MODERN COMPUTER VISION

BY RAJEEV RATAN

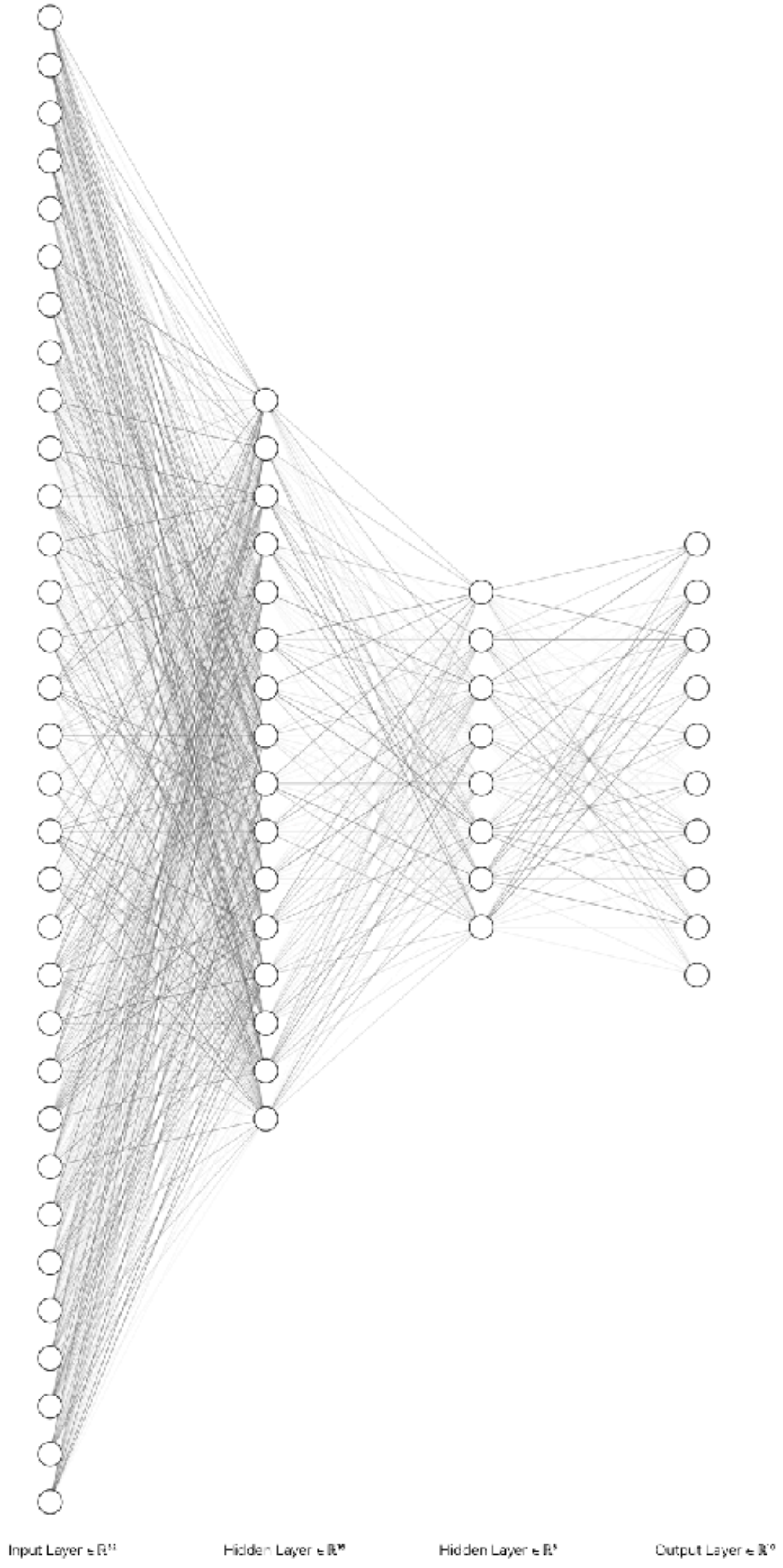
Why CNNs Work So Well For Images

We explore design choices in CNNs and how it relates to their performance in the real world

Standard Neural Networks

A thought experiment…

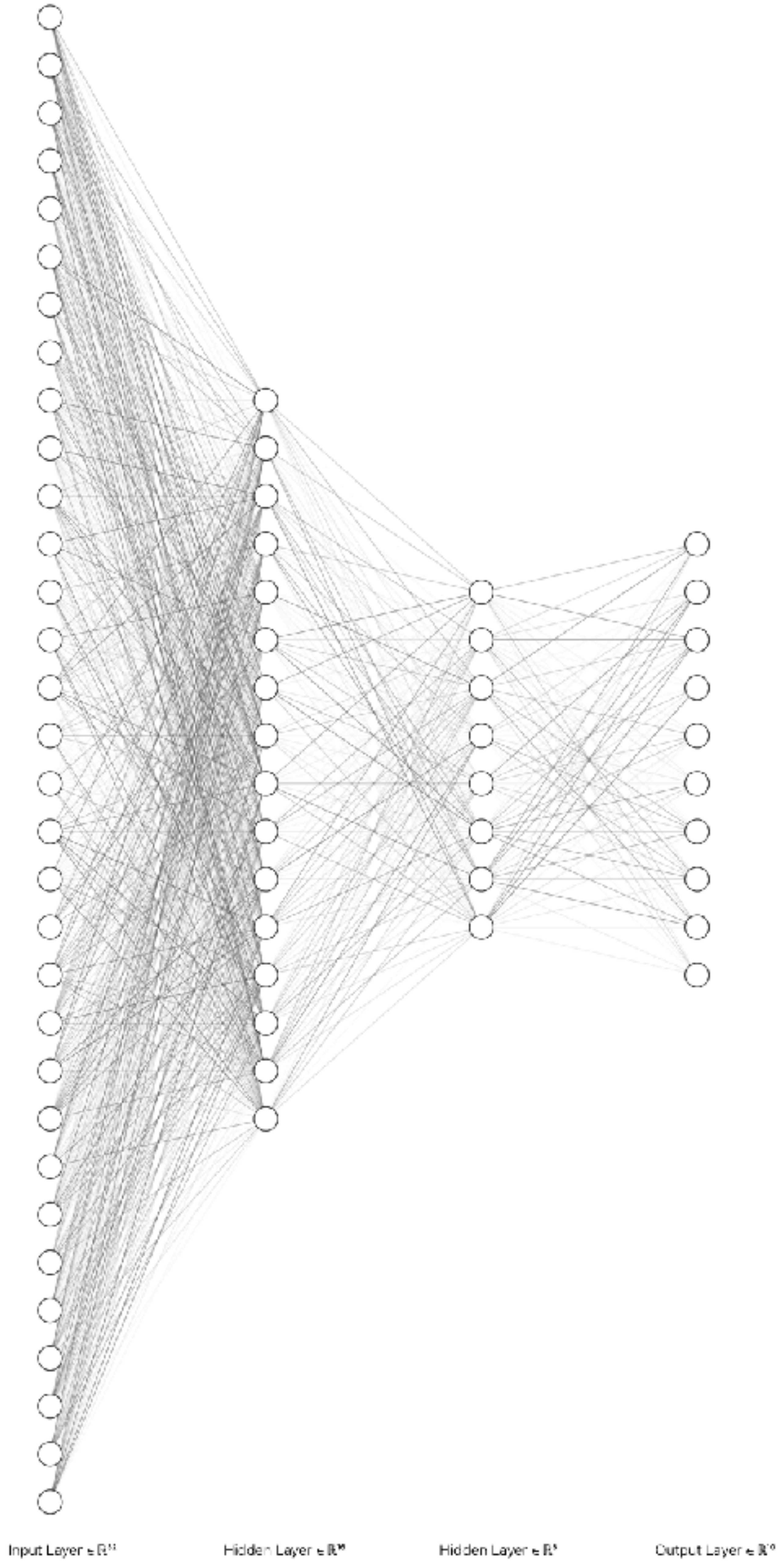
- Standard Neural Networks don't have Convolution Filter Inputs
- For Images every pixel will be it's own input
- Therefore, a small image that's 28 x 28 would have 784 input nodes for our first layer



Standard Neural Networks

A thought experiment…

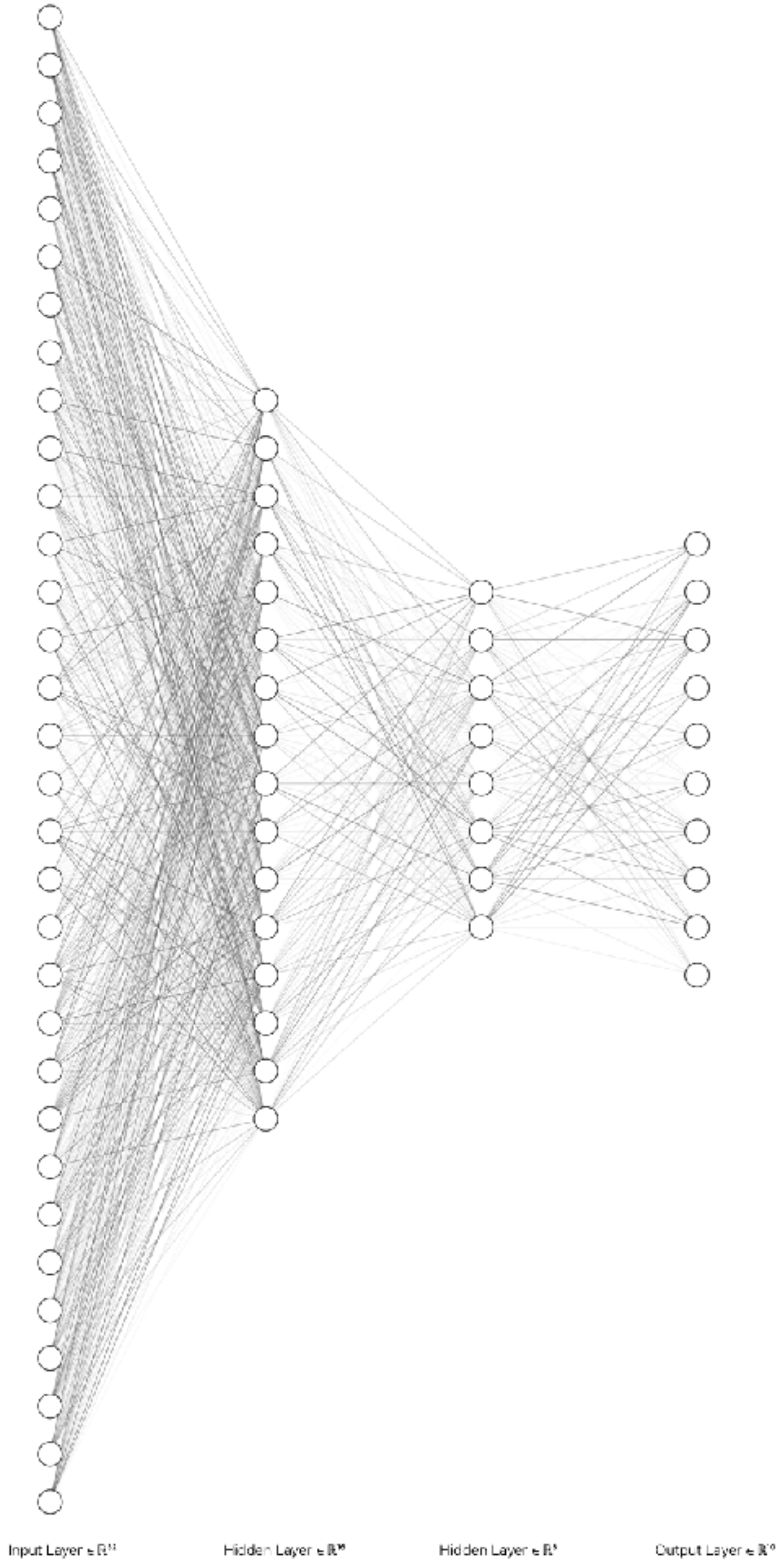
- Our second layer, if we breakdown our previous CNN model would be:
    - 32 Filters for 26 x 26 Feature Maps
    - 13,312
- If they were fully connected, our weight matrix would be:
    - 784 x 13,312 = **10,436,608**
- For just one hidden layer!



Standard Neural Networks

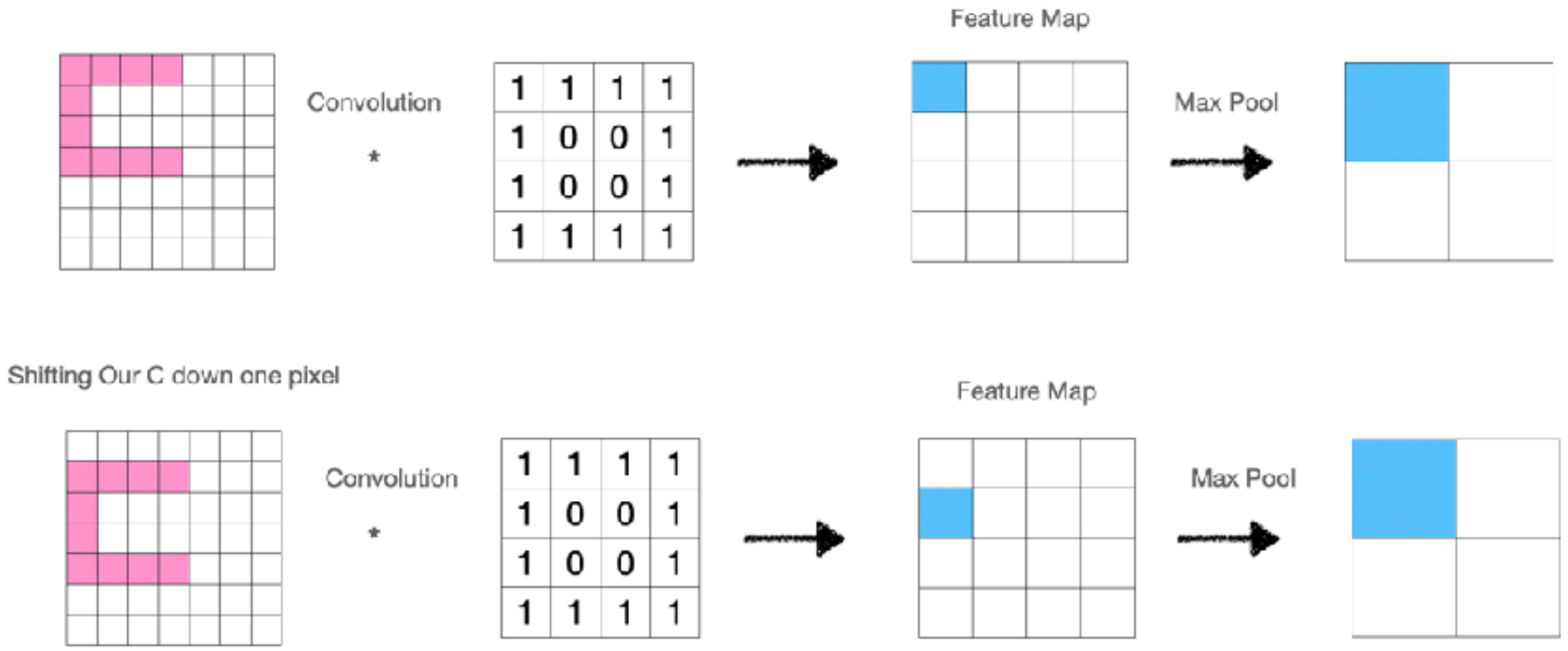
Not Feasible for Image Classification!

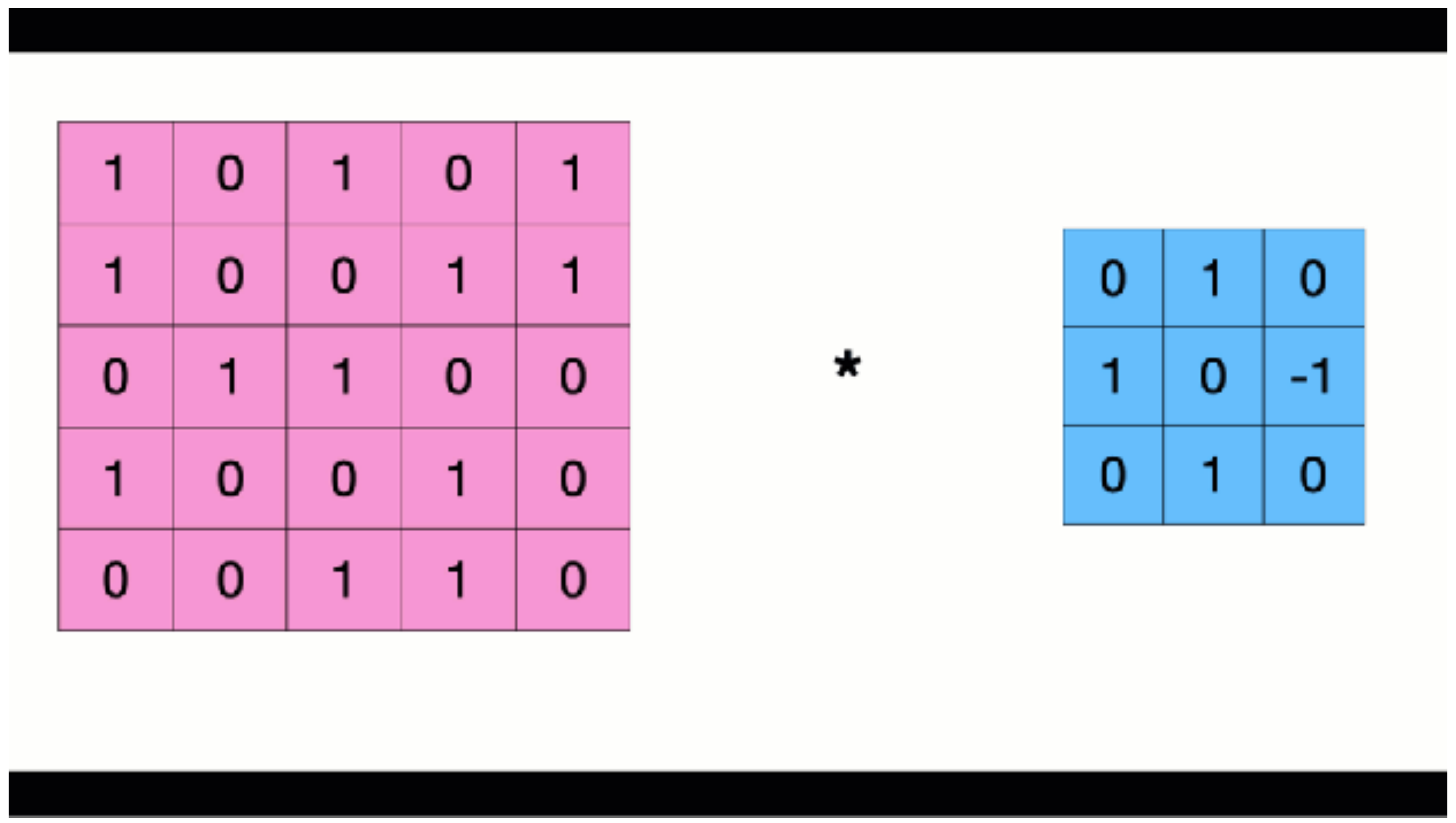
- Not scalable to large data inputs
- Overfitting



Advantages of Convolution Neural Networks

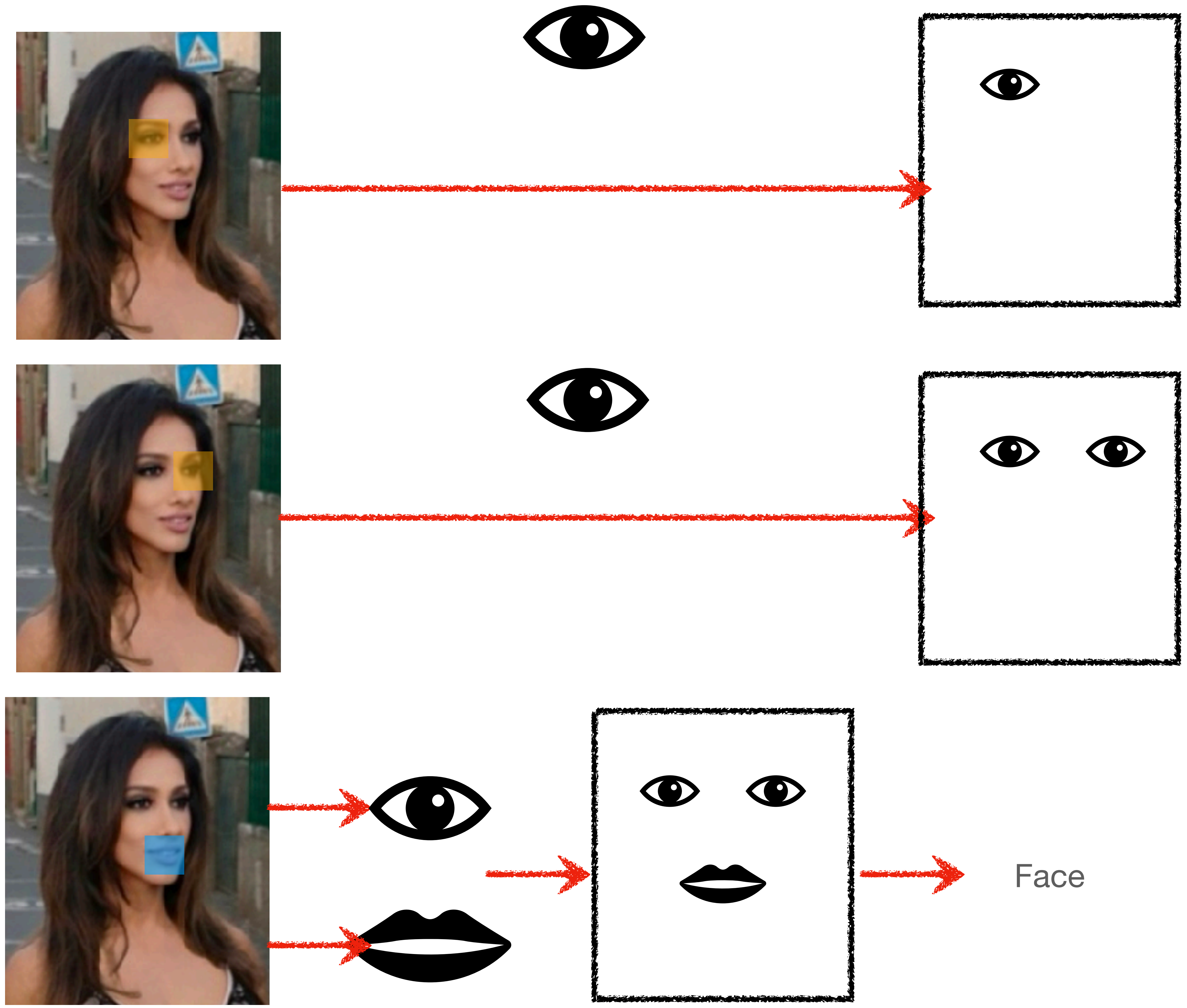
- **Parameter sharing** - where a single filter can be used all parts of an image
- **Sparsity of connections** - As we saw, fully connected layers in a typical Neural Network result in a weight matrix with large number of parameters.
- **Invariance** - Remember our Max Pool Example





Convolution Neural Networks Assumptions

- Low-level features are local
- Features are translational invariant
- High-level features are made up of low-level features



MODERN COMPUTER VISION

BY RAJEEV RATAN

Next…

How to Train a CNN