**Congratulations! You passed!**

**Grade received** 100% **To pass** 80% or higher

Go to next item

# Special Applications: Face Recognition & Neural Style Transfer

**Latest Submission Grade 100%**

**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 point

- (●) True
- ( ) False

**Correct**
Correct.

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

1 / 1 point

- [x] We need to solve a one-shot learning problem.

**Correct**
This is true as explained in the lecture.

- [ ] 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.

**Correct**
No

- [ ] Given how few images we have per person, we need to apply transfer learning.

**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 point

- ( ) True
- (●) False

**Correct**
Correct, to train a network using the triplet loss you would need several pictures of the same person.

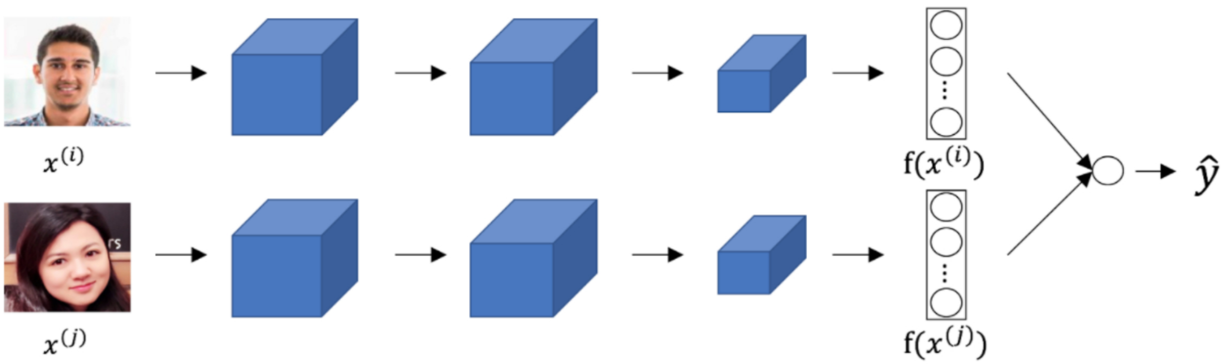
**4.** Which of the following is a correct definition of the triplet loss? Consider that $\alpha > 0$. (We encourage you to figure out the answer from first principles, rather than just refer to the lecture.)

1 / 1 point

- (●) $max(||f(A) - f(P)||^2 - ||f(A) - f(N)||^2 + \alpha, 0)$
- ( ) $max(||f(A) - f(N)||^2 - ||f(A) - f(P)||^2 - \alpha, 0)$
- ( ) $max(||f(A) - f(P)||^2 - ||f(A) - f(N)||^2 - \alpha, 0)$
- ( ) $max(||f(A) - f(N)||^2 - ||f(A) - f(P)||^2 + \alpha, 0)$

**Correct**
Correct

**5.** Consider the following Siamese network architecture: **1 / 1 point**




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

- (●) True
- ( ) False

**Correct**
Yes it is true, parameters are shared among these two networks.

**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 point**

- (●) True
- ( ) False

**Correct**
Yes, this neuron understands complex shapes (cat pictures) so it is more likely to be in a deeper layer than in the first layer.

**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 ($y$). **1 / 1 point**

- ( ) True
- (●) False

**Correct**
Yes, Neural style transfer is about training on the pixels of an image to make it look artistic, it is not learning any parameters.

**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 point**

- ( ) False
- (●) True

**Correct**
Yes, the style matrix $G^{[l]}$ can be seen as a matrix of cross-correlations between the different feature detectors.

**9.** In neural style transfer, what is updated in each iteration of the optimization algorithm? **1 / 1 point**

- ○ The regularization parameters
- ○ The neural network parameters
- ● The pixel values of the generated image $G$
- ○ The pixel values of the content image $C$

**Correct**
Yes, neural style transfer is different from many of the algorithms you've seen up to now, because it doesn't learn any parameter, instead it learns directly the pixels of an image.

**10.** You are working with 3D data. You are building a network layer whose input volume has size 32x32x32x16 (this volume has 16 channels), and applies convolutions with 32 filters of dimension 3x3x3 (no padding, stride 1). What is the resulting output volume?

**1 / 1 point**

- ● 30x30x30x32
- ○ Undefined: This convolution step is impossible and cannot be performed because the dimensions specified don't match up.
- ○ 30x30x30x16

**Correct**
Correct, you have used the formula $\lfloor \frac{n^{[l-1]} - f + 2 \times p}{s} \rfloor + 1 = n^{[l]}$ over the three first dimensions of the input data.