**Reshaping tensors**

Later in the course, you will classify images of sign language letters using a neural network. In some cases, the network will take 1-dimensional tensors as inputs, but your data will come in the form of images, which will either be either 2- or 3-dimensional tensors, depending on whether they are grayscale or color images.

The figure below shows grayscale and color images of the sign language letter A. The two images have been imported for you and converted to the numpy arrays gray\_tensor and color\_tensor. Reshape these arrays into 1-dimensional vectors using the reshape operation, which has been imported for you from tensorflow. Note that the shape of gray\_tensor is 28x28 and the shape of color\_tensor is 28x28x3.



**Instructions**

**100 XP**

* Reshape gray\_tensor from a 28x28 matrix into a 784x1 vector named gray\_vector.
* Reshape color\_tensor from a 28x28x3 tensor into a 2352x1 vector named color\_vector.

# Reshape the grayscale image tensor into a vector

gray\_vector = reshape(gray\_tensor, (784, 1))

# Reshape the color image tensor into a vector

color\_vector = reshape(color\_tensor, (2352, 1))

Excellent work! Notice that there are 3 times as many elements in color\_vector as there are in gray\_vector, since color\_tensor has 3 color channels.