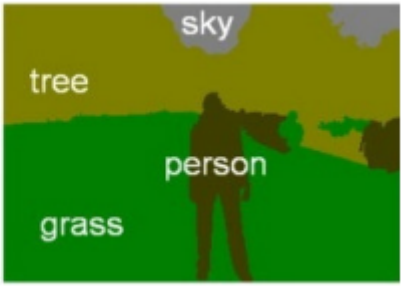
Segmentation

What is Segmentation?

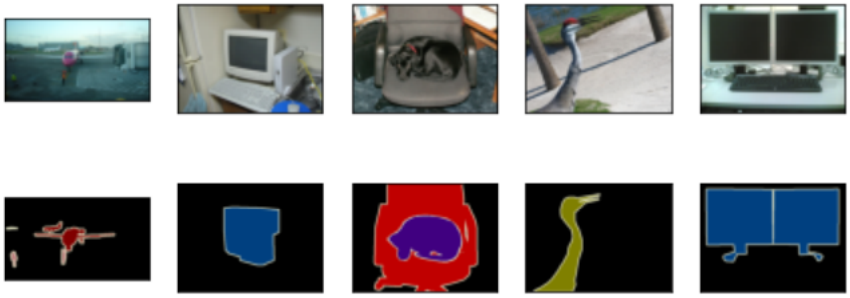
Segmentation is an image processing topic which focuses on dividing an image in parts which have the *same properties* (i.e. the same textures etc..).

Semantic Segmentation

Semantic segmentation: classify each pixel in an image according to the object category it belongs to.



Building supervised training set is expensive, since it requires a complex human operation. The *label is itself an image*, with a different color for each category:



For this reason, semantic segmentation can be regarded as a special case of Image-to-image transformation.

Convolutionalization

Composing convolutions *we still get a convolution.* Specifically, the composition of convolutional layers essentially behaves as a convolutional layer. The stride of the compund covolution is the *product of the strides of the components.*

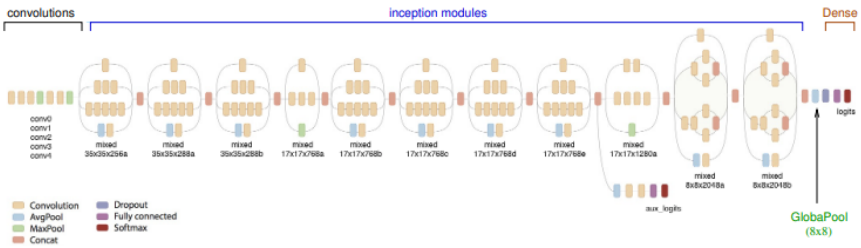
Dimensions

As we know, $\frac{D_{in} + P - K}{S} + 1 = D_{out}$, or equivalently $D_{in} = S * (D_{out} - 1) + K$. Suppose to compose two kernels with dimension 3 and stride 1. Then, the intermediate dimension is $(1-1) * 1 + 3 = 3$ and the initial dimension must be $(3-1) * 1 + 3 = 5$.

Suppose to compose a kernel of dimension $K_1 = 5$ and stride $S_1 = 3$ with another kernel of dimension $K_2 = 3$ and stride $S_2 = 2$. Then, applying the rule $D_{in} = S * (D_{out} - 1) + K$ we get, for $D_2 = 1$: - $D_1 = S_2 * (D_2 - 1) + K_2 = 3$ - $D_0 = S_1 * (D_1 - 1) + K_1 = 11$ $D_0 = 11$ is the dimension of the compund kernel, aka its "*receptive fields*".

What breaks convolutionality

Let us consider a typical architecture for image classification such as Inception



V3:

Composing convolutional layers we still get a convolutional network. What breaks convolutionality are the *dense layers* at the end of networks (if maxpooling has a fixed pooling dimension).