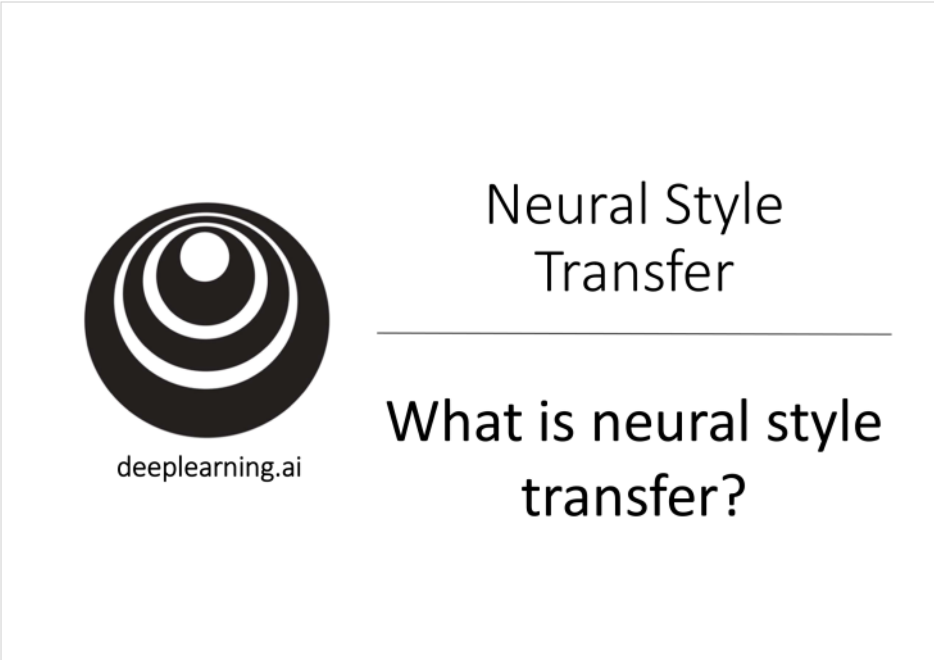
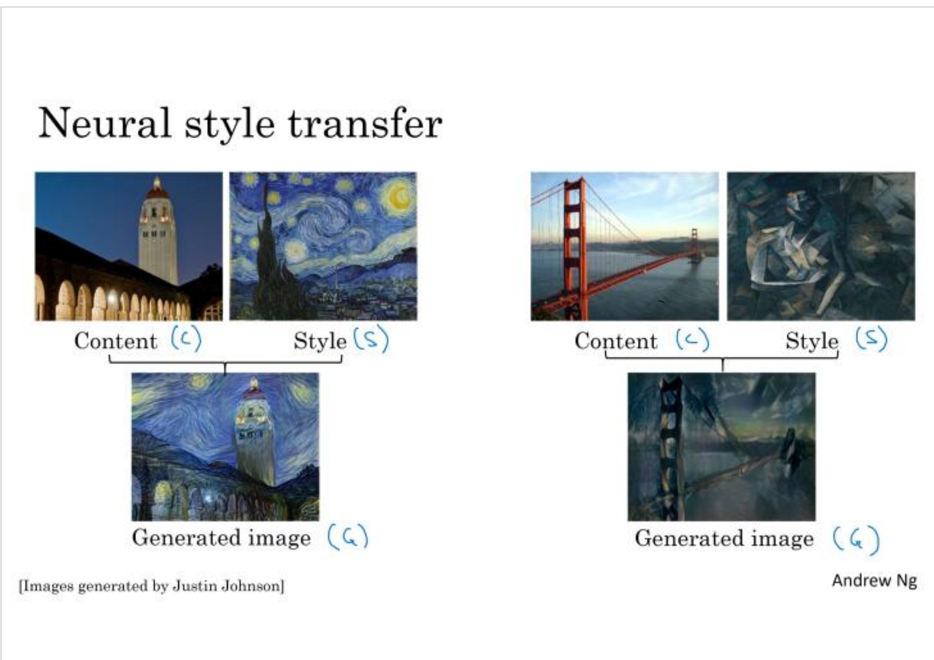
Neural style transfer

Tuesday, June 12, 2018 9:15 PM

what is neural style





Need to look at the features extracted by the conv net to understand how it works

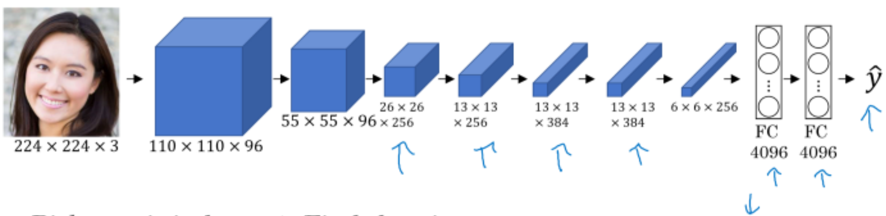
visualize deep convnet

deeplearning.ai

Neural Style Transfer

What are deep ConvNets learning?

Visualizing what a deep network is learning



Pick a unit in layer 1. Find the nine image patches that maximize the unit's activation.

size of filter?,

Repeat for other units.

[Zeiler and Fergus., 2013, Visualizing and understanding convolutional networks]

Andrew Ng

Visualizing deep layers

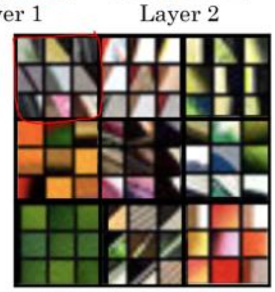
Layer 1 Layer 2 Layer 3 Layer 4 Layer 5

simple features.

Andrew Ng

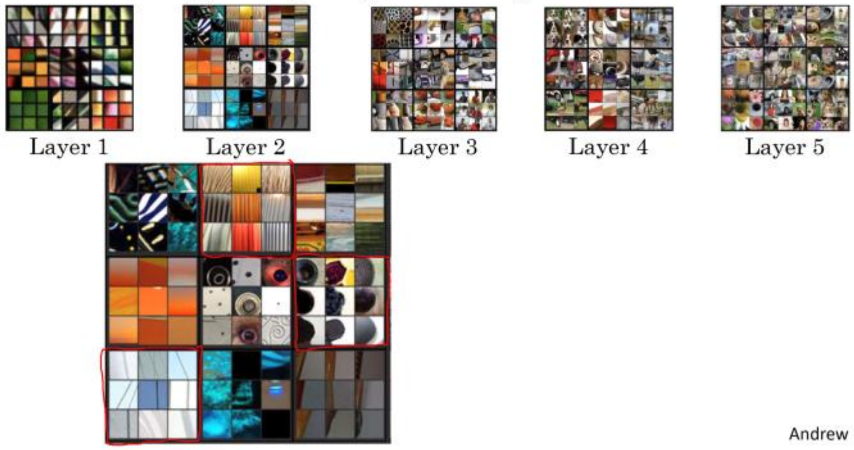
Visualizing deep layers: Layer 1

Layer 1 Layer 2 Layer 3 Layer 4 Layer 5



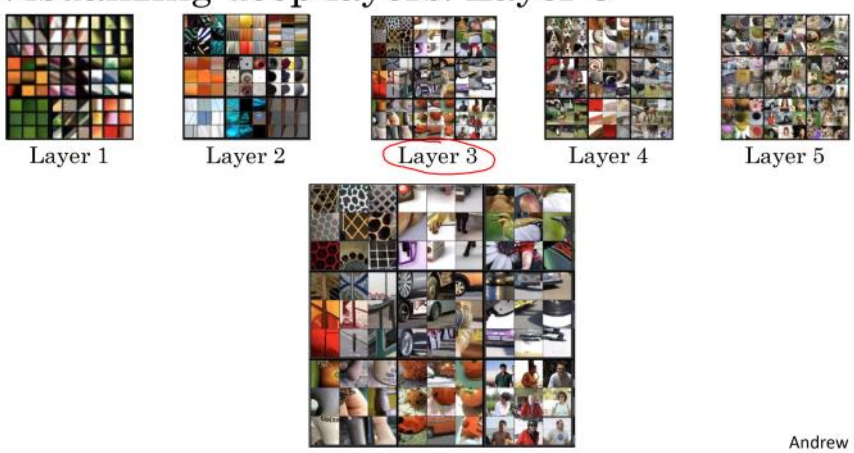
Andrew Ng

Visualizing deep layers: Layer 2



Andrew Ng

Visualizing deep layers: Layer 3



Andrew Ng

Visualizing deep layers: Layer 4

Deeper layer — complex structure

Andrew Ng

flower

dog

Read the paper!

Looks interesting

But not sure how this is done

neural style cost functin

Neural style transfer cost function

Content C

Style S

Generated image G

$$J(G) = \alpha J_{\underline{Content}}(C, G)$$ how similar is the content

$$+ \beta J_{\underline{Style}}(S, G)$$

how similar is the style.

Seems redundant to use two hyper param α, β.

But this is how it is done in the original paper

[Gatys et al., 2015. A neural algorithm of artistic style. Images on slide generated by Justin Johnson] Andrew Ng

Find the generated image G

1. Initiate G randomly

   G: $100 \times 100 \times 3$ (RGB)

2. Use gradient descent to minimize $J(G)$

$$G := G - \frac{\partial}{\partial G} J(G)$$

[Gatys et al., 2015. A neural algorithm of artistic style] Andrew Ng

How to define the two parts of the cost function.

content cost function

Content cost function

$$\underline{J(G)} = \alpha\, \underline{J_{content}(C, G)} + \beta\, J_{style}(S, G)$$

- Say you use hidden layer l to compute content cost.
- Use $\boxed{\text{pre-trained}}$ ConvNet. (E.g., VGG network)
- Let $a^{[l](C)}$ and $a^{[l](G)}$ be the activation of layer l on the images
- If $a^{[l](C)}$ and $a^{[l](G)}$ are similar, both images have similar content

$$J_{content}(C, G) = \frac{1}{2} \underbrace{\| a^{[l](C)} - a^{[l](G)} \|^2}$$

pretrained

Activation – some fully connected representation ?

[Gatys et al., 2015. A neural algorithm of artistic style]

Andrew Ng

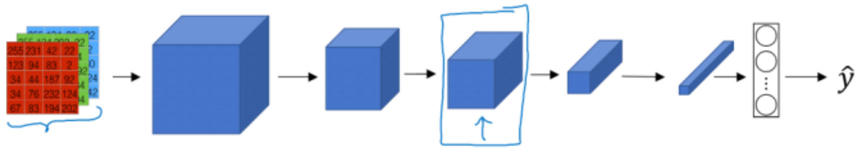
style cost function

Neural Style Transfer

Style cost function

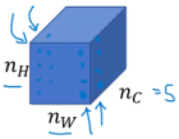
deeplearning.ai

Meaning of the "style" of an image



Say you are using layer l's activation to measure "style."
Define style as correlation between activations across channels.

across channel



How correlated are the activations across different channels?

[Gatys et al., 2015. A neural algorithm of artistic style]

Andrew Ng

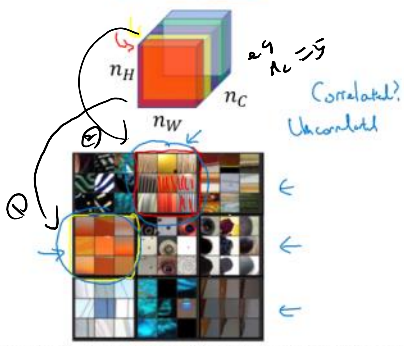
Intuition about style of an image

How correlated are the red & yellow activation

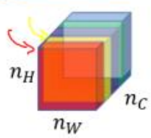
The dgree of correlation across styles

Why does this capture style?

Style image



Generated Image



What does it mean by ①② highly correlated.

wherever ① appear (orange)

② (the /// texture) appear.

[Gatys et al., 2015. A neural algorithm of artistic style]

Andrew Ng

Style matrix

Let $a_{i,j,k}^{[l]}$ = activation at (i, j, k). $G^{[l]}$ is $n_c^{[l]} \times n_c^{[l]}$

position: H W C at hidden [l]

n_c

$G_{kk'}^{[l]}$

$k = 1, \dots, n_c^{[l]}$

style image $\rightarrow$ $G_{kk'}^{[l](S)} = \sum_{i=1}^{n_H^{[l]}} \sum_{j=1}^{n_W^{[l]}} a_{ijk}^{[l](S)} a_{ijk'}^{[l](S)}$ channel k & k' compare

generated image $\rightarrow$ $G_{kk'}^{[l](G)} = \sum_{i=1}^{n_H^{[l]}} \sum_{j=1}^{n_W^{[l]}} a_{ijk}^{[l](G)} a_{ijk'}^{[l](G)}$

"Gram matrix" in linear algebra

unnormalize.

$$\begin{bmatrix} G_{11}^{[l](S)} & G_{12}^{[l](S)} & \cdots \\ & & \end{bmatrix}$$

$$G_{kk'}^{[l](S)} = \sum_{i=1}^{n_H^{[l]}} \sum_{j=1}^{n_W^{[l]}} a_{ijk}^{[l](S)} a_{ijk'}^{[l](S)}$$

Channel们两两相比

$$J_{style}^{[l]}(S, G) = \frac{1}{(\dots)} \| G^{[l](S)} - G^{[l](G)} \|_F^2$$

$$= \frac{1}{(2 n_H^{[l]} n_W^{[l]} n_C^{[l]})^2} \sum_k \sum_{k'} \left(G_{kk'}^{[l](S)} - G_{kk'}^{[l](G)} \right)^2$$

$\uparrow$ β

[Gatys et al., 2015. A neural algorithm of artistic style]

Andrew Ng

Style cost function

normalization.

$\| G^{[l](S)} - G^{[l](G)} \|_F^2$

$$J_{style}^{[l]}(S, G) = \frac{1}{\left(2 n_H^{[l]} n_W^{[l]} n_C^{[l]}\right)^2} \sum_k \sum_{k'} \left(G_{kk'}^{[l](S)} - G_{kk'}^{[l](G)}\right)$$

$$J_{style}(S, G) = \sum_l \lambda^{[l]} J_{style}^{[l]}(S, G)$$

Add layers

weights for earlier / later layers.

Assignment 含义. 估计要在中间.

$$J(G) = \alpha J_{content}(C, G) + \beta J_{style}(S, G)$$

G

[Gatys et al., 2015. A neural algorithm of artistic style]

Andrew Ng

conv 1d and 3d

Aside Apply ConvNet to (1D, 3D) data

Convolutional Networks in 1D or 3D

1D and 3D

deeplearning.ai

Convolutional Networks in 1D or 3D

1D and 3D generalizations of models

Convolutions in 2D and 1D

2D input image
14 × 14

$*$

2D filter
5 × 5

$14 \times 14 \times \underline{3} \ * \ 5 \times 5 \times \underline{3}$

$\rightarrow 10 \times 10 \times 16$

$10 \times 10 \times \underline{16} \ * \ 5 \times 5 \times \underline{16}$

$\rightarrow 6 \times 6 \times \underline{32}$

$14 \times \underline{1} \ * \ 5 \times \underline{1}$

$\rightarrow 10 \times \underline{16}$ — 16 filters.

1	20	15	3	18	12	4	17

14

$*$

1	3	10	3	1

5

$10 \times 16 \ * \ 5 \times \underline{16}$

$\rightarrow 6 \times \underline{32}$

EKG data: voltage at each instance

Andrew Ng

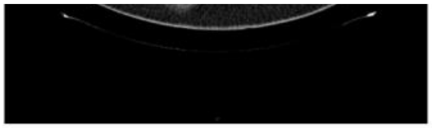
— can convolve with a ~~of~~ 1D filter.

or — can use RNN.

Compare pros & cons.

1D CNN vs RNN

3D data



Andrew Ng

3D data

Andrew Ng

3D data

Andrew Ng

3D data

Andrew Ng

3D data

Andrew Ng

3D data

Andrew Ng

3D data

Andrew Ng

3D data

Andrew Ng

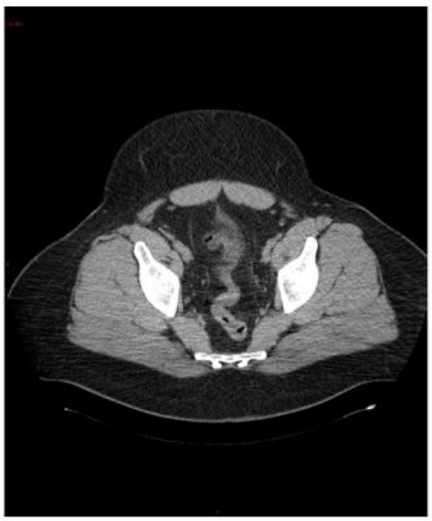
3D data

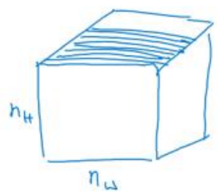
Andrew Ng

3D data

Andrew Ng

3D data





Andrew Ng

3D convolution

3D volume

*

3D filter

need not be cube. just an example

$14 \times 14 \times 14 \times \boxed{1}$ (n_c)

$* \; 5 \times 5 \times 5 \times 1$ 16 filters

$\rightarrow 10 \times 10 \times 10 \times 16$

$* \; 5 \times 5 \times 5 \times 16$ 32 filters

$\rightarrow 6 \times 6 \times 6 \times 32$

Can extend it to : movie data

Different time as one slice

⇒ detect MOTION !

Andrew Ng

2. Why do we learn a function $d(img1, img2)$ for face verification? (Select all that apply.)

- [x] We need to solve a one-shot learning problem. ✓
- [ ] This allows us to learn to predict a person's identity using a softmax output unit, where the number of classes equals the number of persons in the database plus 1 (for the final "not in database" class). ✗
- [x] This allows us to learn to recognize a new person given just a single image of that person.
- [ ] Given how few images we have per person, we need to apply transfer learning.

7. Neural style transfer is trained as a supervised learning task in which the goal is to input two images (x), and train a network to output a new, synthesized image (y).

- ○ True
- ● False