Transfer Learning & Neural Style Transfer

School of AI - Singapore

21 Jan 2019

Agenda

- Intro session
- Intro to Transfer Learning
- Style Transfer - Code Deep Dive
- Live Demo
- Q/A
- Hackathon
- Resources to Start
- Feedback, Quick Polls (Upcoming Topics)
- Networking

https://www.theschool.ai/

Singapore School of AI

https://www.facebook.com/groups/2231906417039626/

Our Core Values

1. Embrace the Weird.
2. Inspire and Educate.
3. Data Driven Optimism.
4. Rapid Experimentation.
5. Be Frugal.
6. Choose Love, not Fear.
7. Draw the Owl.

Types of Events

- Class or Meetup
- Study Group
- Hands-on Workshop (Coding)
- Research Papers Reviews
- Webinars
- Hackathons

CORE TEAM

Sunil Thakur (Dean)
Email: sunilforu@gmail.com

Karthik Naga (Dean, USA)
Email: knaga82@gmail.com

Lokesh Korapati

Bala Prasanna (Dean)
Email: balaprasannav2009@gmail.com

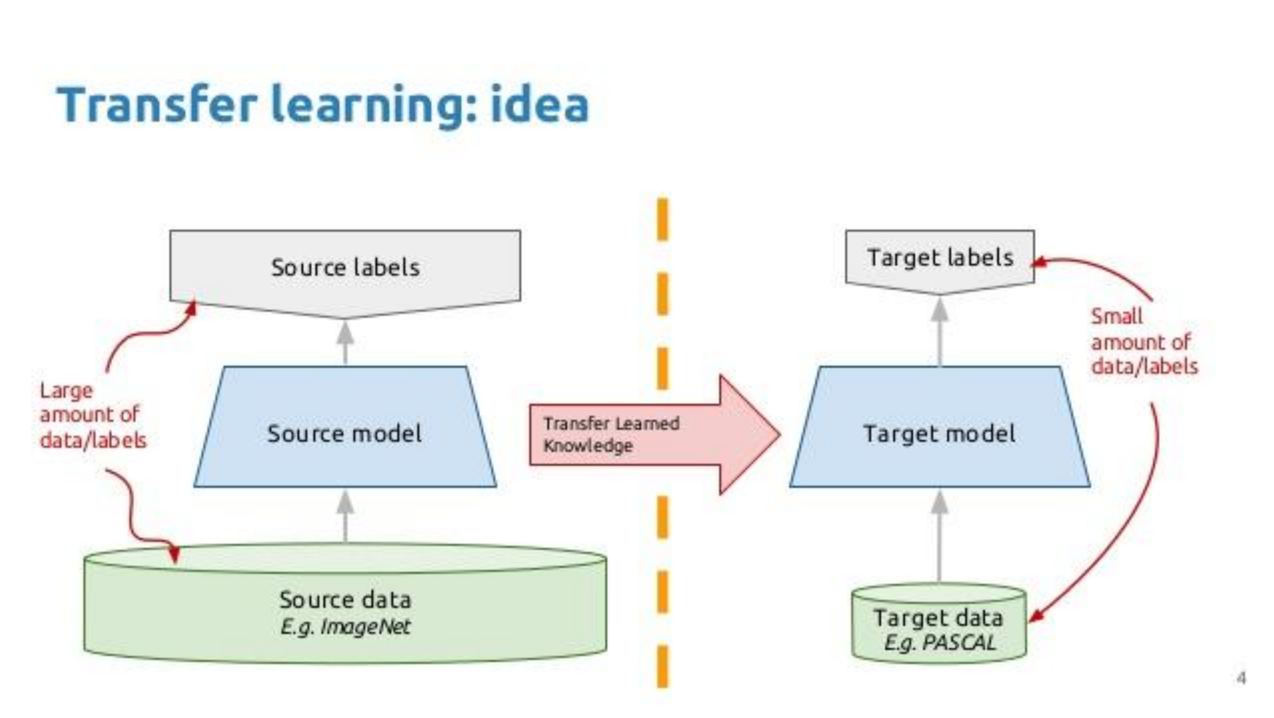
Sami Jawhar

Davis

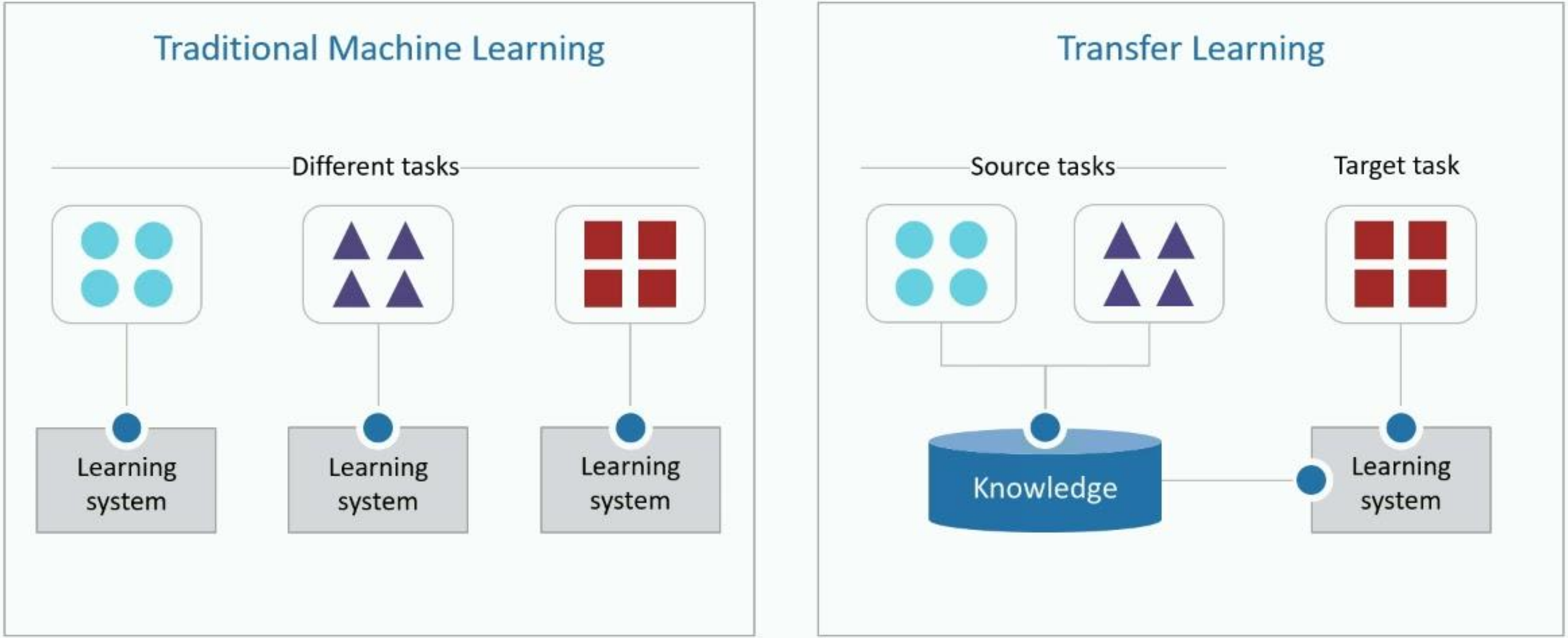
Introduction to

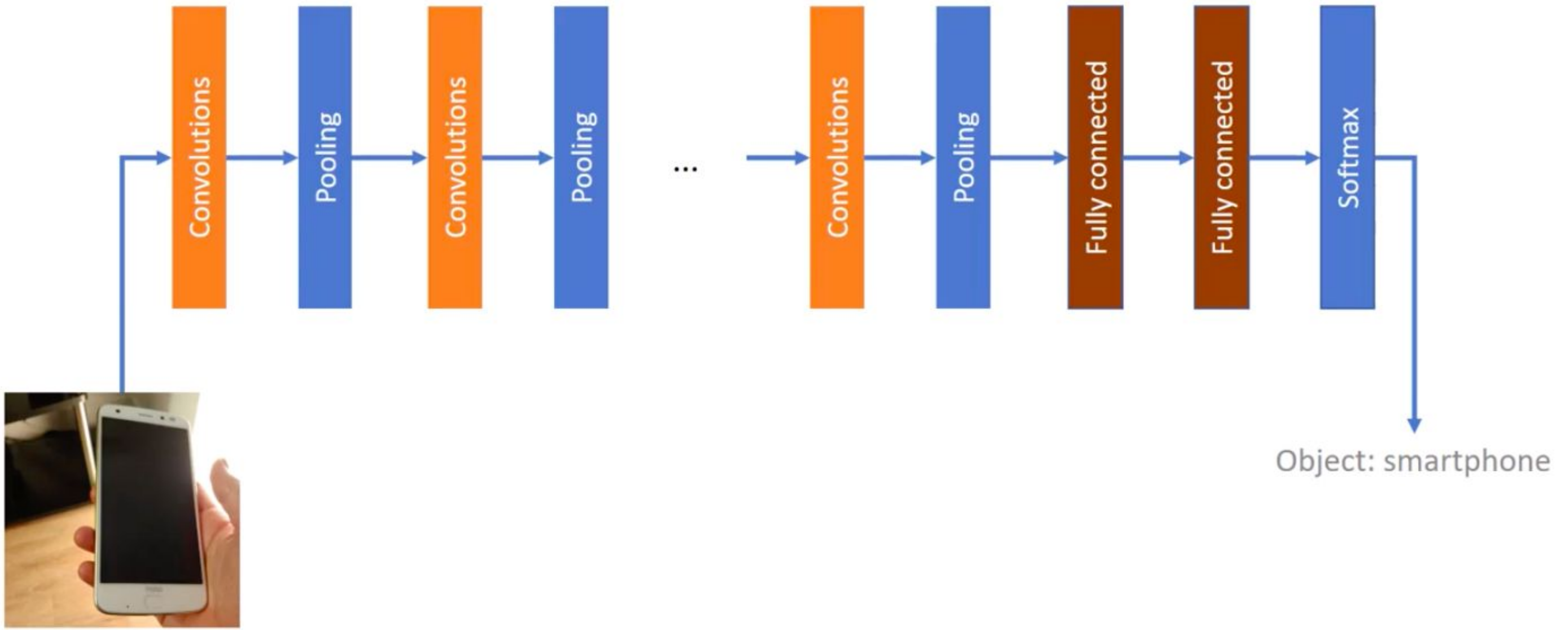
Transfer learning

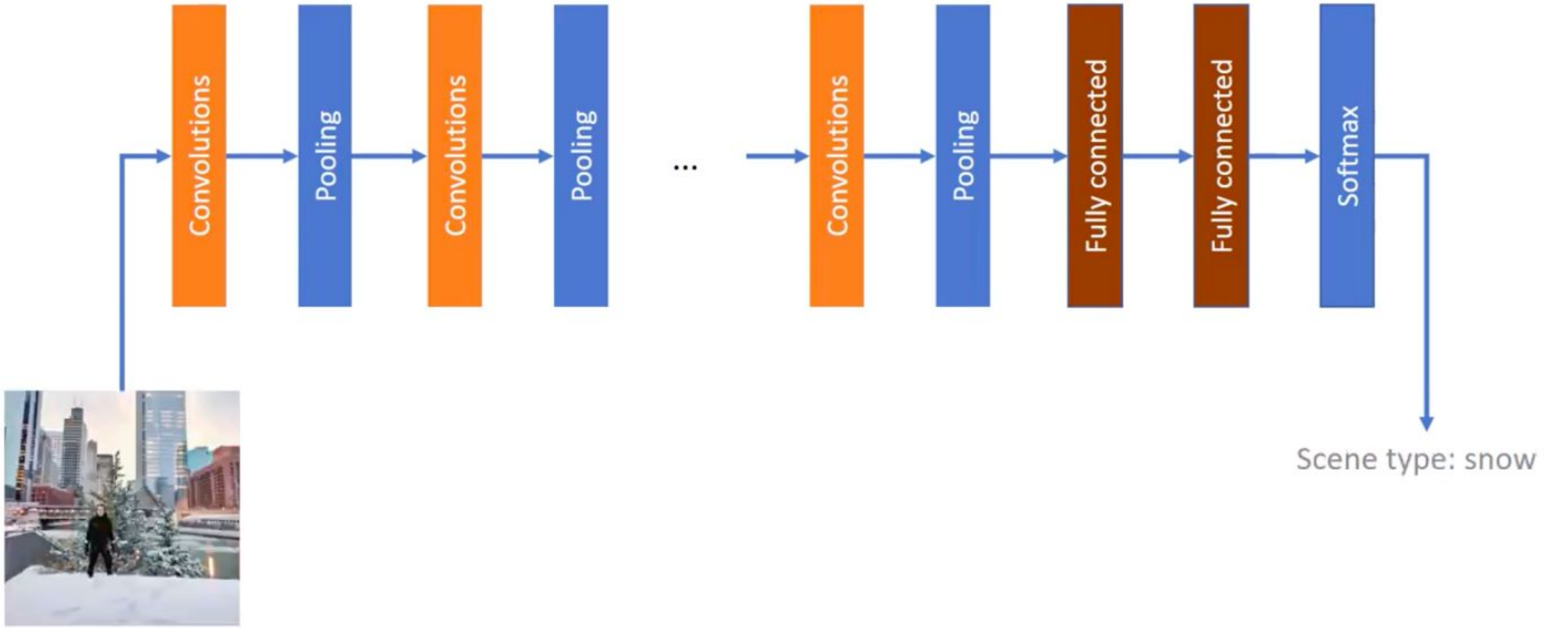
Intro to Transfer Learning



Traditional versus Transfer learning




Convolutions
Pooling
Convolutions
Pooling
...
Convolutions
Pooling
Fully connected
Fully connected
Softmax
Object: smartphone



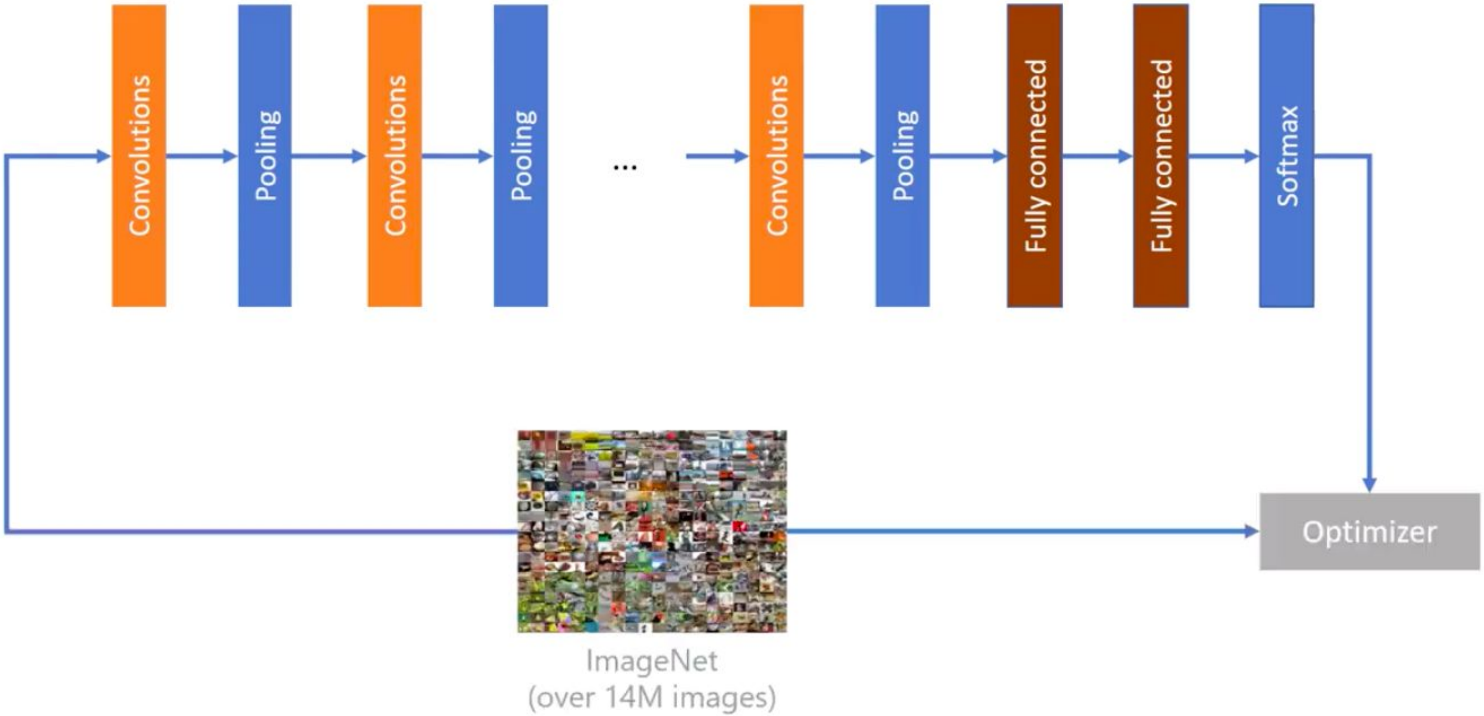
Convolutions
Pooling
Convolutions
Pooling
...
Convolutions
Pooling
Fully connected
Fully connected
Softmax
Scene type: snow

Transfer Learning scenarios

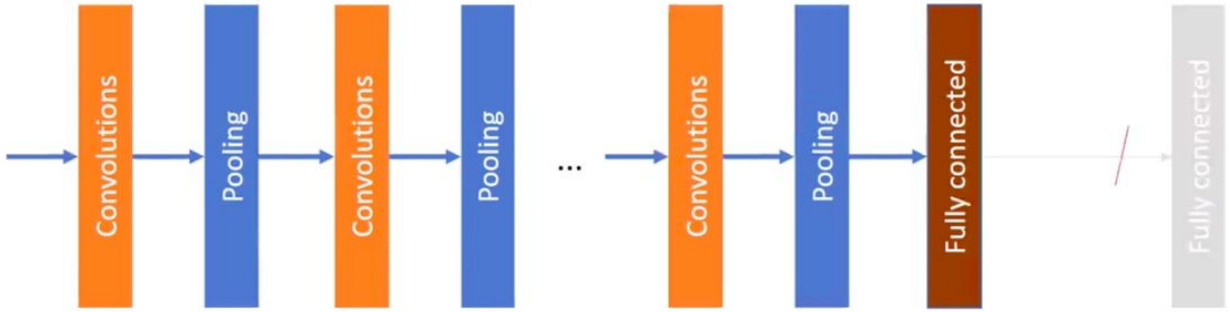
- ConvNet as fixed feature extractor
- Fine-tuning the ConvNet
- Pretrained models

Scenario 1 ⇒ ConvNet as fixed feature extractor

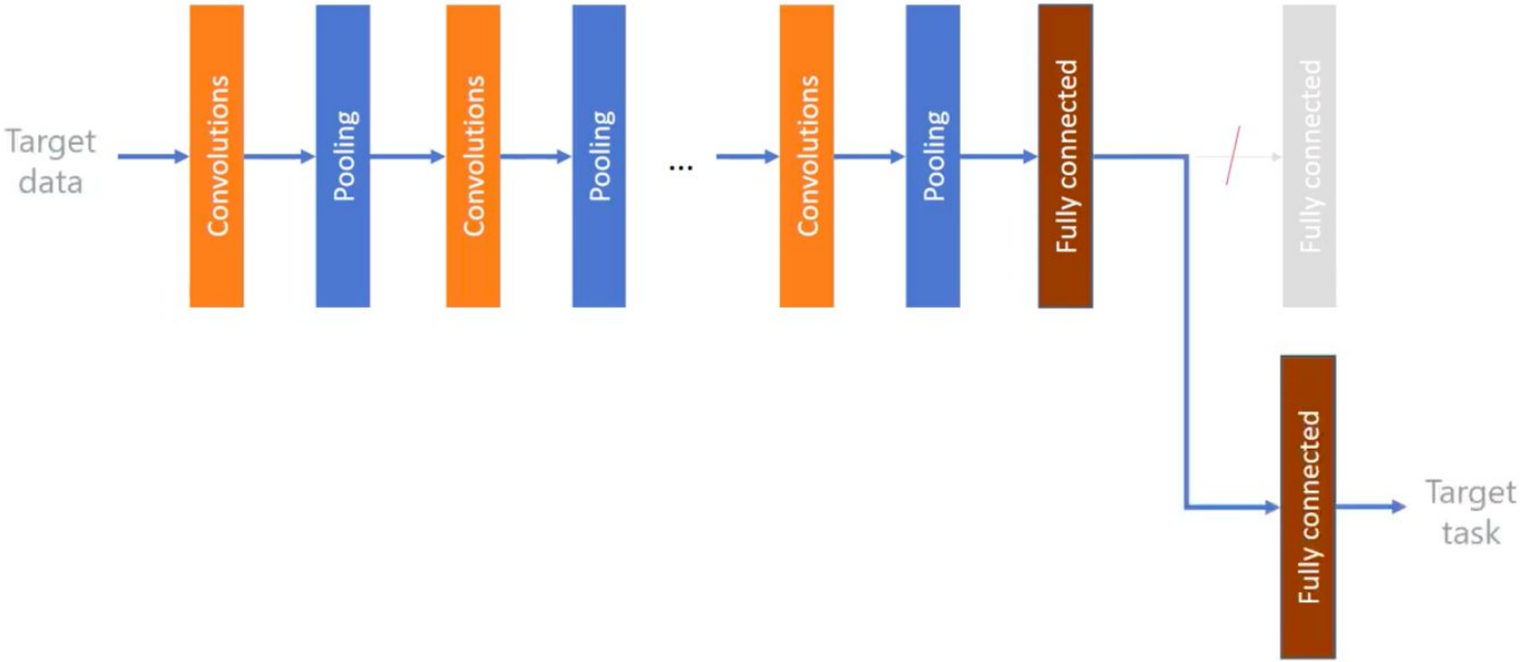
1. Take a ConvNet Pre-trained on ImageNet
2. Remove the last fully-connected layer
    a. (this layer's outputs are the 1000 class scores for a different task like ImageNet)
3. Treat the rest of the ConvNet as a fixed feature extractor for the new dataset.
4. Once you extract the feature vector / CNN codes for input images, train a linear classifier (e.g. Linear SVM or Softmax classifier) for the new dataset.


Convolutions
Pooling
Convolutions
Pooling
...
Convolutions
Pooling
Fully connected
Fully connected
Softmax
Optimizer
ImageNet
(over 14M images)

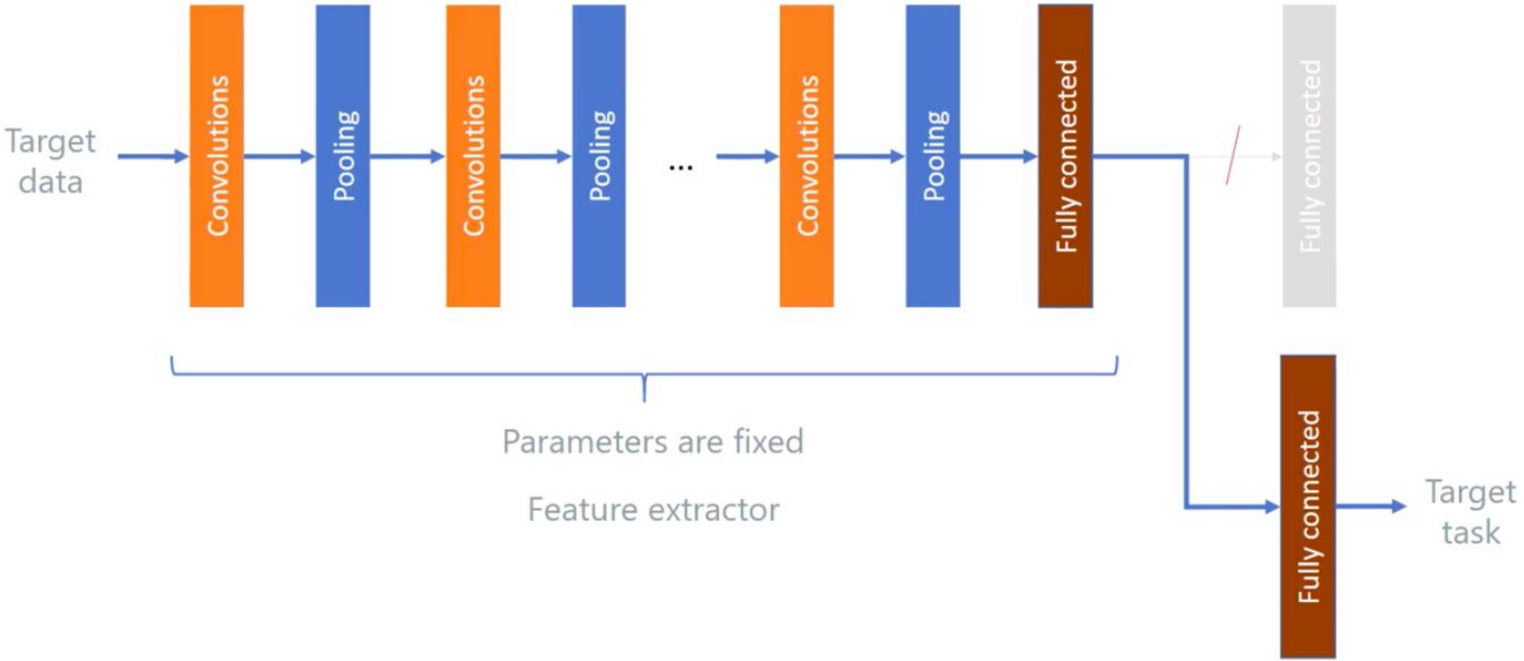
Scenario I: train the top layer only



Scenario I: train the top layer only



Scenario I: train the top layer only



Scenario 2 ⇒ Fine-tuning the ConvNet

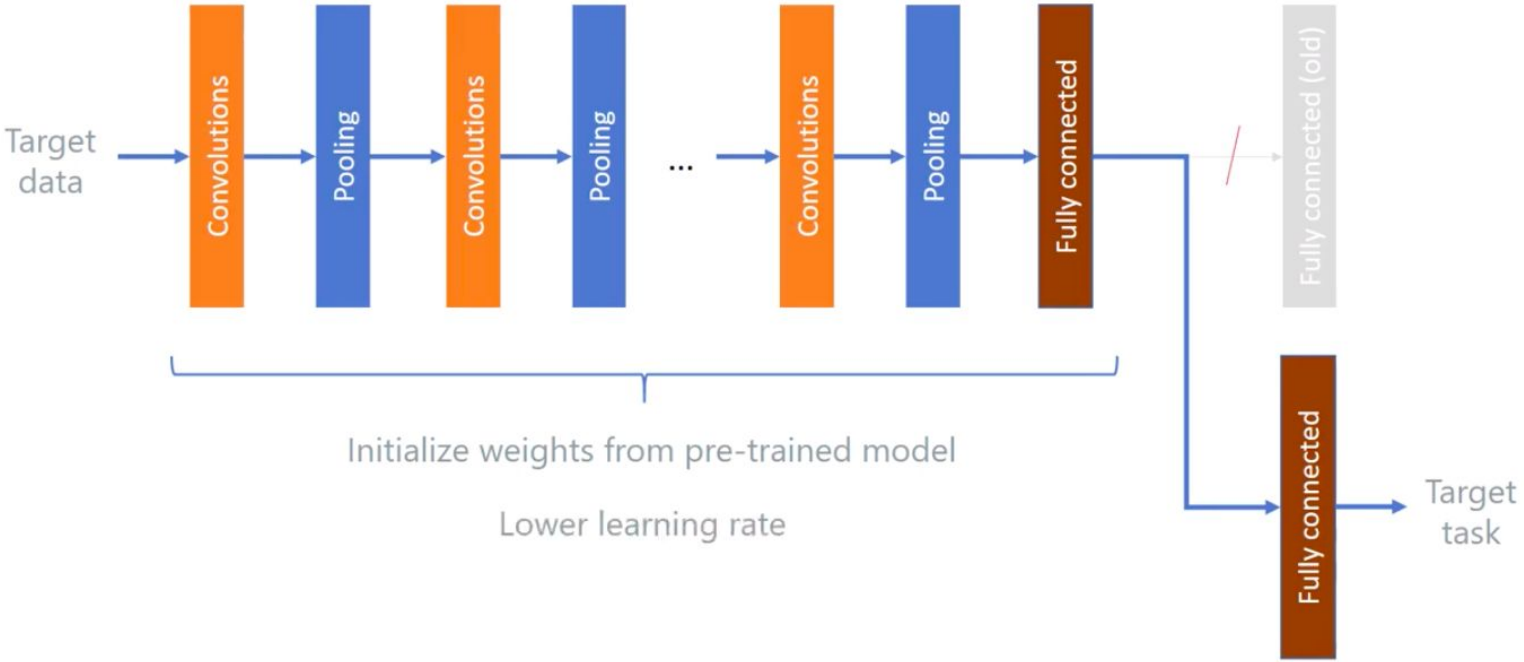
1. Not only replace and re-train the classifier on top of the ConvNet on the new dataset, but to also fine-tune the weights.
2. But How ?
   a. Using backpropagation
3. Fine-Tune
   a. All the layers of ConvNet
   b. Keep some of earlier layers fixed and only fine-tune later layers

Fine-Tune - All the layers

1. Because the weights are better than randomly initialized
2. You can try differential learning rates

```
>>>
>>> lr = 1e-3
>>>
>>> lr
0.001
>>>
>>> diff_lrs = [lr/6 , lr/3, lr]
>>>
>>> diff_lrs
[0.00016666666666666666, 0.0003333333333333333, 0.001]
>>>
```

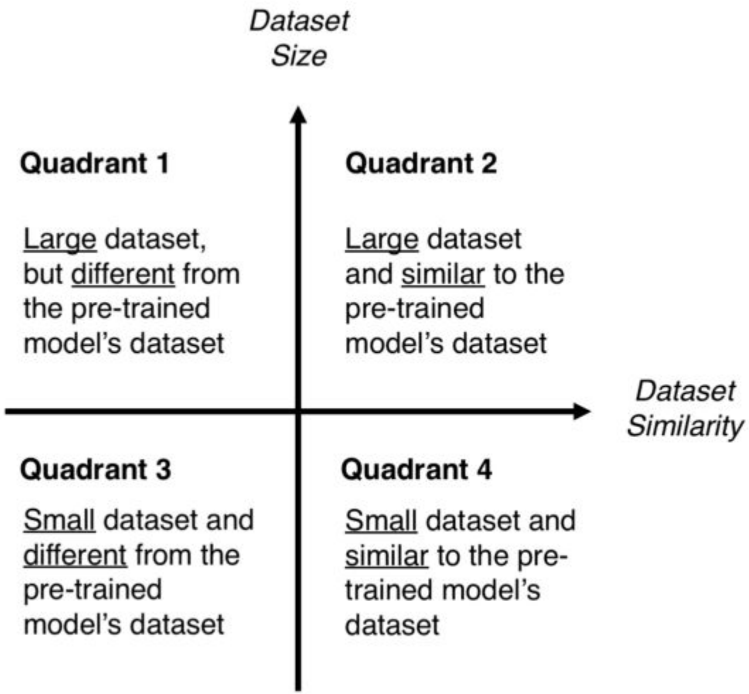
Scenario II: fine-tune all layers

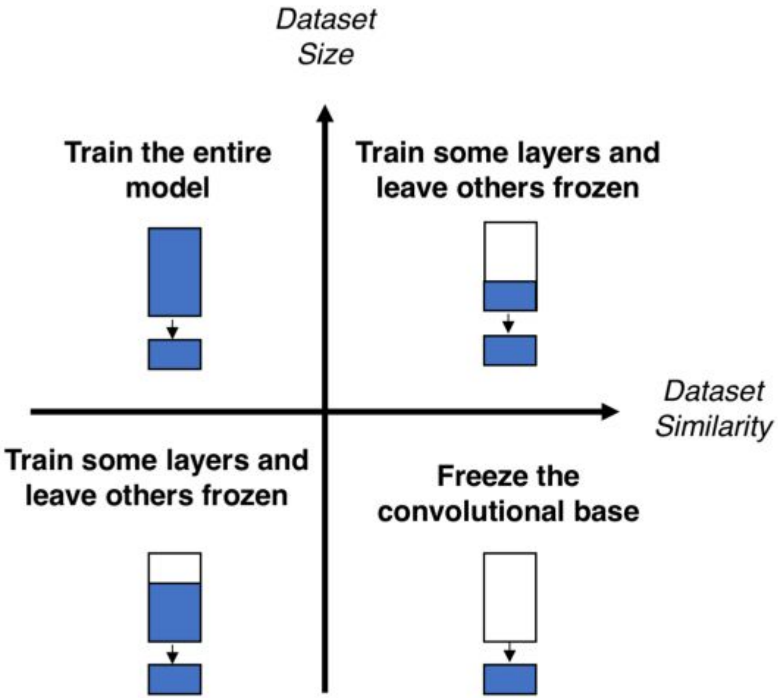


Keep earlier layers fixed and only fine-tune later layers

1. Earlier features of ConvNet contain more generic features
    a. **[f'{_}detectors' for _ in [edge, color]]**
2. Later layers of the ConvNet becomes progressively more specific to the details of the classes contained in the original dataset

Scenario : Recap

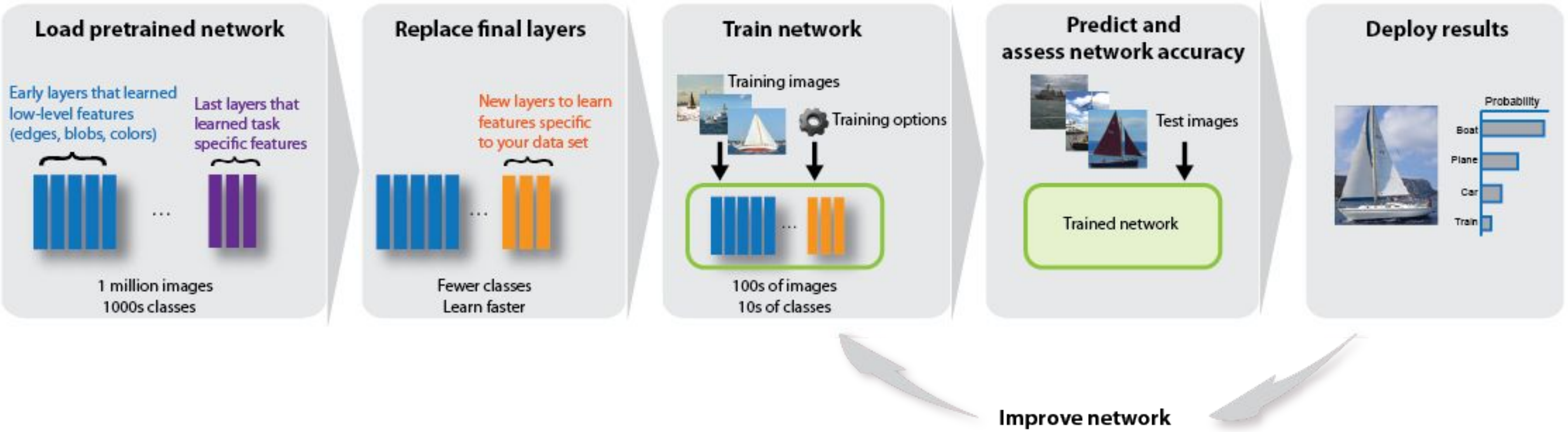




Pre Trained Models

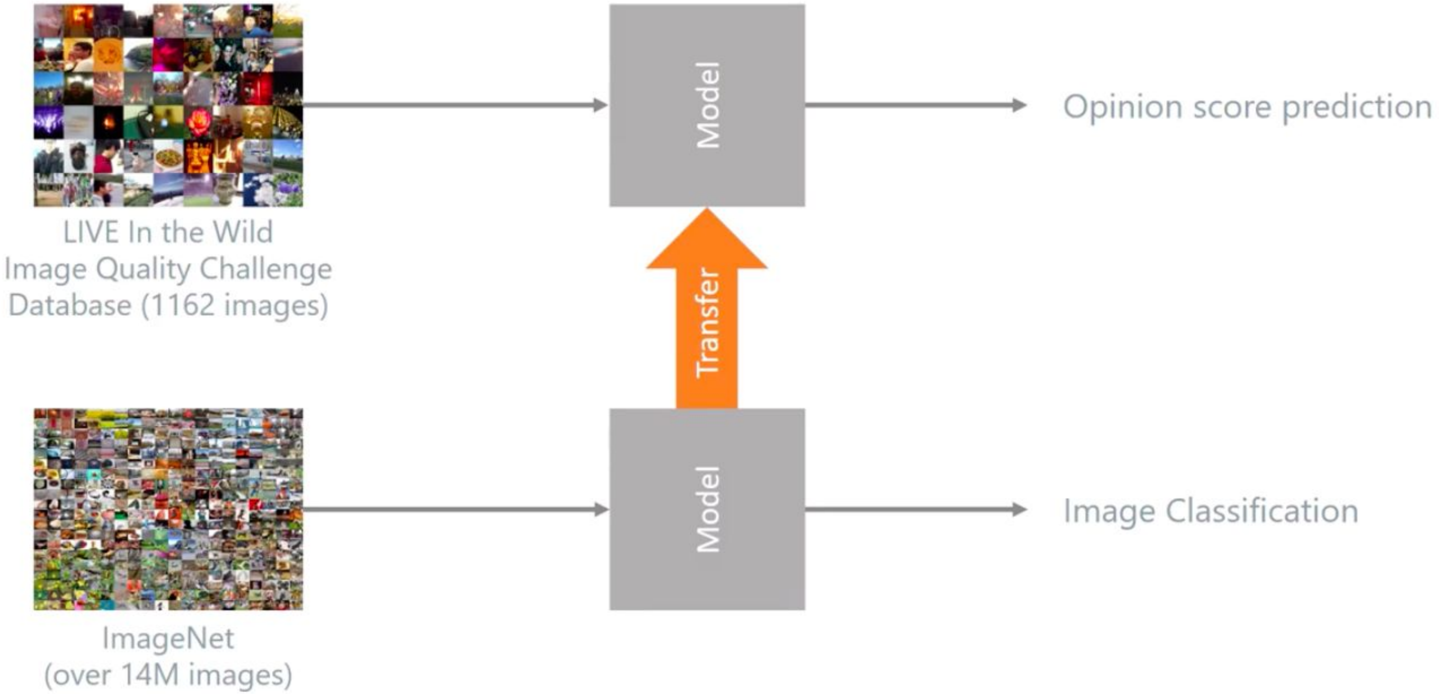
1. modern ConvNets take 2-3 weeks to train across multiple GPUs on ImageNet
2. common to see people release their final ConvNet checkpoints
3. For eg: the Caffe library has a *Model Zoo* where people share their network weights.

Reuse Pretrained Network

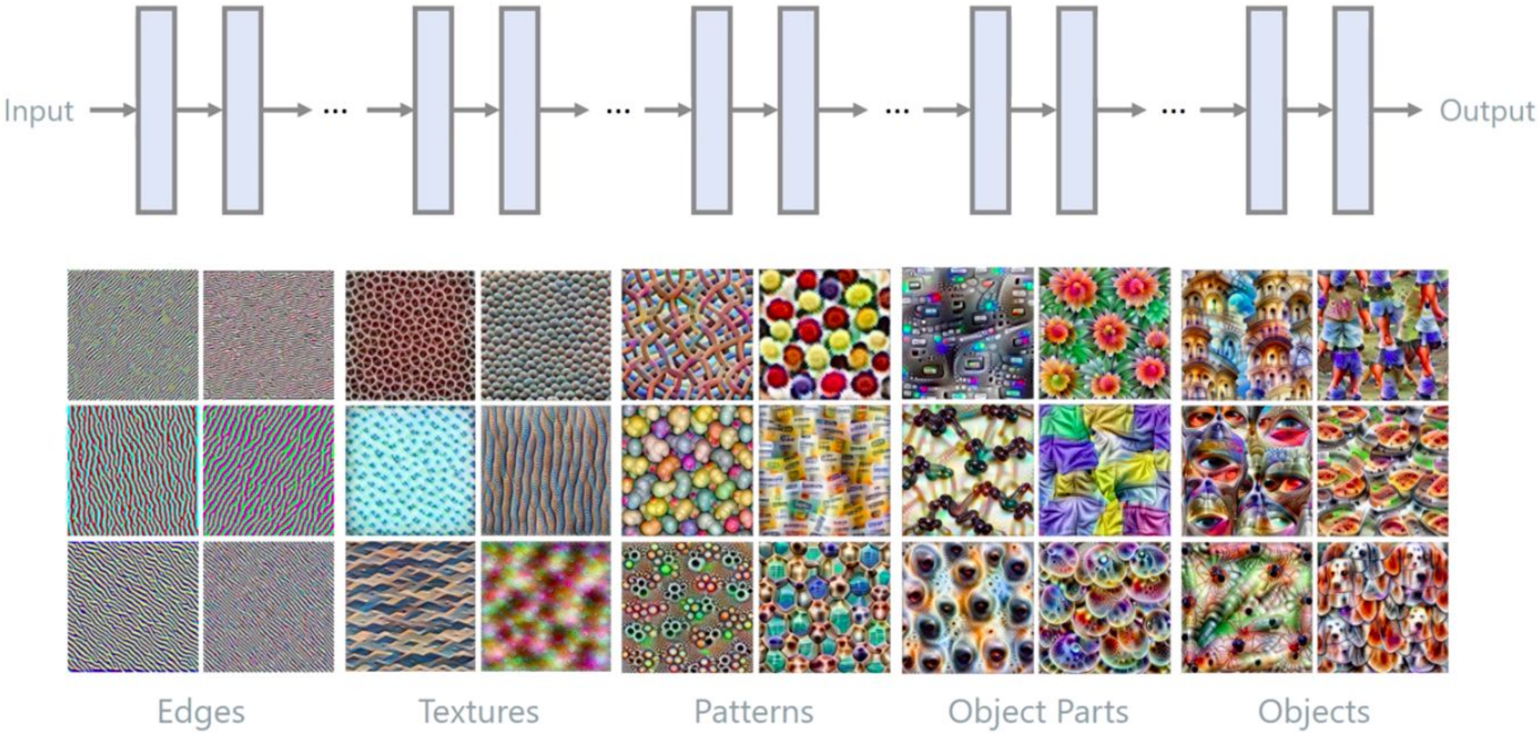


Quiz:

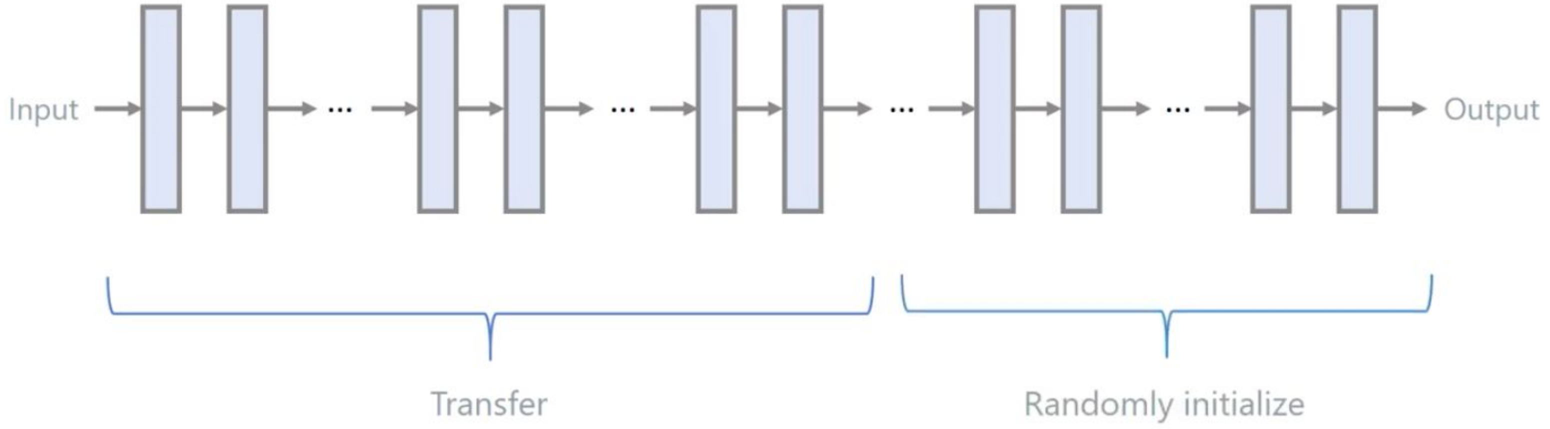
What are we transferring ??



LIVE In the Wild
Image Quality Challenge
Database (1162 images)
ImageNet
(over 14M images)
Model
Transfer
Model
Opinion score prediction
Image Classification



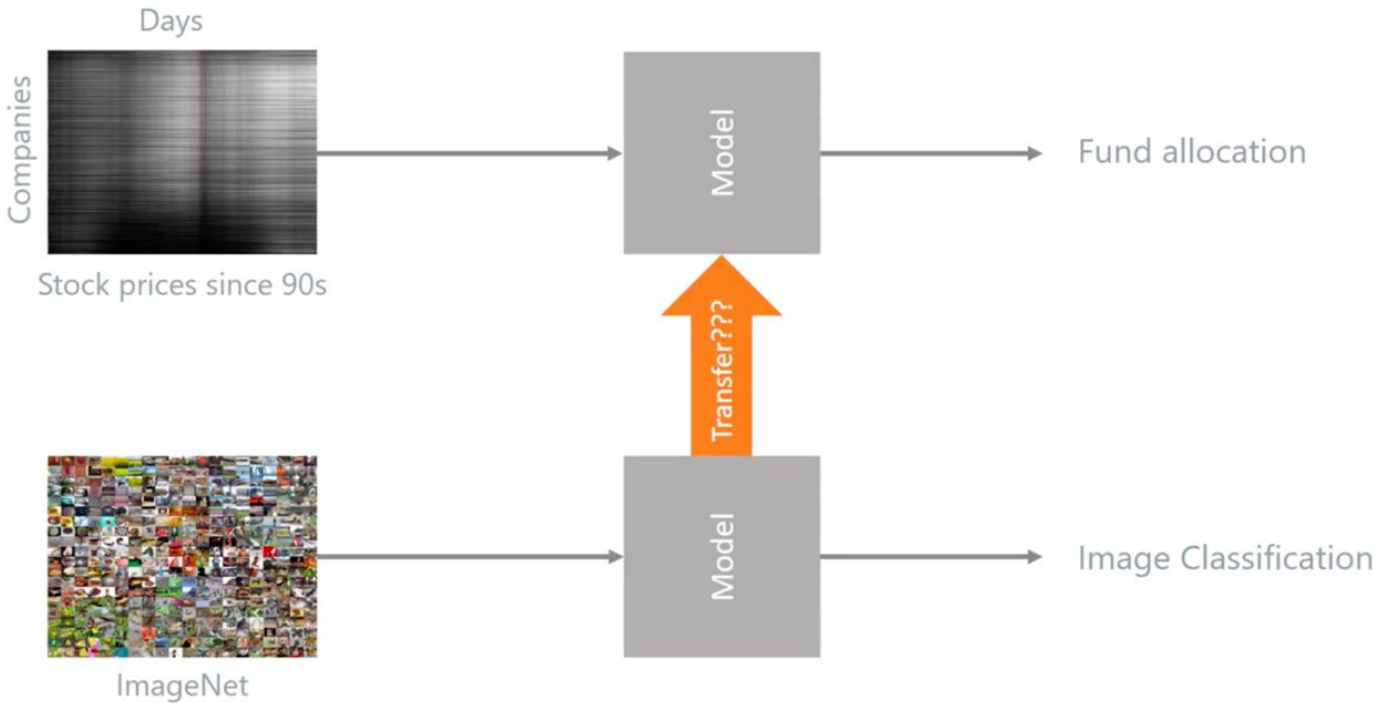
Visualization credit: https://distill.pub/2017/feature-visualization/


Input
Output
Transfer
Randomly initialize

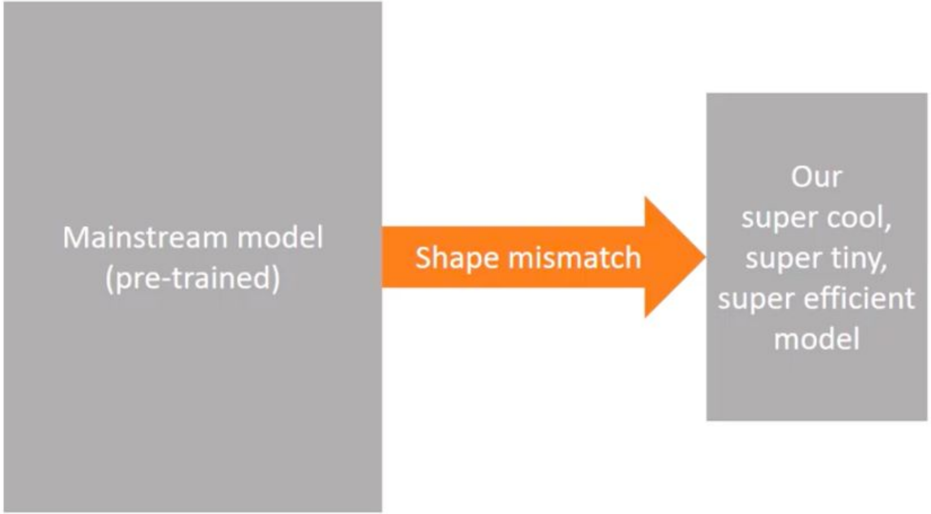
Limitations

- Transfer Learning Makes Sense If
  - Task A and B have the same input x
  - You have a lot more data for Task A than Task B

Limitations

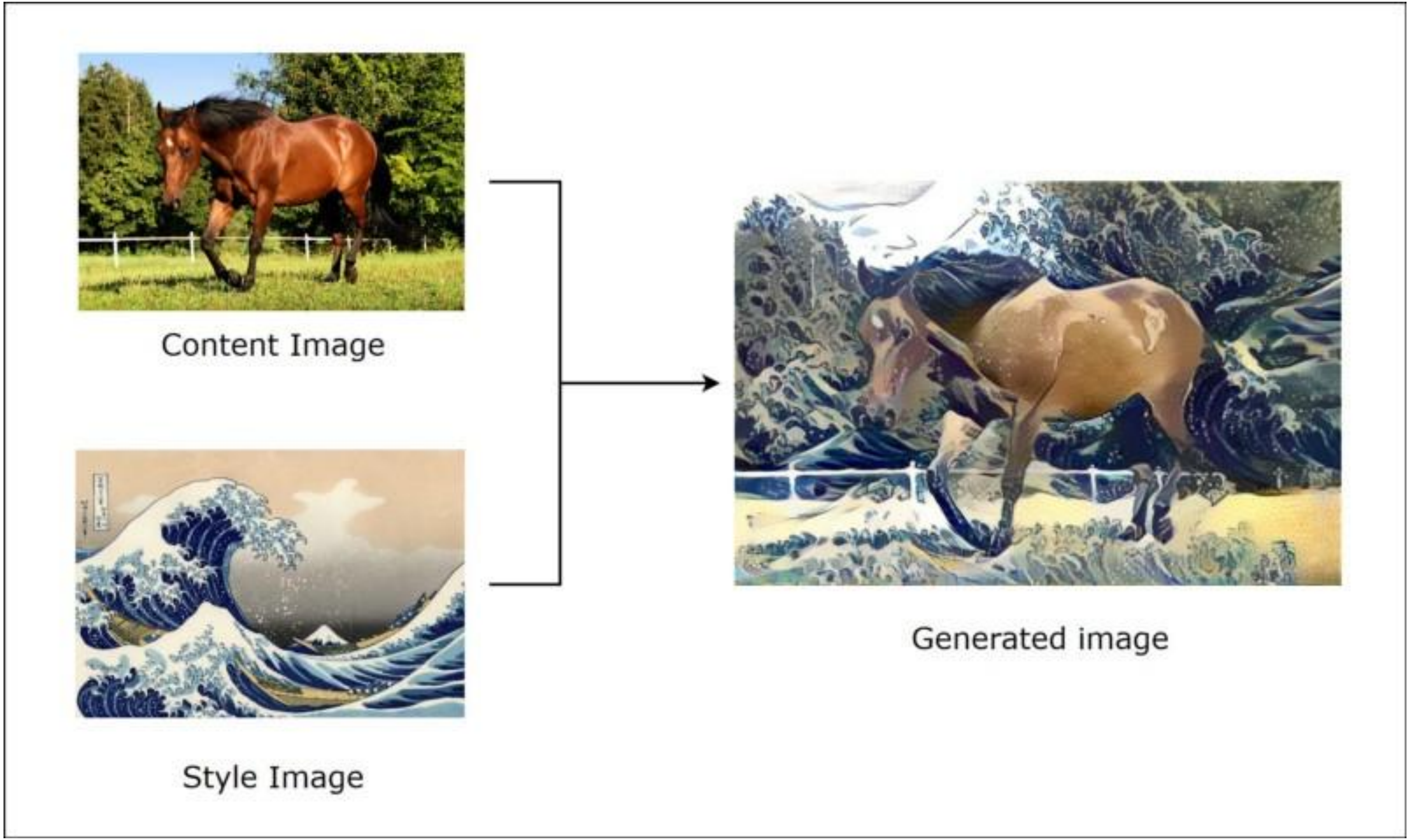


Limitations



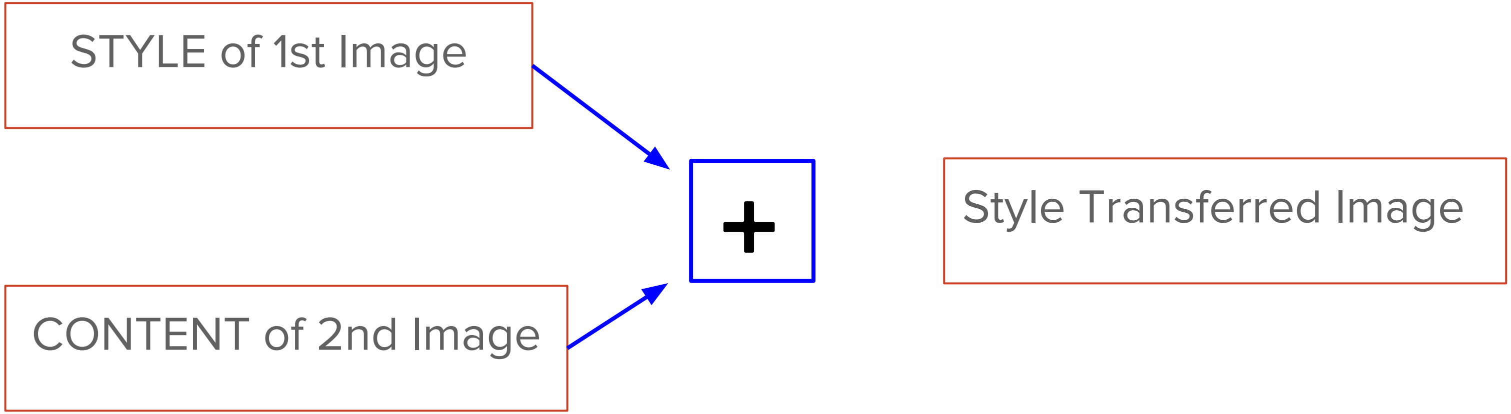
Application:

Style Transfer - Code Deep Dive


Content Image
Style Image
Generated image

Style Transfer

- Use Pre-Trained ConvNet to Extract feature representations from an image

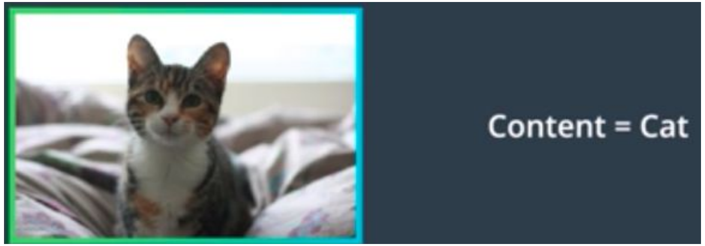


STYLE of 1st Image

$+$

Style Transferred Image

CONTENT of 2nd Image



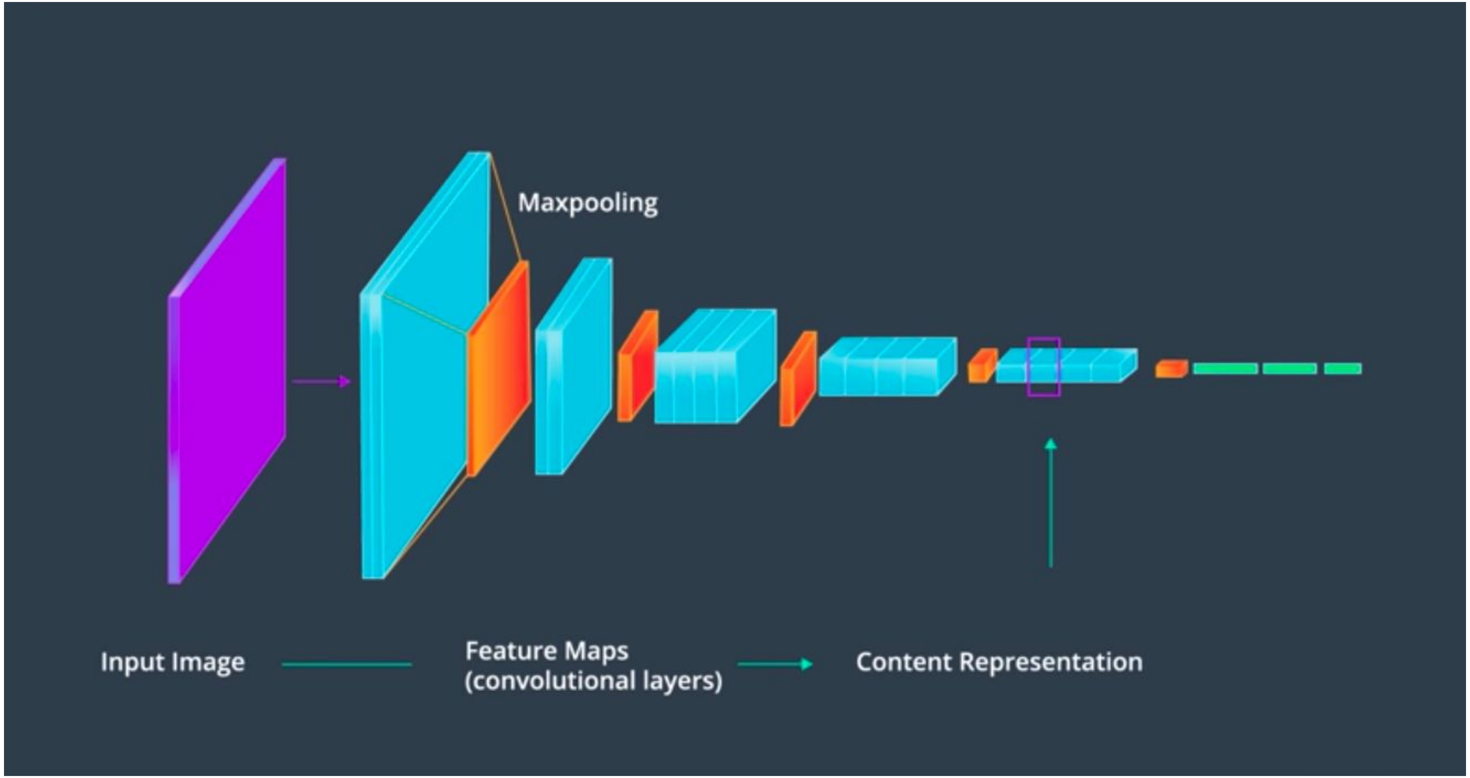
But How to extract Style & Contents from Images

1. Take a pre-trained model (say, vgg19)
2. Remove the head (last layer - classifier)
3. Do a forward Propagation
   a. Get feature maps for all different layers

STYLE - Earlier Layers

CONTENT - Later layers

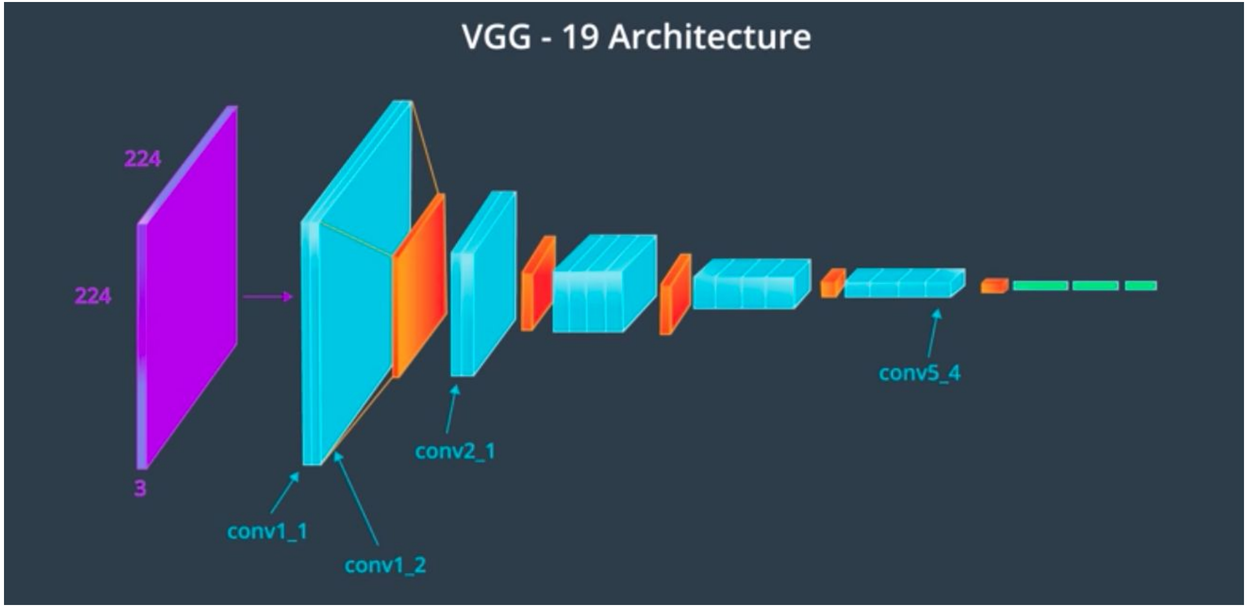
How to Extract Content Representation



Cont..

Lets look at VGG -19 Arch

[“stack”.join (conv1_1, conv1_2)]

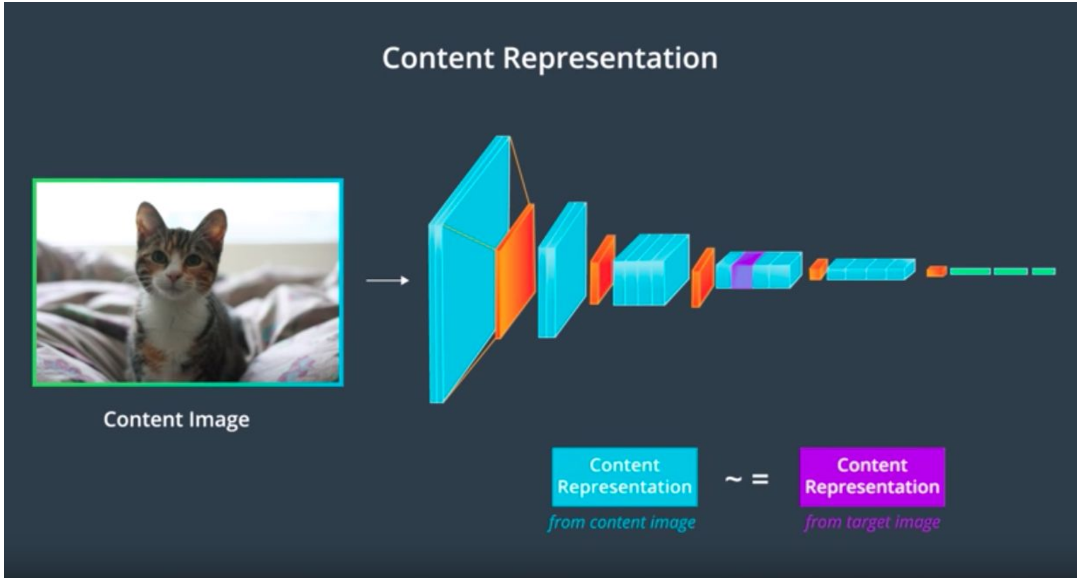


Conv5_4

Refer to Pytorch Code, You will see this in Content Loss

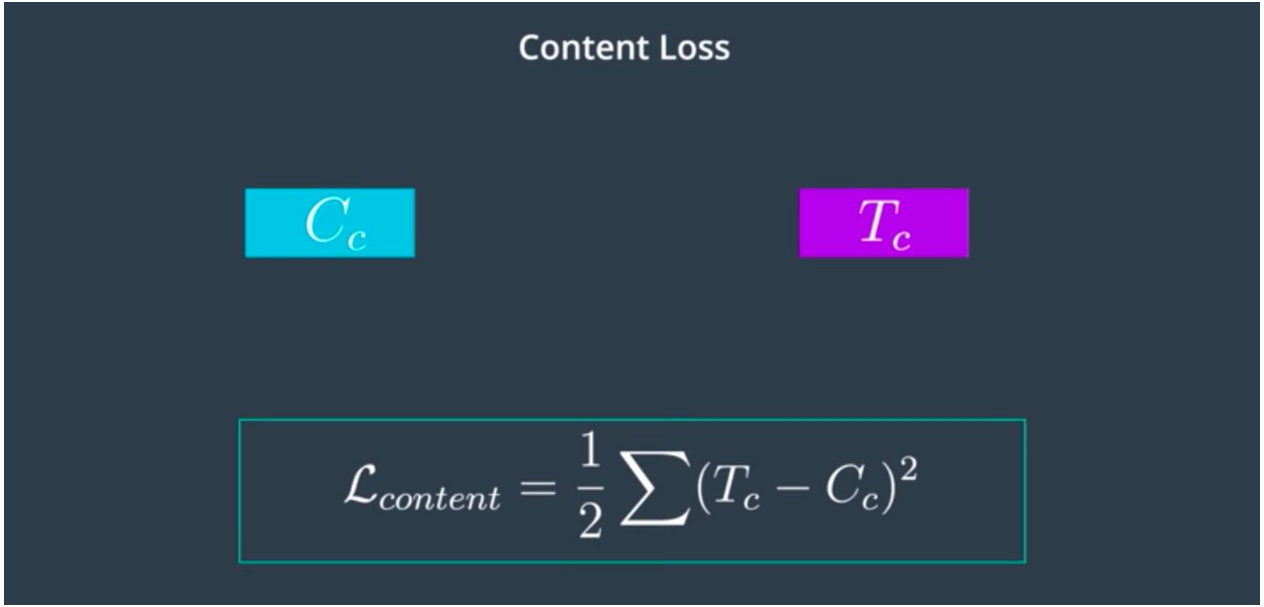
Content Loss

Intuition



Content Loss

Let’s look at content loss



Use backpropagation to optimize this loss.

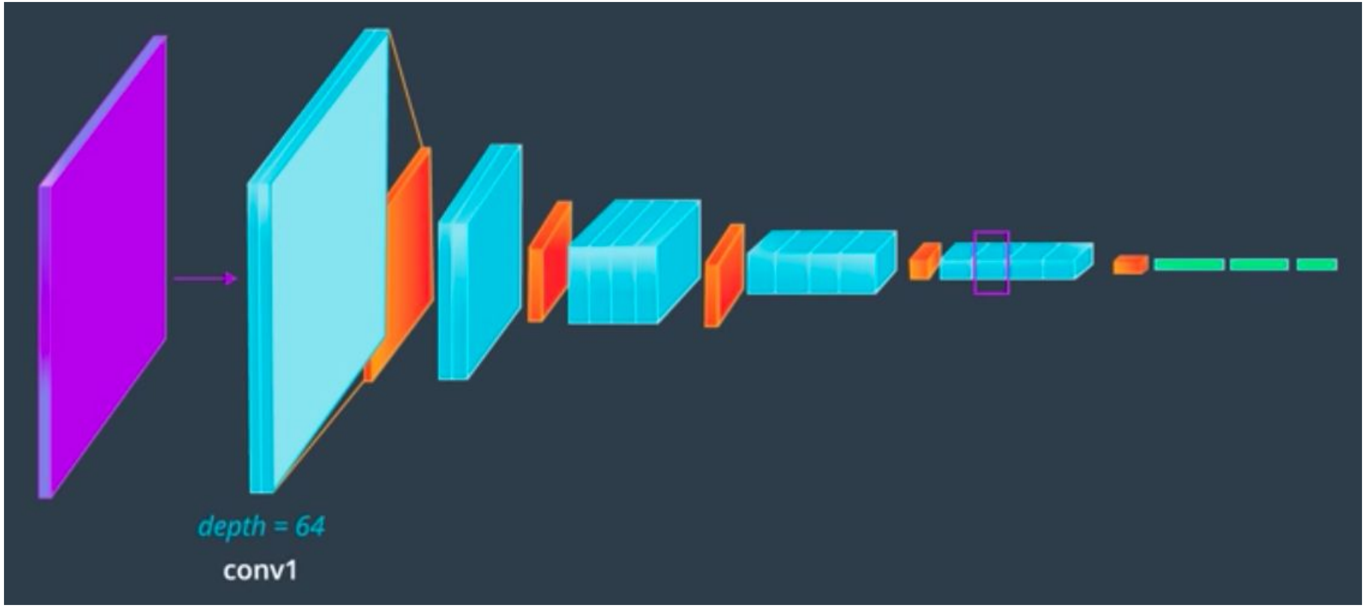
We will look at Style Loss later

How to Extract Style

1. A feature space should capture Texture & Color information need to be used
2. This space looks at spatial correlation within layer of network

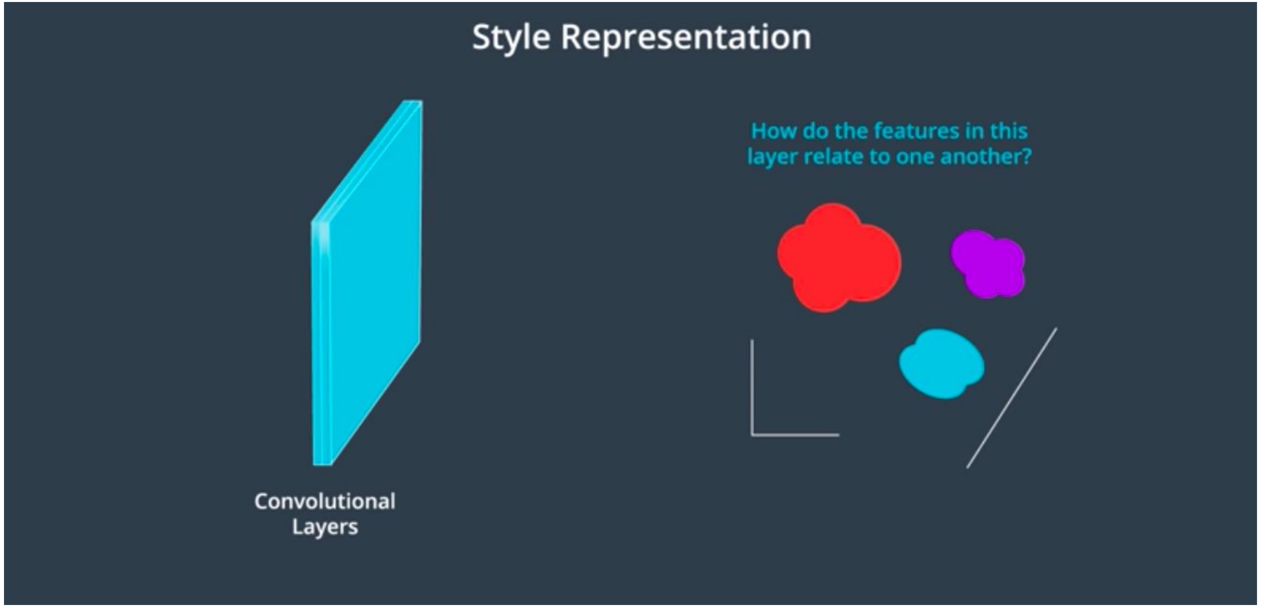
Cont..

For ex, we can look at feature maps at the first layer



Cont..

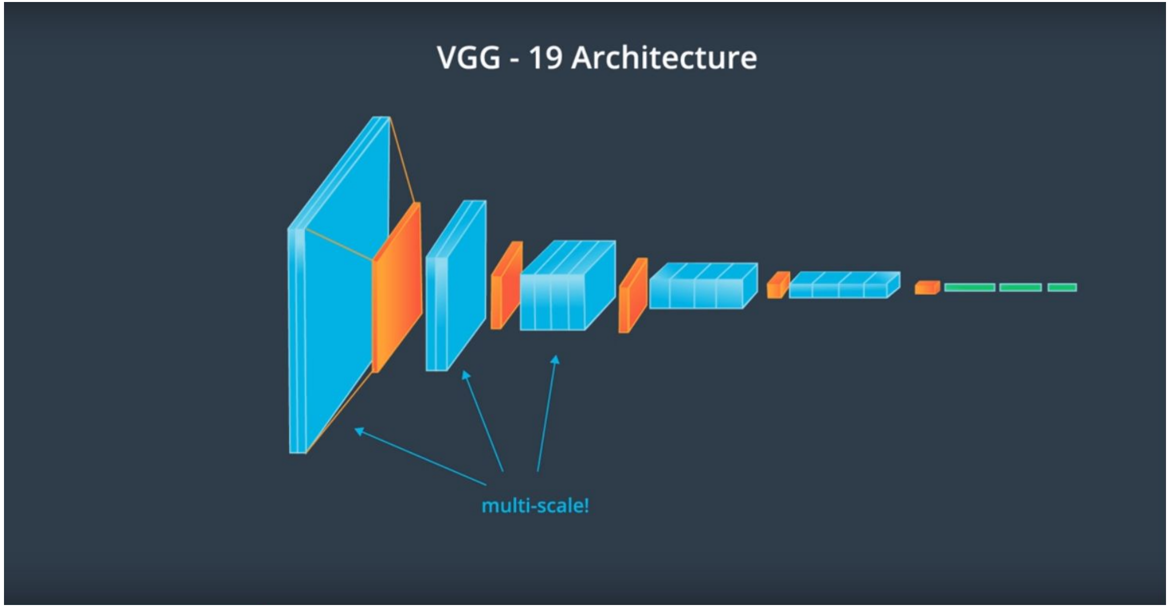
Correlation between feature maps at the first layer of depth k



- Is certain **color detected** in one map, **similar** to another map ?
- See which colors & shapes in set of feature map are **related** and which are **not**

Cont..

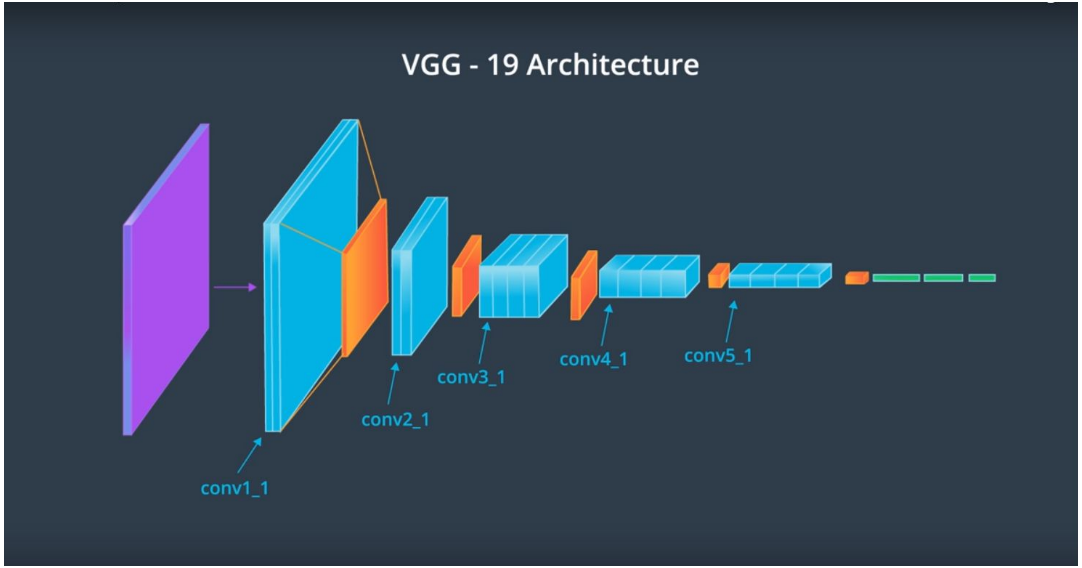
Multiscale - Style Representation



- First conv layer in earlier blocks of VGG

Cont..

Con1_1 , Conv2_1, Conv3_1, Conv4_1, Conv5_1



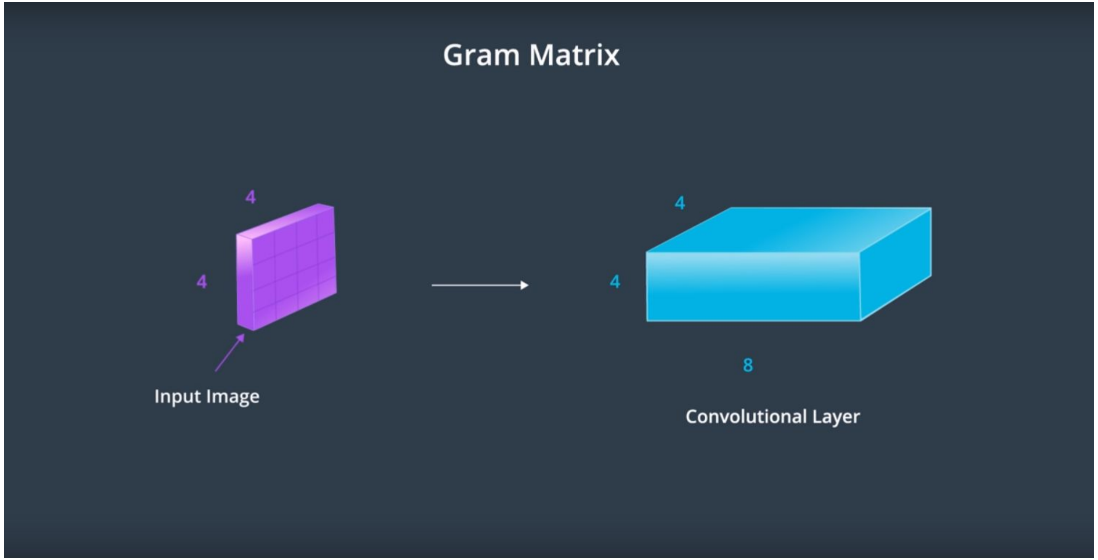
Con1_1
Conv2_1
Conv3_1
Conv4_1
Conv5_1

Refer to Pytorch Code, You will see this in Style Loss

Gram Matrix

Cont..

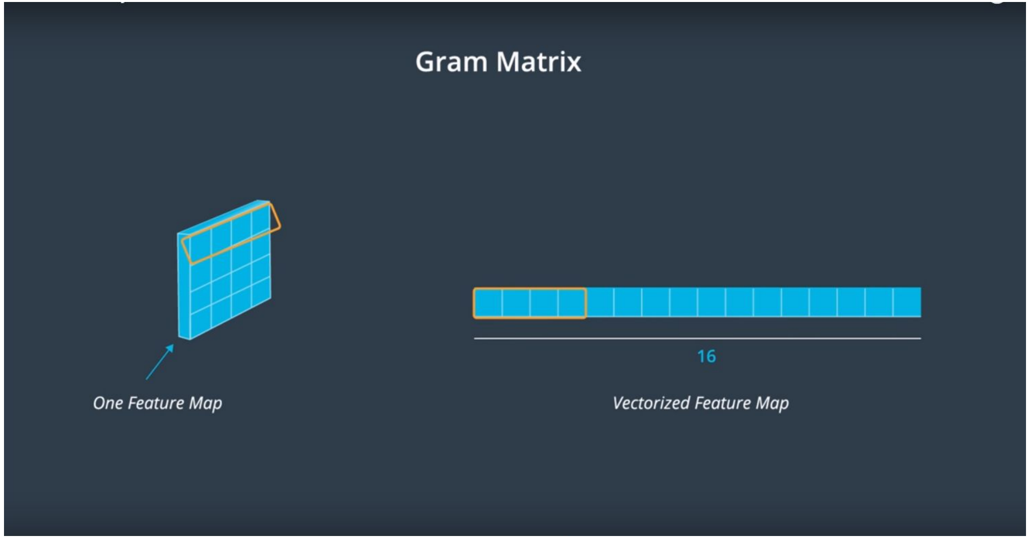
An input (4,4) to Conv Layer of Depth 8, (4,4,8)



- SAME Convolution padding to retain size
- 8 Feature map
  - (Depth)
-

Cont..

One Feature Map -> Flatten to a row vector



fm.shape

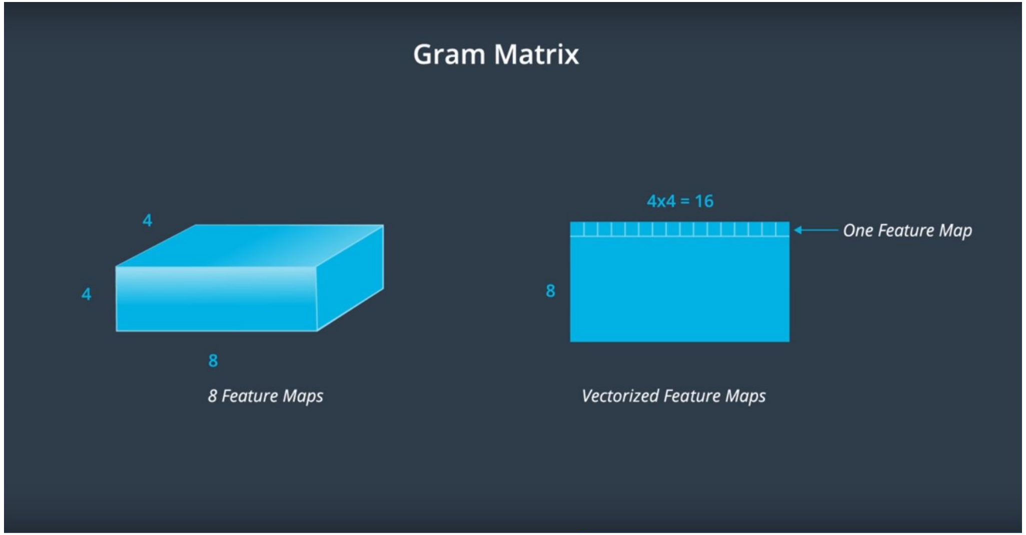
(4,4)

fm.view(1, -1)

(1,16)

Cont..

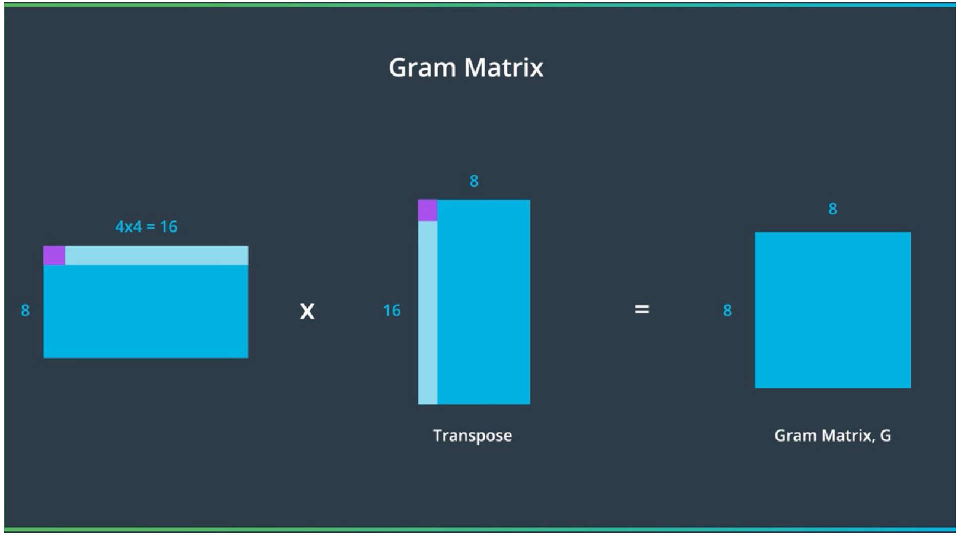
Conv layer (8, 4,4) -> to Vectoried Feature Maps 8, (4*4)



- cv.shape
  - (8,4,4)
- cv.view(8, (4*4))
  - (8, 16)

Correlations of Each Layer is given by Gram Matrix

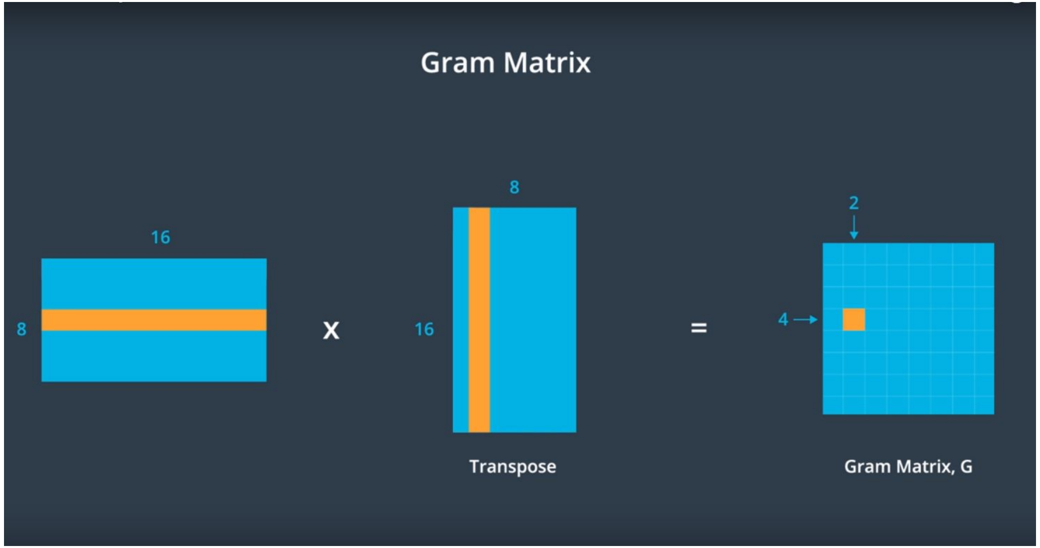
Gram Matrix is just torch.mm(fm_vectorized , fm_vectorized.T())



- Correlation across feature maps in individual layers of a VGG Net
- Find Similarities across features in a each layer
- How similar features in a single layer are ?

Cont..

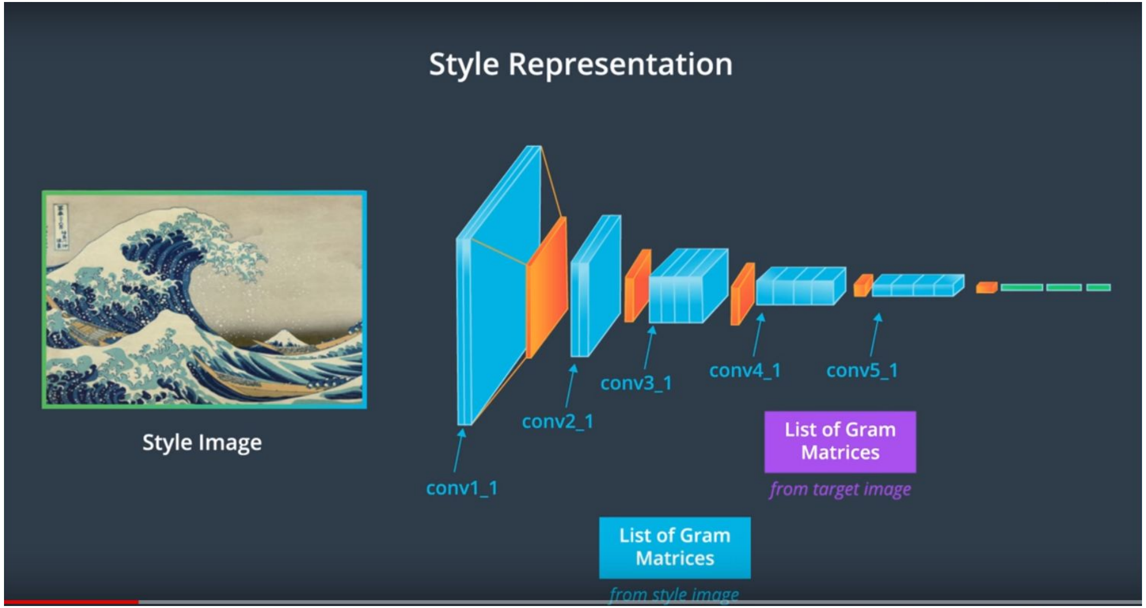
How similar each feature maps are in a particular layer



- Finally, this 8,8 Gram matrix indicate similarities between feature maps
- Indicate the similarity between 4th and 2nd feature map in a layer

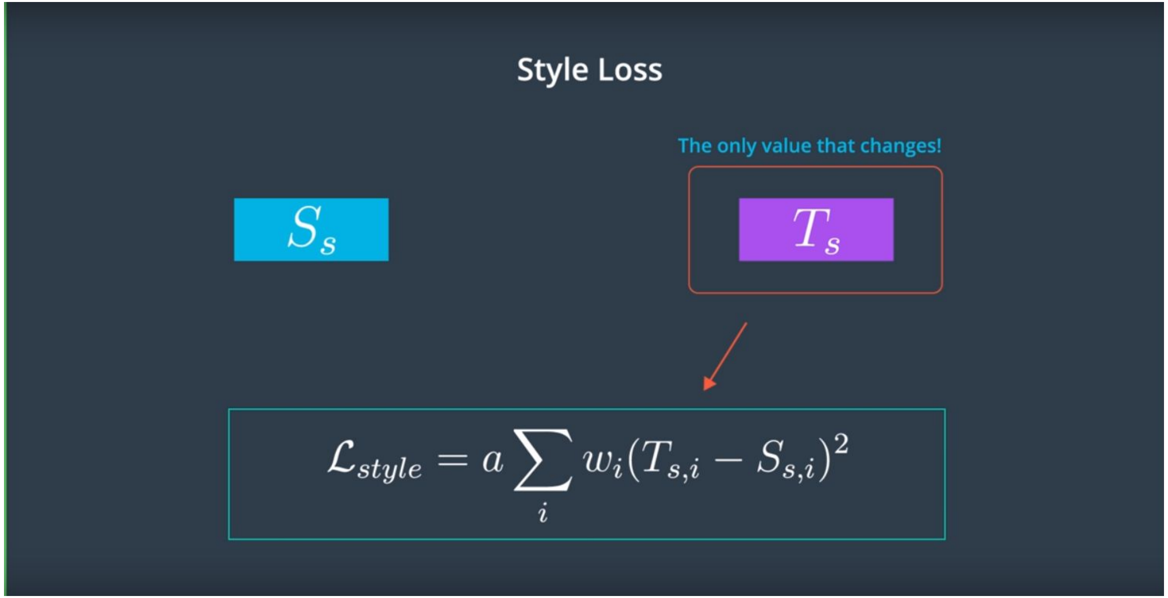
Style Loss

To compute the style loss, compute the list of gram matrices



Style Loss

Style Loss can be weighted using a scale w for each layers in this



Con1_1
Conv2_1
Conv3_1
Conv4_1
Conv5_1

W = [
1.0,
0.8,
0.6,
0.4,
0.3]

Total Loss

Content Loss + Style Loss

Total Loss

$$\mathcal{L}_{content} = \frac{1}{2}\sum(T_c - C_c)^2$$

+

$$\mathcal{L}_{style} = a\sum_i w_i(T_{s,i} - S_{s,i})^2$$

Style Loss

Con1_1
Conv2_1
Conv3_1
Conv4_1
Conv5_1

W = [
1.0,
0.8
0.6,
0.4,
0.3]

Content Loss

Conv5_4

Balancing both style & content loss

Multiply content and style loss with constant term alpha , beta

alpha / beta

alpha = 1

beta = 10

Ratio = 1 / 10

Some Outputs

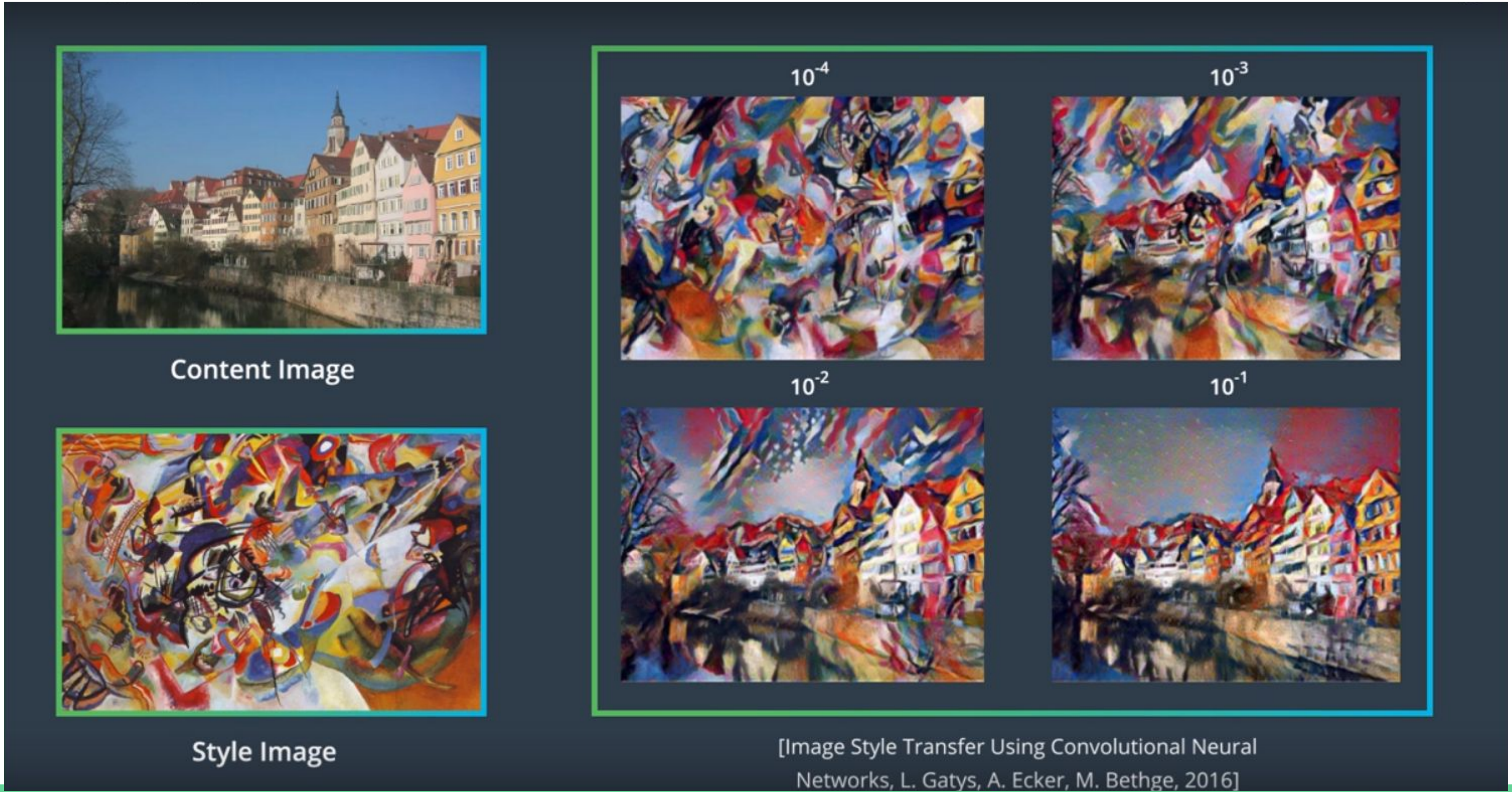
Content Image

Style Image

$$\boxed{\frac{\alpha}{\beta}} = \frac{1}{10}$$

[Image Style Transfer Using Convolutional Neural Networks, L. Gatys, A. Ecker, M. Bethge, 2016]

Cont..


10^-4
10^-3
10^-2
10^-1
Content Image
Style Image
[Image Style Transfer Using Convolutional Neural Networks, L. Gatys, A. Ecker, M. Bethge, 2016]

Code, Jupyter Notebook

https://github.com/balaprasanna/neural_style_transfer

Live Demo

https://is.gd/dnO1D6

Q/A

School of AI / Accenture Hackathon

#healthhack

school of ai

accenture

High performance. Delivered.

24-hour Global Hackathon

Theme: Healthcare (UN Sustainable Development Goal #3)

Start time: 16 February 2019 at 2:00 pm - End time: 17 February 2019 at 6:00 pm

Prizes: **USD 10,000** for global winning team and **USD 1,500** for Singapore winning team; **USD 6,500** for consolation prizes

Singapore Location: Accenture Digital Hub

For more information: knaga82@gmail.com

Resources to Start

- Programming Skills
- Math Skills
- Artificial Intelligence
- Machine Learning
- Deep Learning
- Tooling and Python Libraries
- Frameworks
- YouTube channels
- Blogs and Research Papers

Programming Languages

- Python
  - https://www.udacity.com/course/programming-foundations-with-python--ud036
  - https://developers.google.com/edu/python/
  - https://www.kaggle.com/learn/python

Math Skills

- Probability & statistics
    - https://ocw.mit.edu/courses/mathematics/18-05-introduction-to-probability-and-statistics-spring-2014/index.htm
- Linear Algebra
    - Brown University course on Linear Algebra for CS. 3Blue1Brown
- Calculus
    - https://www.khanacademy.org/math/differential-calculus
    - MIT lectures on Multivariable Calculus
    - MIT linear algebra videos by Gilbert Strang
    -
- Advanced
    - Computational Linear Algebra—Fast.ai
    - Multi-variate Calculus—Khan Academy

Artificial Intelligence

- Book
  - Artificial Intelligence: A Modern Approach http://aima.cs.berkeley.edu/
  - https://www.udacity.com/course/intro-to-artificial-intelligence--cs271
  - https://www.edx.org/course/artificial-intelligence-ai-columbiax-csmm-101x-4

Machine Learning

- Udacity
  - https://eu.udacity.com/course/intro-to-machine-learning--ud120
- Coursera
  - Andrew Ng https://www.coursera.org/learn/machine-learning
- Learn Machine Learning in 3 Months
  - https://github.com/llSourcell/Learn_Machine_Learning_in_3_Months
  - https://www.youtube.com/watch?v=Cr6VqTRO1v0

Deep Learning

- Online Courses
  - https://www.coursera.org/specializations/deep-learning
  - https://www.udacity.com/course/deep-learning--ud730
  - http://cs231n.stanford.edu
- Books
  - http://www.deeplearningbook.org/
  - http://neuralnetworksanddeeplearning.com/index.html
- Learn Deep Learning in 6 Weeks
  - https://github.com/llSourcell/Learn_Deep_Learning_in_6_Weeks
  - https://www.youtube.com/watch?v=waXHrc2m9K8

Frameworks

- Tensorflow
  - https://www.tensorflow.org/tutorials/
- Pytorch
  - https://pytorch.org/
- Keras.
  - https://keras.io/
- Framework comparisons
  - https://www.youtube.com/watch?v=MDP9FfsNx60

Tooling and Python Libraries

- Anaconda & Jupyter Notebook—These are a must for ML & data science.
  - Follow the instructions here to install and set them up.
  - https://colab.research.google.com/github/tensorflow/lucid/blob/master/notebooks/tutorial.ipynb
- Numpy, Matplotlib, Pandas, Scikit-Learn
  - https://medium.com/activewizards-machine-learning-company/top-15-python-libraries-for-data-science-in-in-2017-ab61b4f9b4a7
  - https://medium.freecodecamp.org/essential-libraries-for-machine-learning-in-python-82a9ada57aeb

Blogs & Research Papers

- fast.ai blog
- Distill .pub—Machine Learning Research explained clearly
- Two Minute Papers—Short video breakdowns of AI and other research papers
- Arvix Sanity—More intuitive tool to search through, sort, and save research papers
- Deep Learning Papers Roadmap
- Machine Learning Subreddit—They have 'what are you reading' threads discussing research papers
- Arxiv Insights—This channel has some great breakdowns of AI research papers
- https://github.com/floodsung/Deep-Learning-Papers-Reading-Roadmap

YouTubers (recommended)

- **Siraj Raval**
- Arxiv Insight
- Sentdex
- Two Minute Papers
- Deep Lizard

Thank you