

Special applications: Face recognition & Neural style transfer

10/10 points
(100.00%)

Quiz, 10 questions

✓ **Congratulations! You passed!**

Next Item



1 / 1
points

1.

Face verification requires comparing a new picture against one person's face, whereas face recognition requires comparing a new picture against K person's faces.



1 / 1
points

2.

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



1 / 1
points

3.

In order to train the parameters of a face recognition system, it would be reasonable to use a training set comprising 100,000 pictures of 100,000 different persons.



1 / 1
points

4.

Special applications: Face recognition & Neural style transfer

10/10 points
(100.00%)

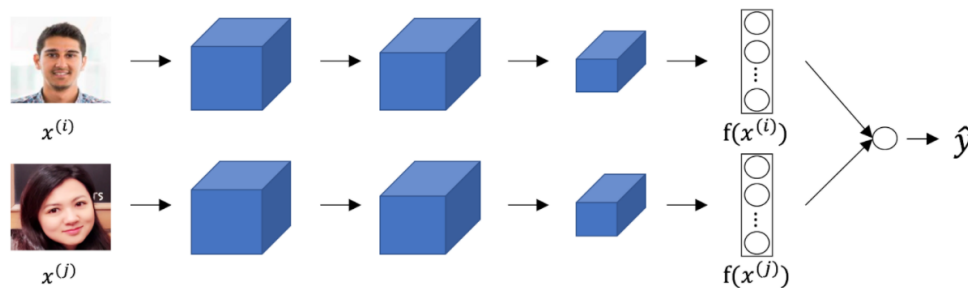
Quiz, 10 questions



1 / 1
points

5.

Consider the following Siamese network architecture:



The upper and lower neural networks have different input images, but have exactly the same parameters.



1 / 1
points

6.

You train a ConvNet on a dataset with 100 different classes. You wonder if you can find a hidden unit which responds strongly to pictures of cats. (I.e., a neuron so that, of all the input/training images that strongly activate that neuron, the majority are cat pictures.) You are more likely to find this unit in layer 4 of the network than in layer 1.



1 / 1
points

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).

Special applications: Face recognition & Neural style transfer

10/10 points
(100.00%)

Quiz, 10 questions

✓
1 / 1
points

8. In the deeper layers of a ConvNet, each channel corresponds to a different feature detector. The style matrix $G^{[l]}$ measures the degree to which the activations of different feature detectors in layer l vary (or correlate) together with each other.

✓
1 / 1
points

9. In neural style transfer, what is updated in each iteration of the optimization algorithm?

✓
1 / 1
points

10. You are working with 3D data. You are building a network layer whose input volume has size $32 \times 32 \times 32 \times 16$ (this volume has 16 channels), and applies convolutions with 32 filters of dimension $3 \times 3 \times 3$ (no padding, stride 1). What is the resulting output volume?

