# CS342 - Section 7

- Homework 6
  - Due 10/11, can still turn it in until Sunday
  - How many layers does your model have?
- Homework 7
  - Posted, due next Thursday
  - On image segmentation
- Homework 5
  - Grades posted

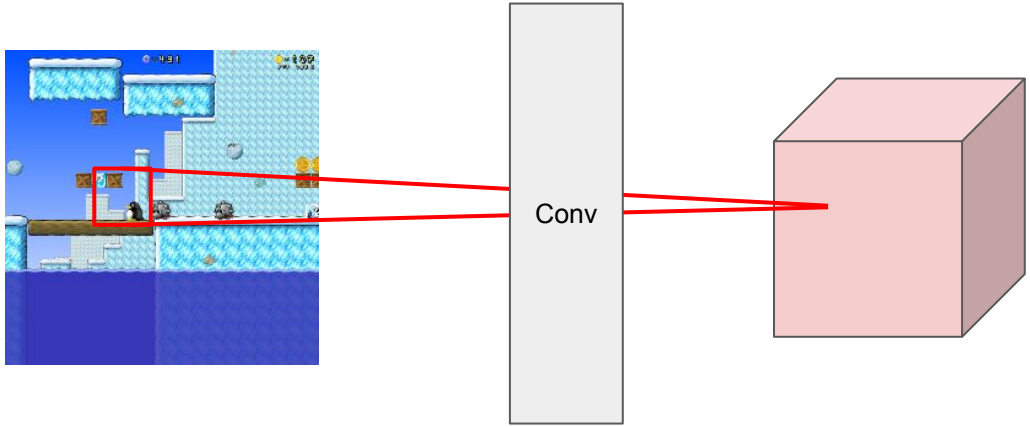
Questions?

# Agenda

- Receptive Field
- Fully connected layer ⇔ convolutional layer
- Up-convolution
- Linear-upsampling
- PyTorch code walk through on up-convolution/FCN

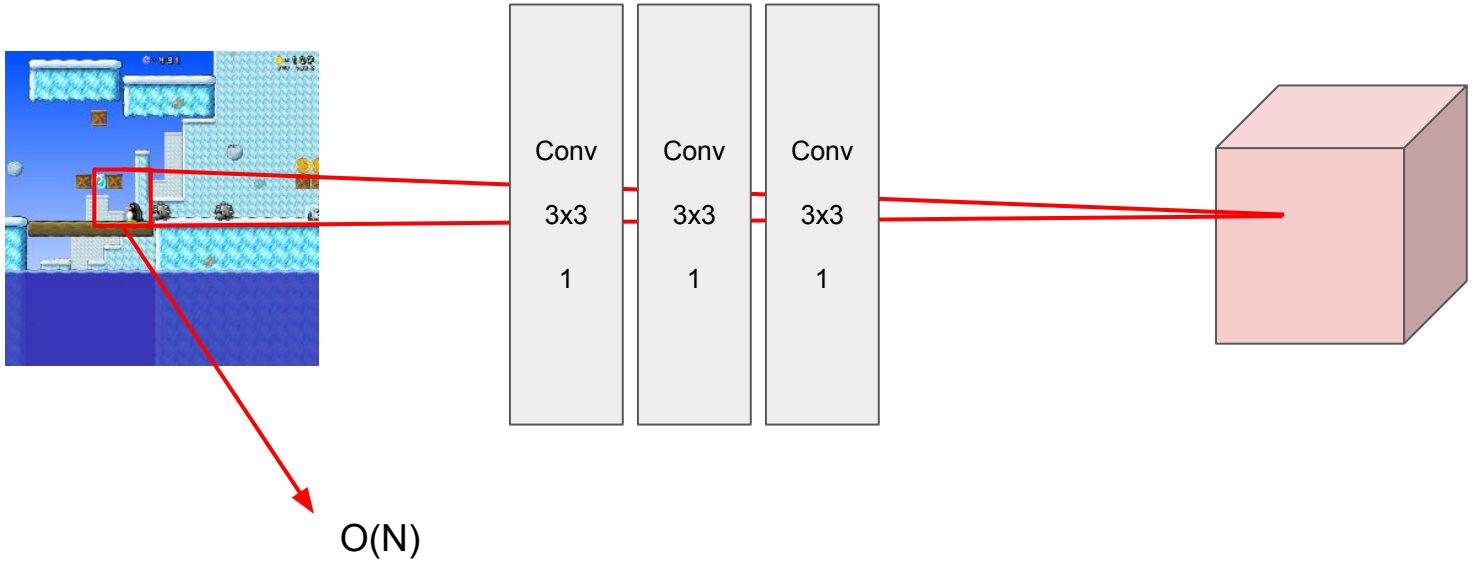
# Receptive Field

- Region of input that affect the value of certain pixel in later layers



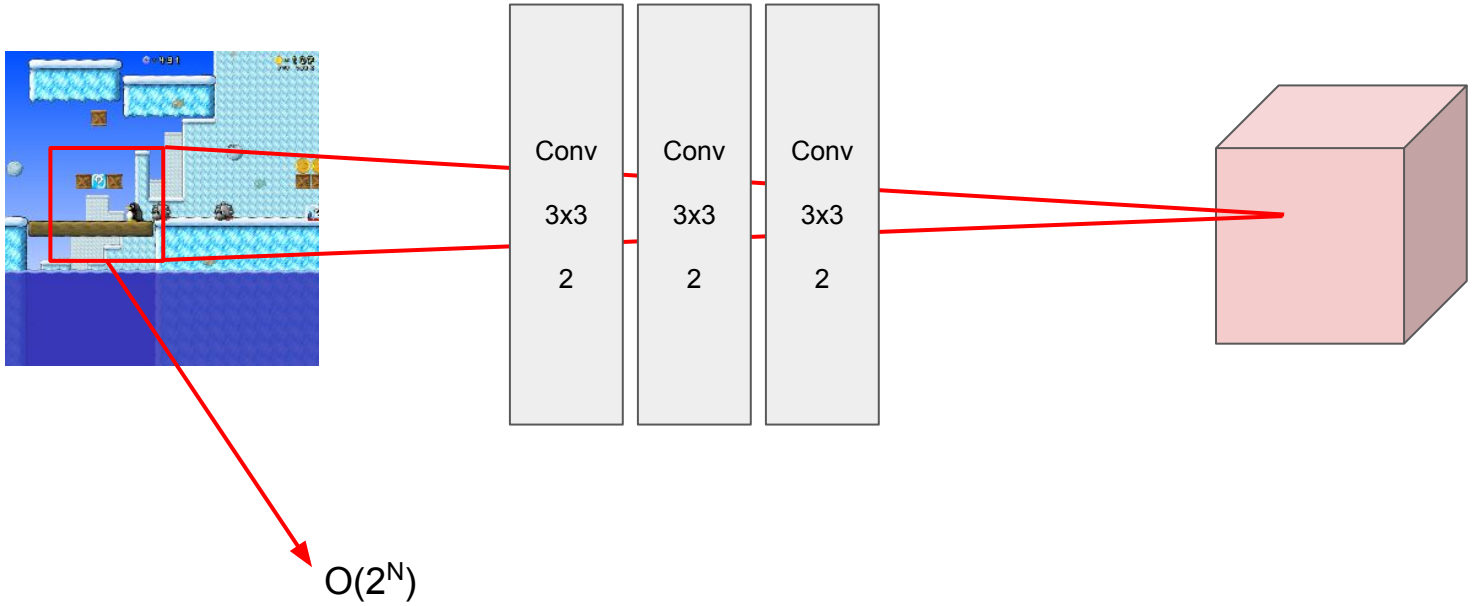
# Receptive Field

- How fast does it grow if there is no stride/pooling?



# Receptive Field

- How fast does it grow if there is stride/pooling?

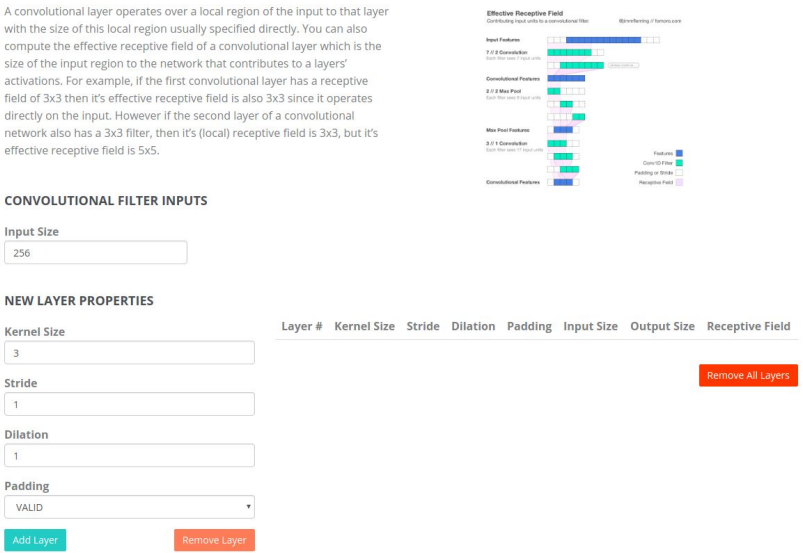


# Receptive Field

- How to calculate the receptive field
  - By hand
    - Use kernel size, stride, pooling, pad in each layer
    - Corner cases can be really dirty
  - NaN trick
    - Pass in NaN as gradient, back prop to input to see how many end up NaN too
    - PyTorch Demo
    - https://gist.github.com/dianchen96/8ac971ab667cc875619240cb2254425a

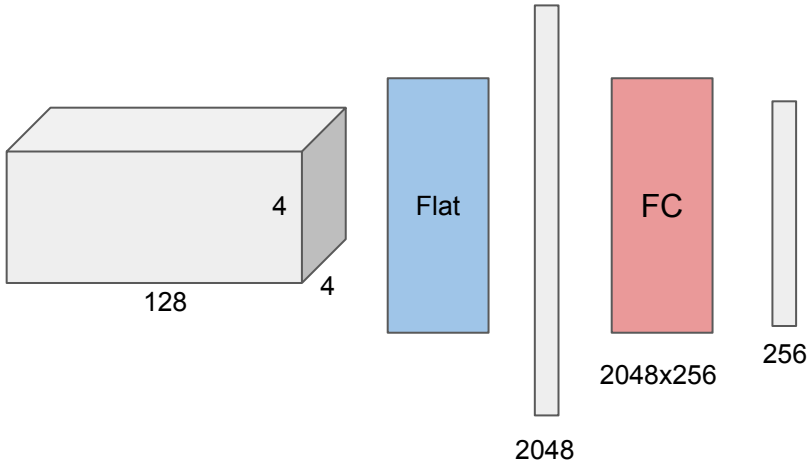
# Receptive Field

- Online receptive field calculation
  - http://fomoro.com/tools/receptive-fields/



# Fully connected layer ⇔ convolutional layer

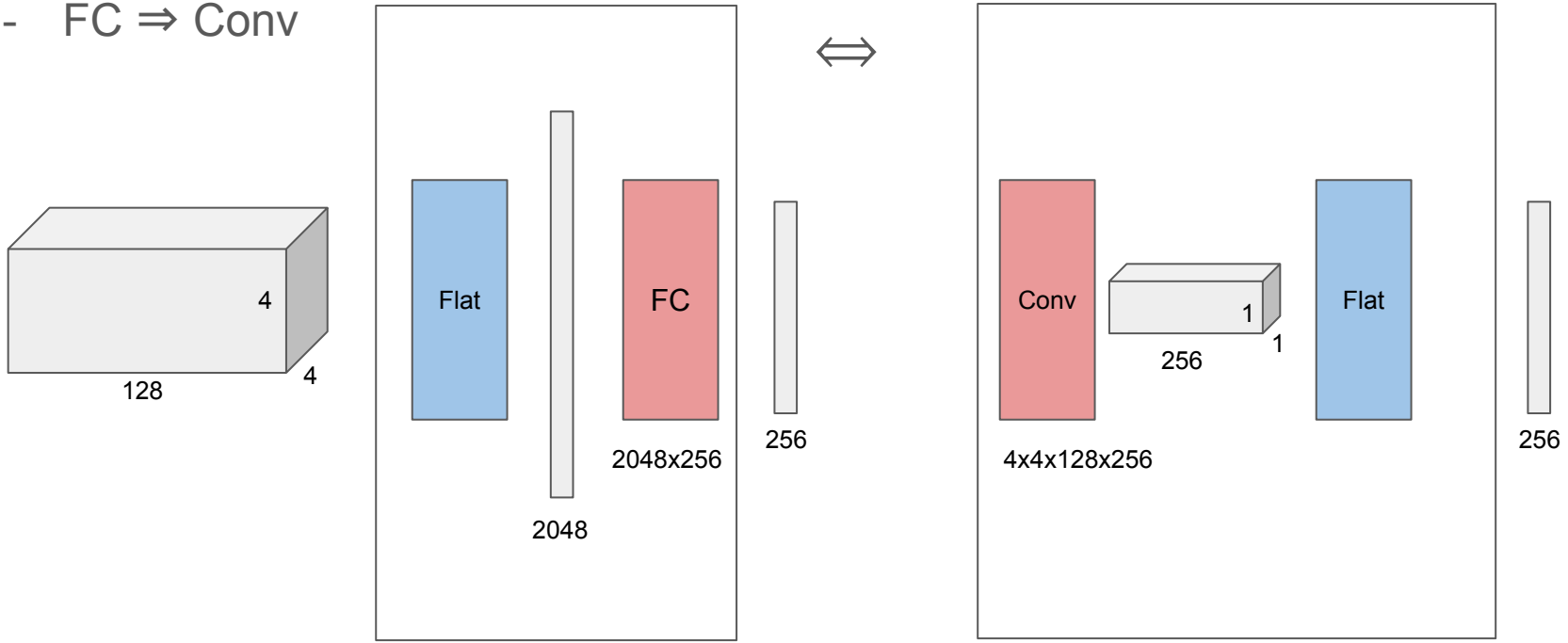
- FC ⇒ Conv




- How to convert this into convolution?

# Fully connected layer ⇔ convolutional layer

- FC ⇒ Conv



# Linear Upsampling

- Recap
- One way to upsample
  - Pro: best results
  - Con: slow
- PyTorch
  - torch.nn.Upsample(mode='bilinear')

# Up-convolution

- Recap
- Another way to upsample
  - Pro: faster
  - PyTorch
- PyTorch
  - torch.nn.ConvTranspose2d(...)

# PyTorch Code walkthrough