Mapping Features to Labels

Imagine you are teaching a computer how to ‘see’ something. By that we mean not just be able to process the pixels in an image for color or shade, or anything like that, but to be able to understand the *contents* of an image.

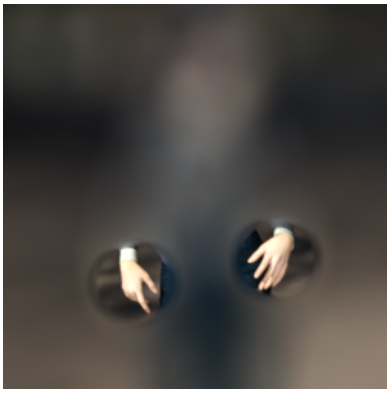
To simulate this, you could say that the understanding of the contents of the image are effectively blurred...like this:

There’s information in there, but I don’t know what it represents, so I simulate that by blurring the image.

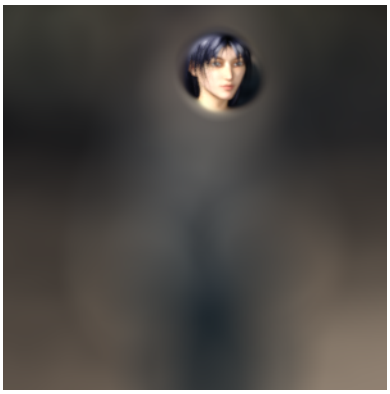
Now, next, imagine that there’s a convolution (filter) that can extract something from the image consistently, and the something that it extracts is *always* present when the image is labelled with a particular class. In other words, if there’s a filter that always produces something like this, when the image is labelled *human*.

You and I both know that these clearly seem to be human legs, but the computer doesn’t know that. It just knows that two cylinder-like objects like these tend to show up for a particular filter, and only on images that are labelled human.

Then, similarly, there's another filter that produces this, and only for human images:

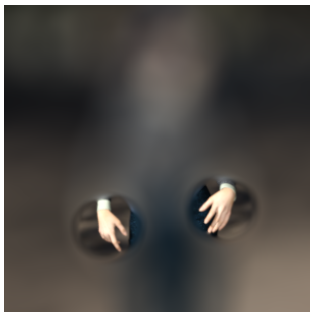
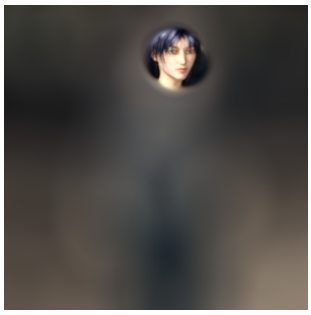


...and then another that produces this, or something similar, again, only for human images:



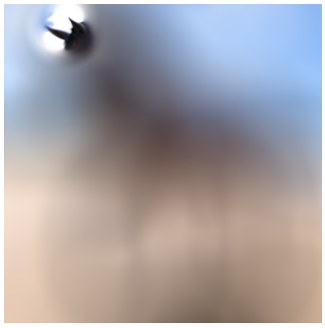
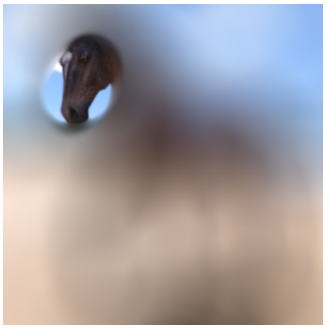
You and I both recognize this as a human face. But the computer doesn't. It only knows that something clear is regularly extracted by that filter, when the image is labelled as human.

Thus, when a set of filters is learned that consistently extracts content like this when the image is labelled as human, we could say that the following 'equation' holds:

 +  +  = HUMAN

If there are 64 filters, for example, in the final layer, they may all return ‘nothing’ and it’s just these three end up having significance for this class.

Similarly, a different set of filters could return values for the label HORSE, and everything else (including the three filters that gave us human hands, legs and face) would return nothing, so we’d get:

 +  +  = HORSE

Now we have a set of filters that a model has learned that can extract the features that indicate what is a horse and what is a human!

Do note that for this example I used features that you and I recognize, like hands, legs and feet as distinguishing between the two, but the computer is NOT limited to that. It might be able to consistently ‘see’ patterns in images that you do not, and that might be a more accurate determinant of the class of the image. The field of convolutional visualization studies this, and it’s fascinating to learn the interpretability of images that are classified using this method!