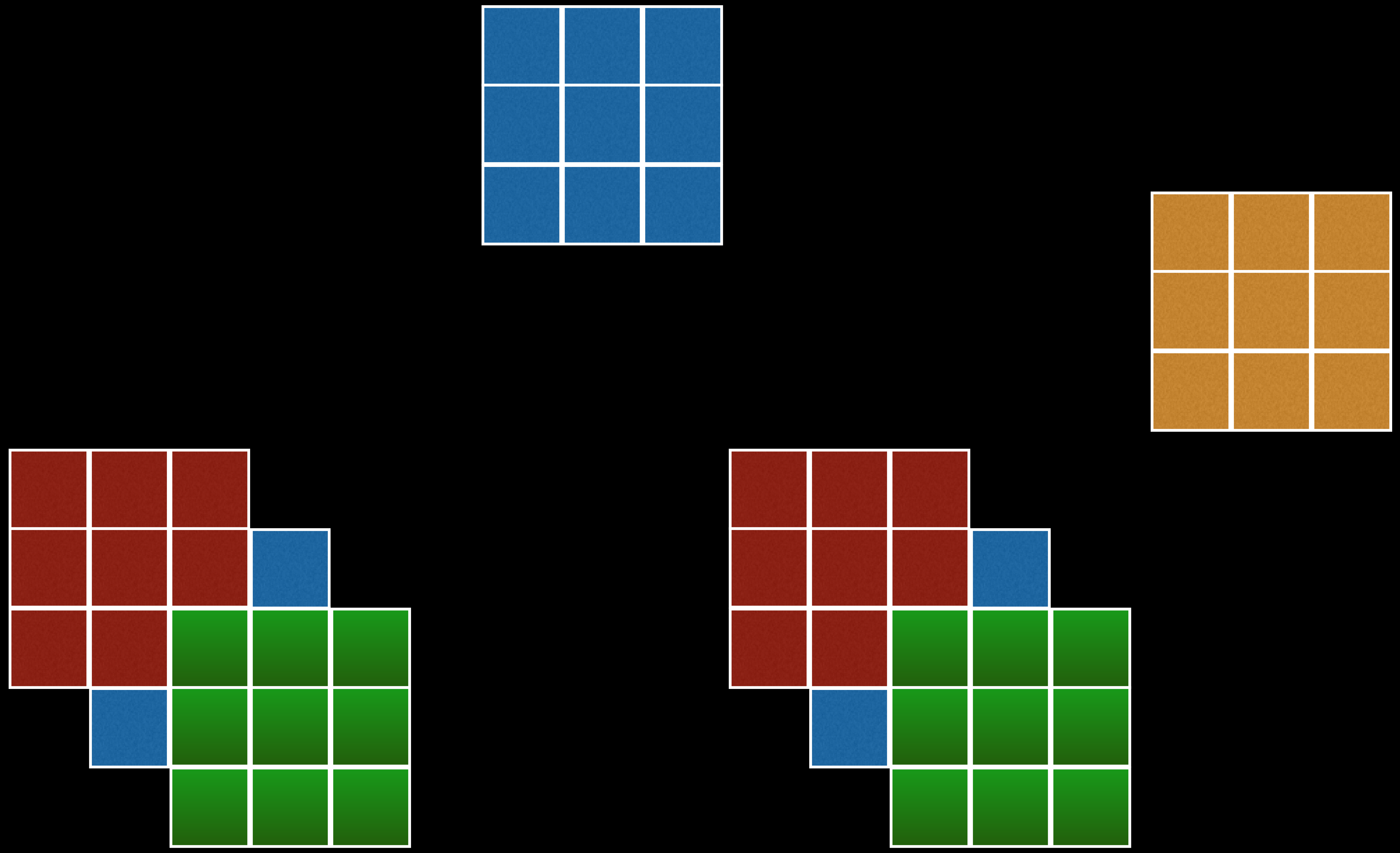
Reduction of HxW
increase of Filters

What are these pixels?
Done at smaller image size

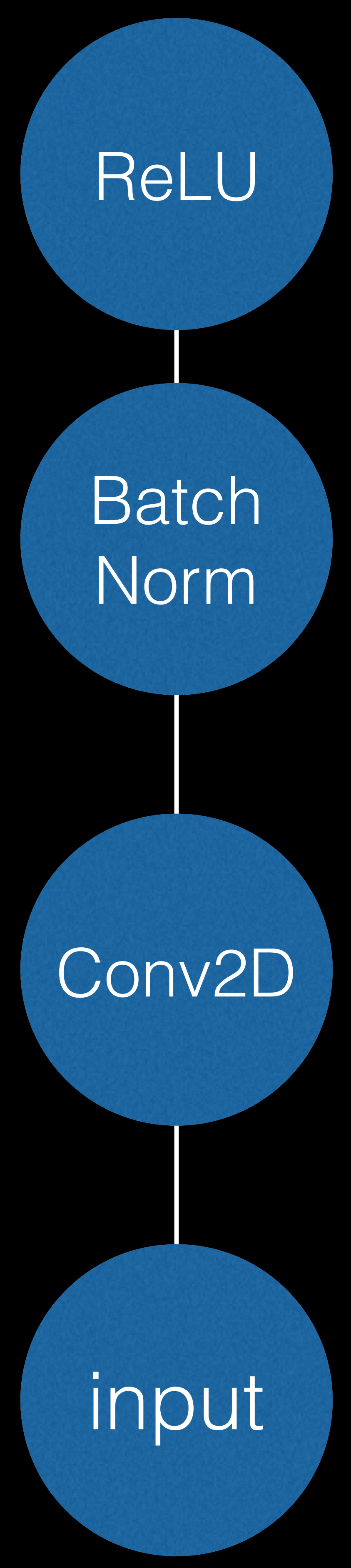
Increase Receptive field

Receptive Field

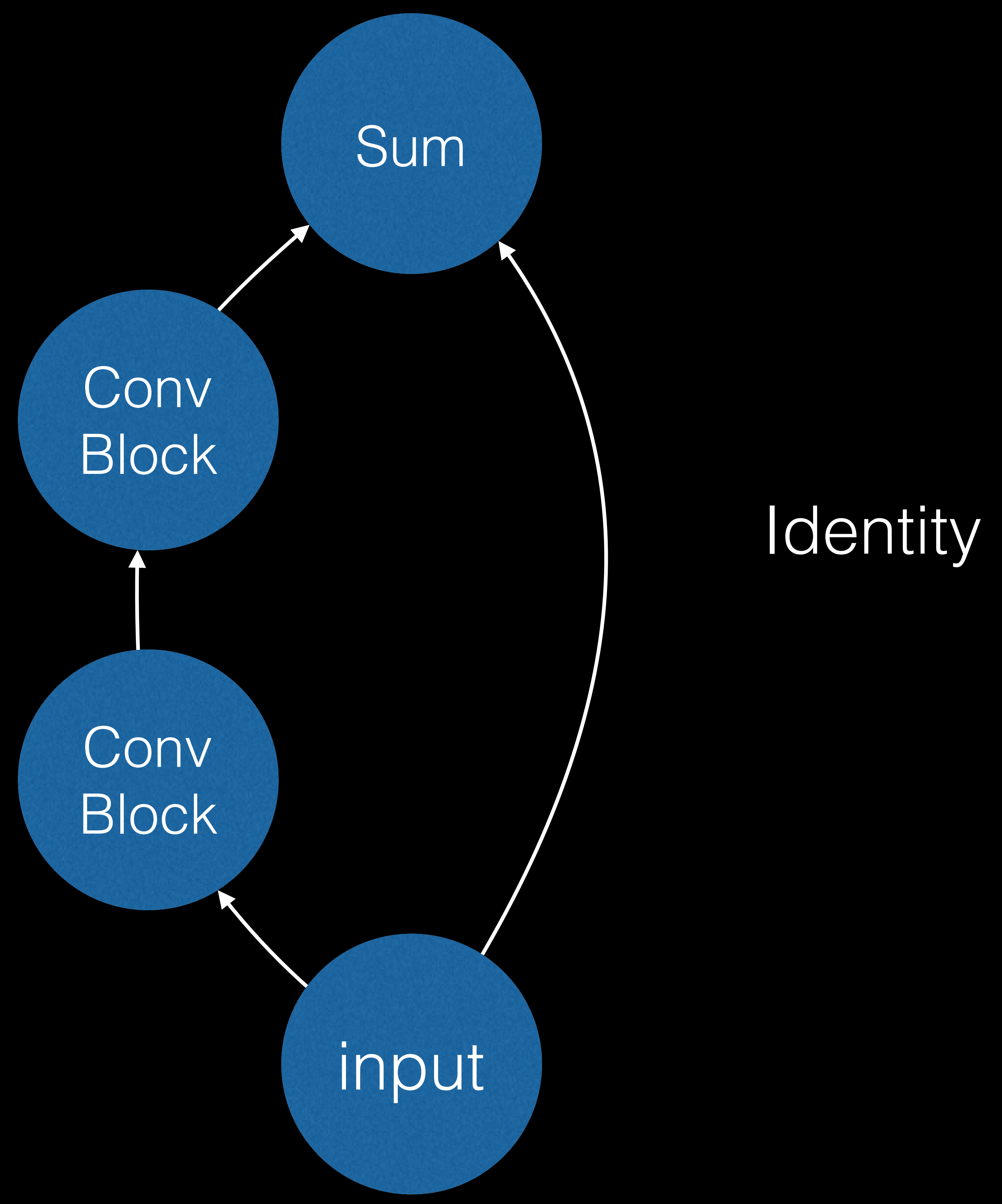
3x3 filter on 3x3 input give 5x5 receptive field



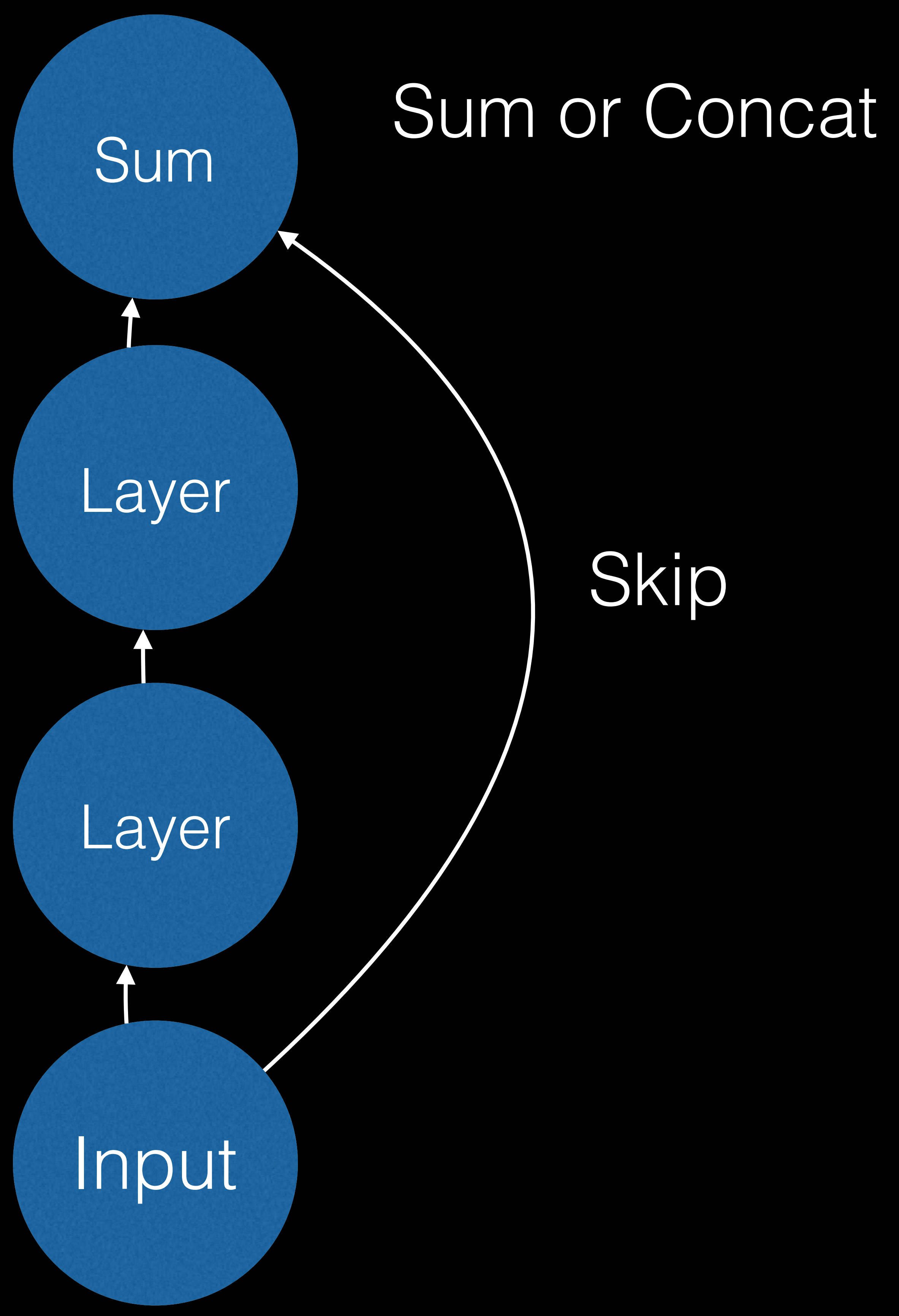
Conv Block



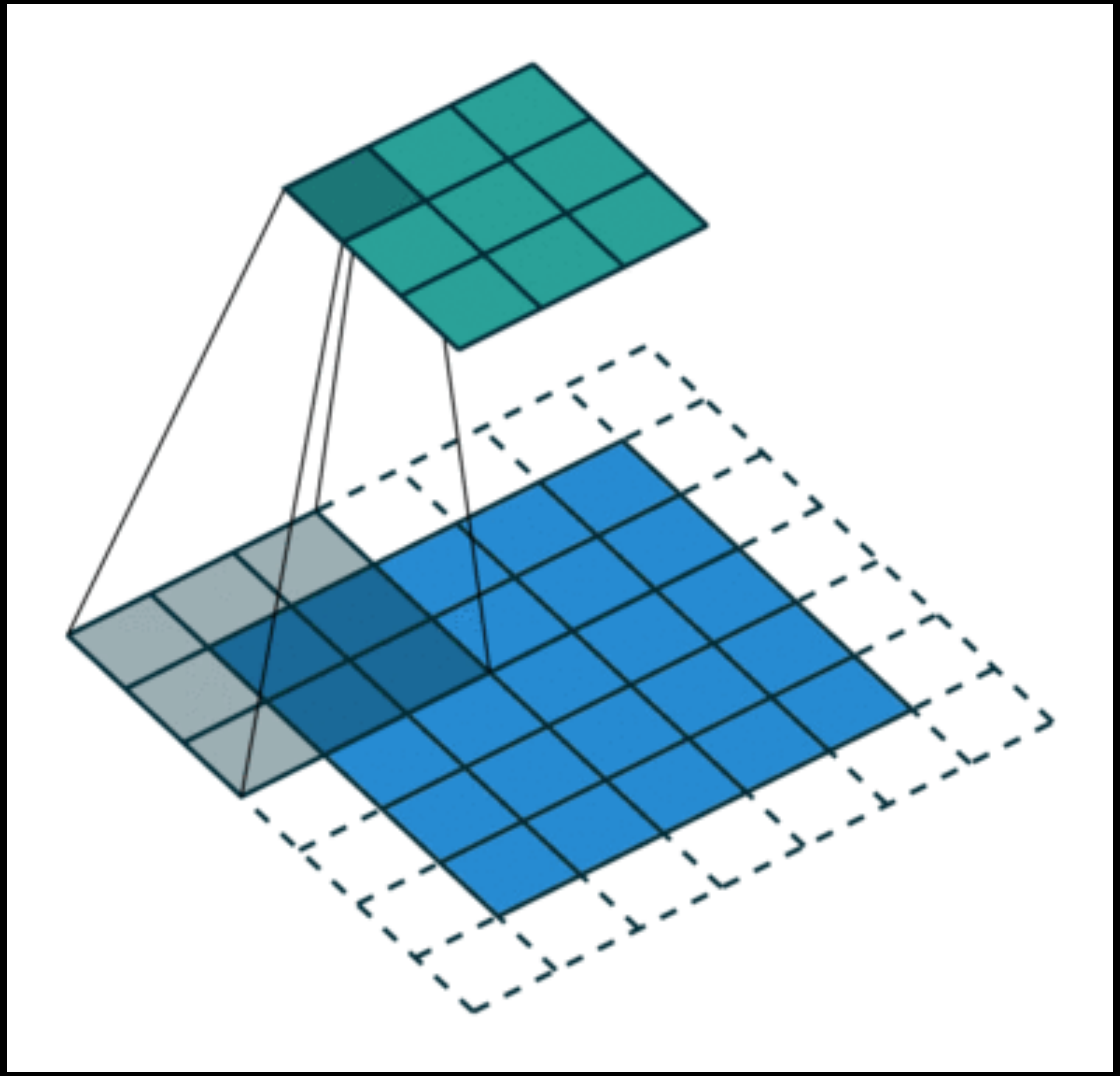
ResNet Block



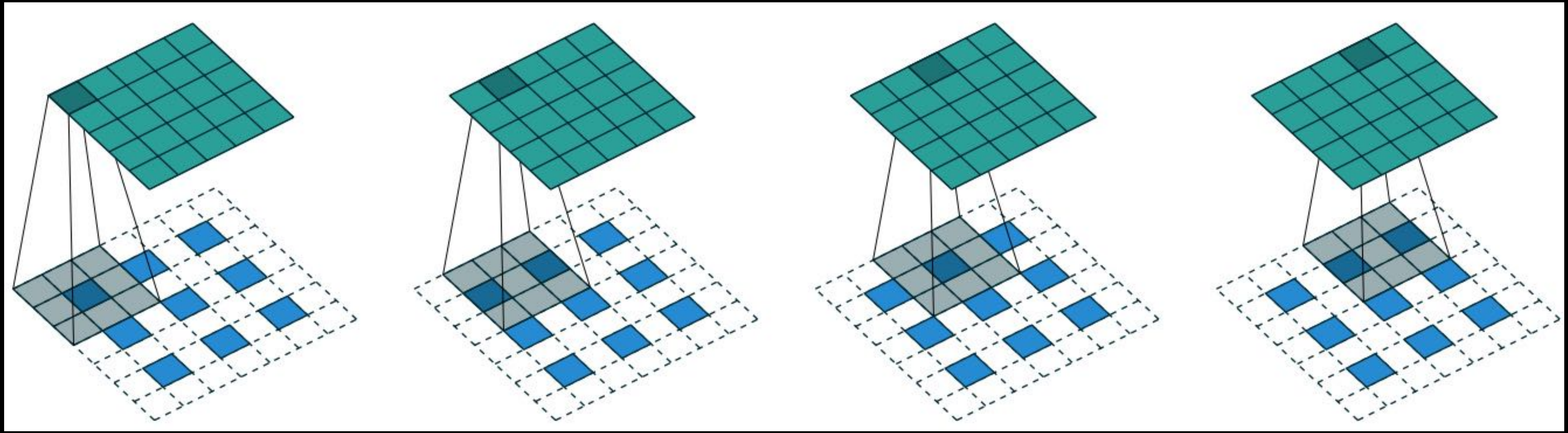
Skip Connections



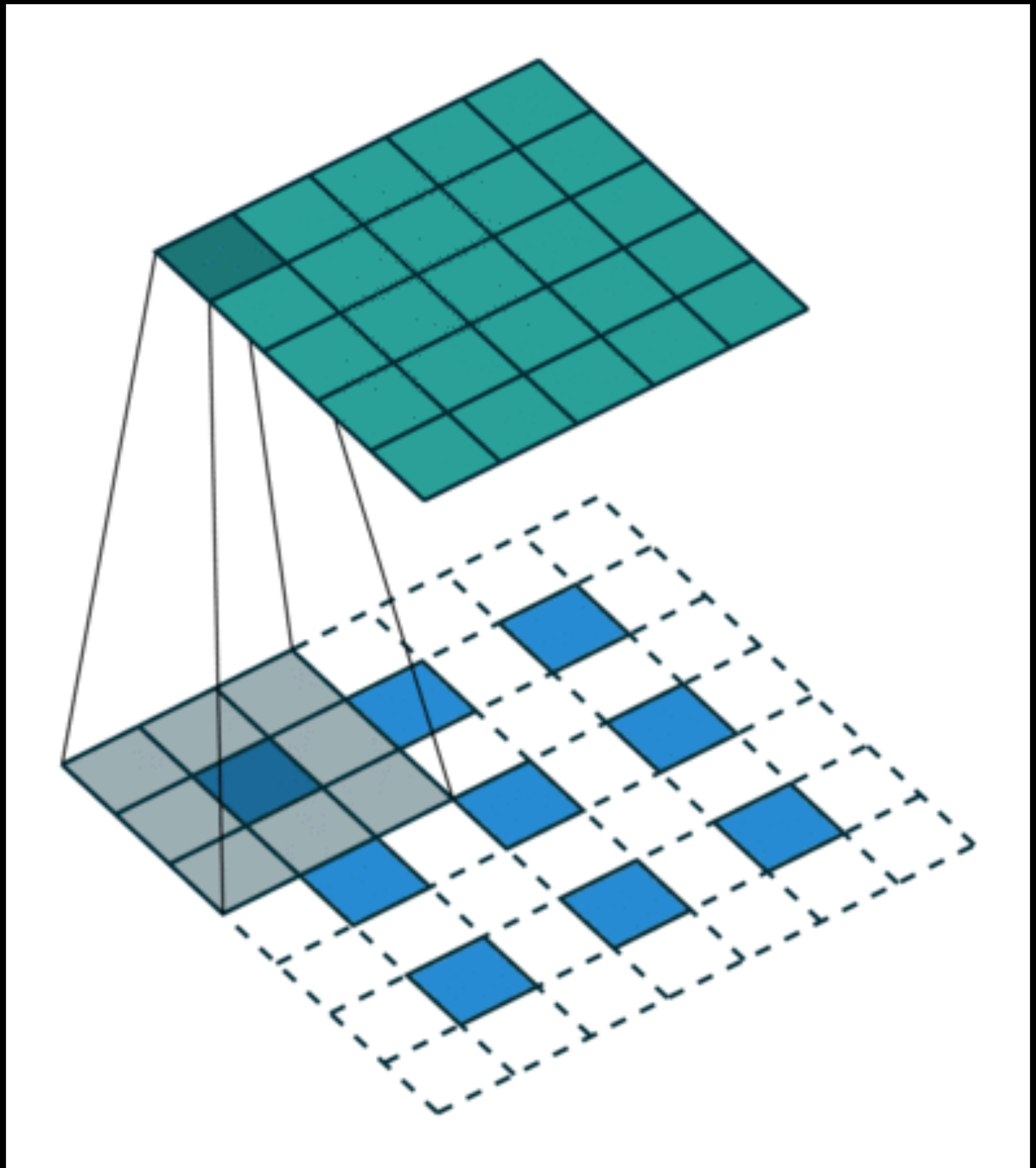
Normal Convolutions



Transposed Convolutions



Transposed Convolutions



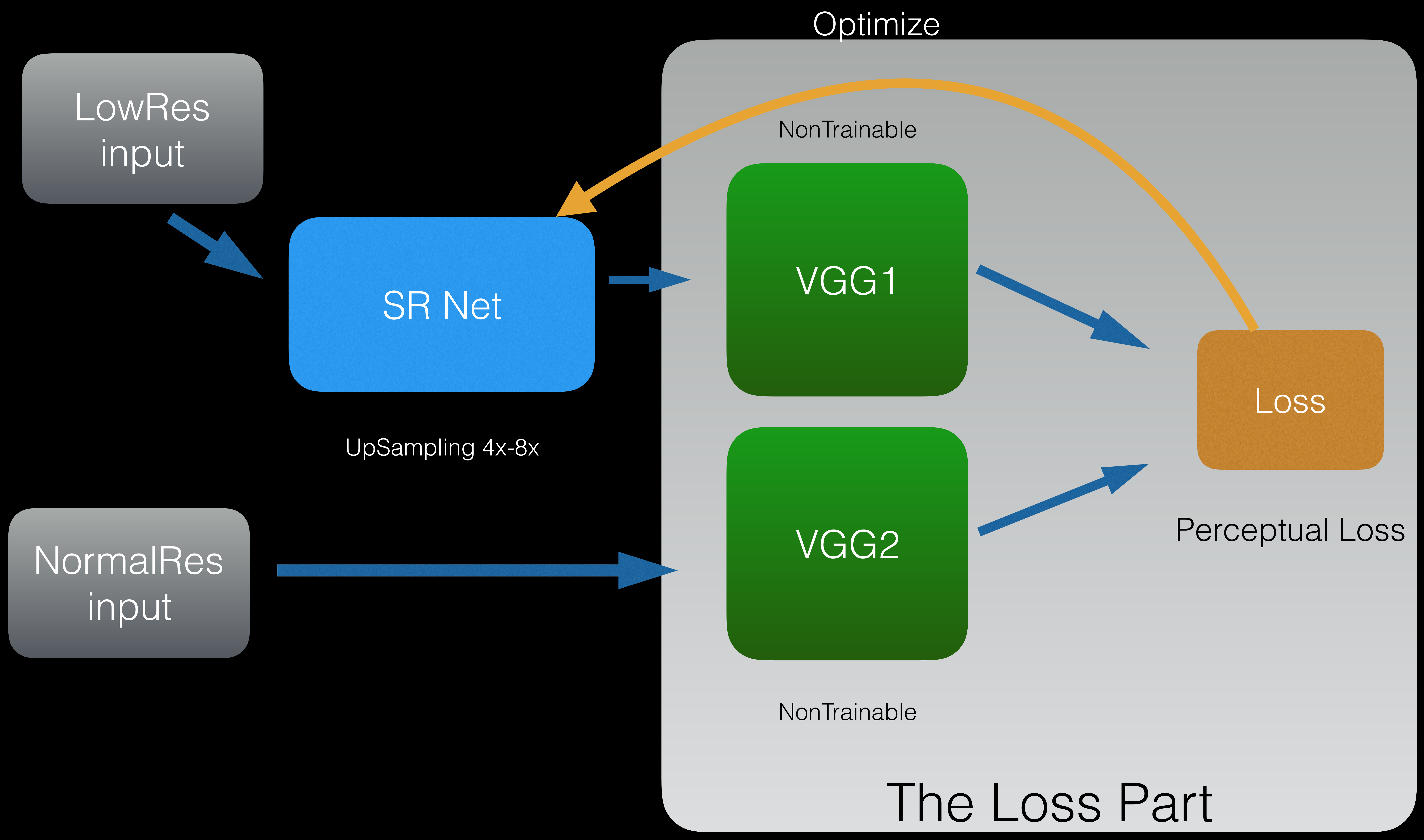
Transposed Convolutions

Deconvolutions (not correct)

Transposed Convolutions

Fractional Strided Convolutions

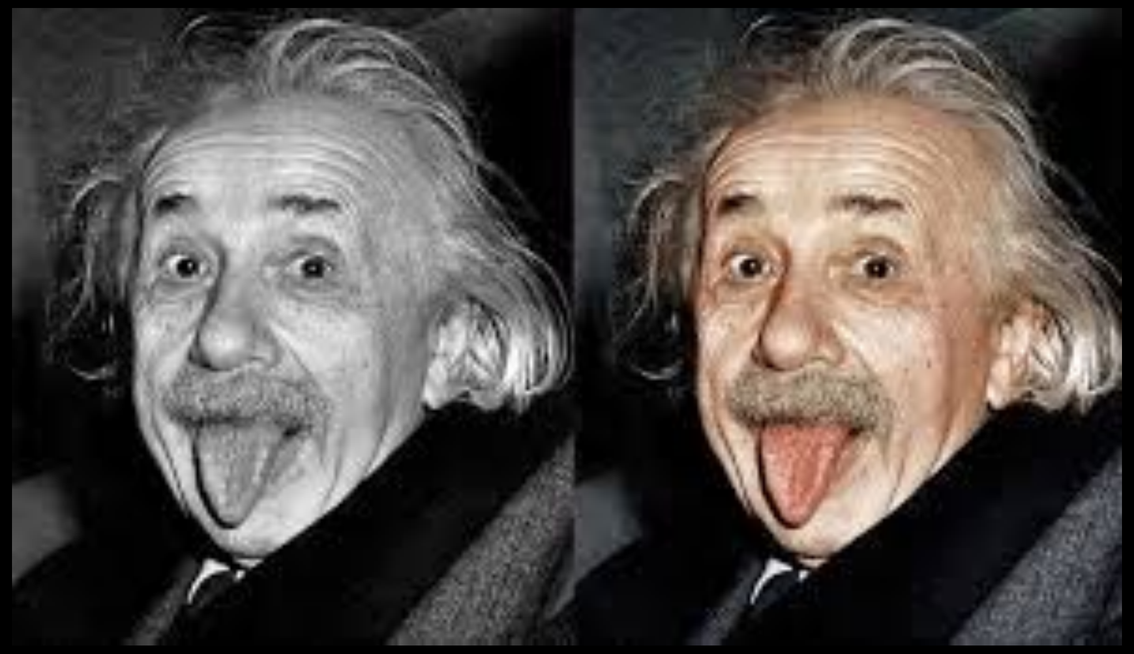
The SR Network



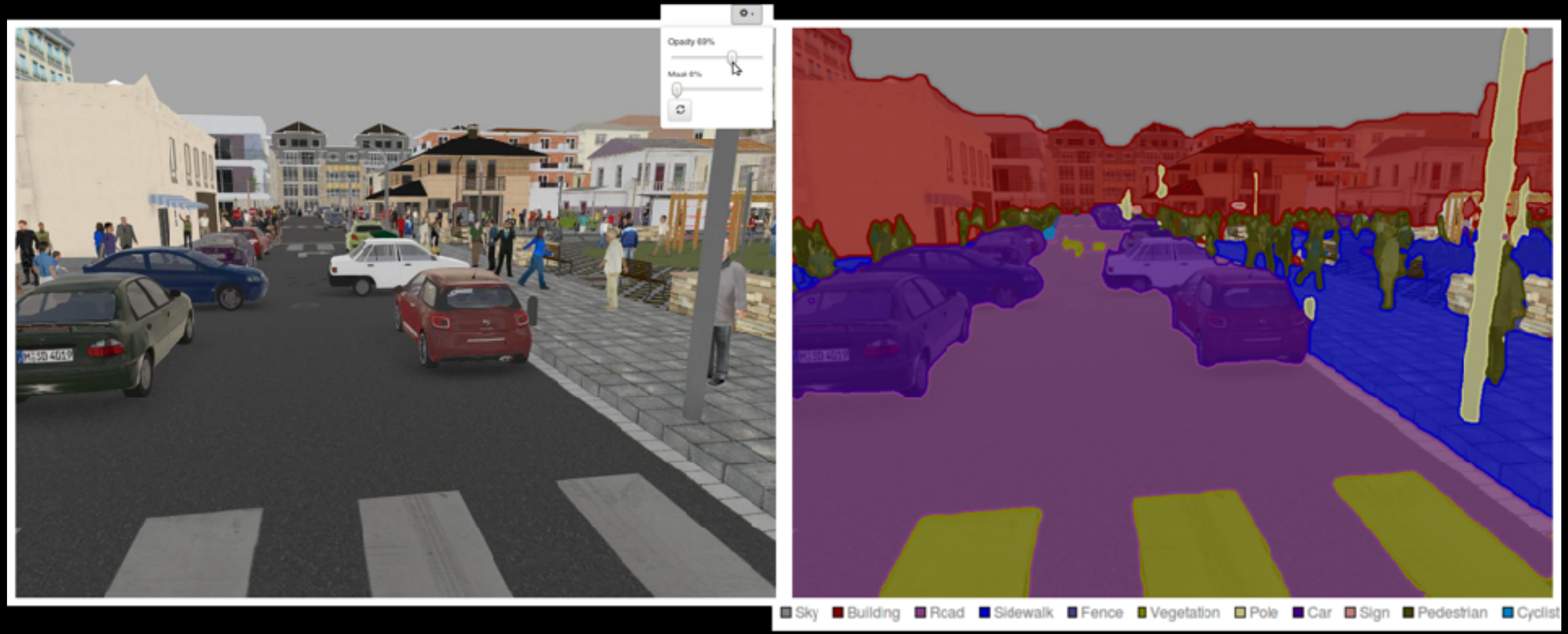
Code Walk through

CNN between anything?

Colorization



Segmentation



CNN between anything?

Others

- Depth Perception
- De-noising
- Visual Filters
- Audio Clarity
- Audio Filters

Summary

- Generative models go far beyond just artist models
- The power of CNN beyond classification
- Perceptual Loss from comparing 2 CNNs
- Generative = image in -> image out
- Try putting a CNN between some data to manipulate it to get what you want

Links

- https://buptldy.github.io/2016/10/29/2016-10-29-deconv/
- https://datascience.stackexchange.com/questions/6107/what-are-deconvolutional-layers
- http://deeplearning.net/software/theano/tutorial/conv_arithmetic.html