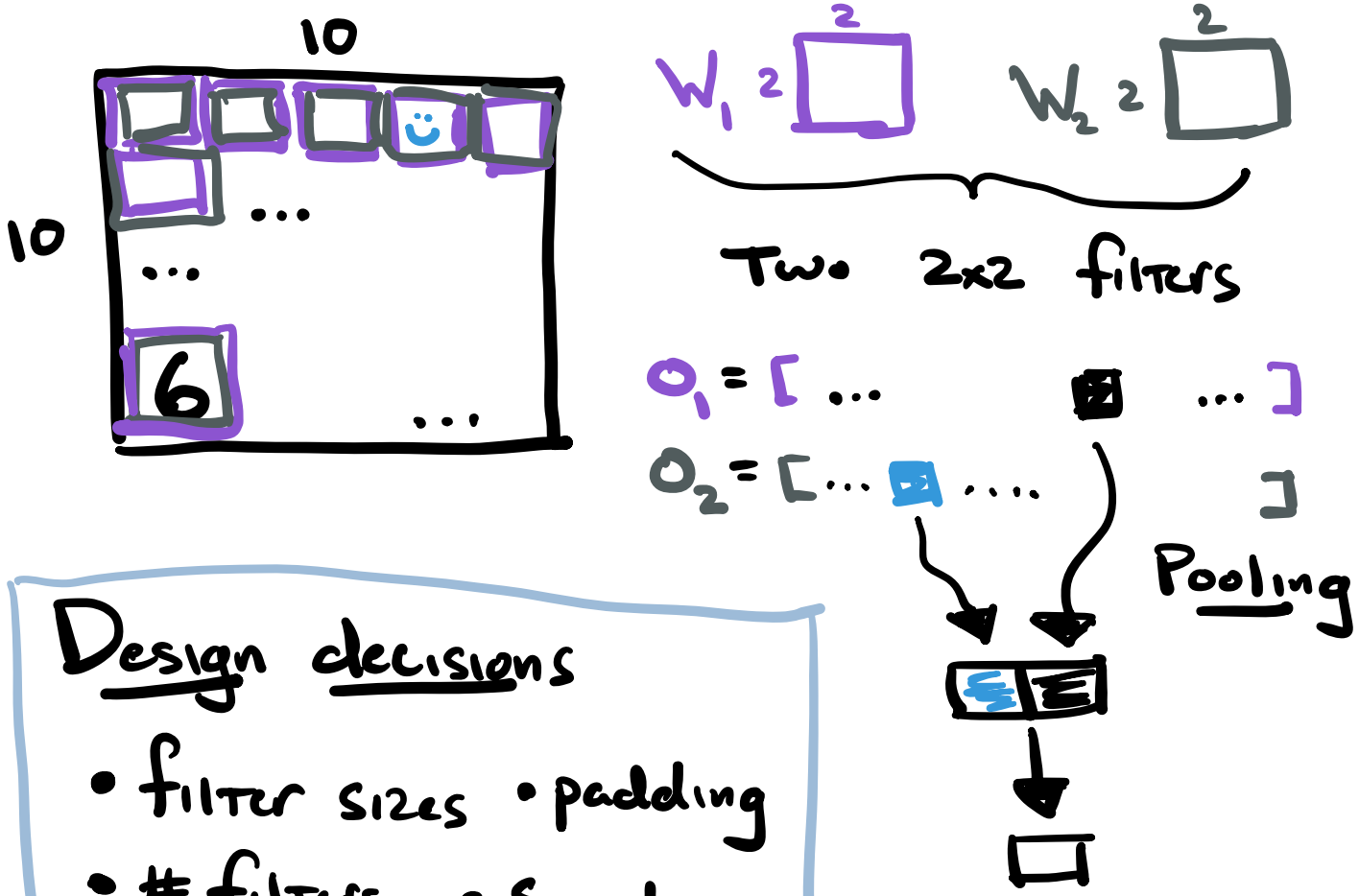
DS 4440 Conv Nets (2)



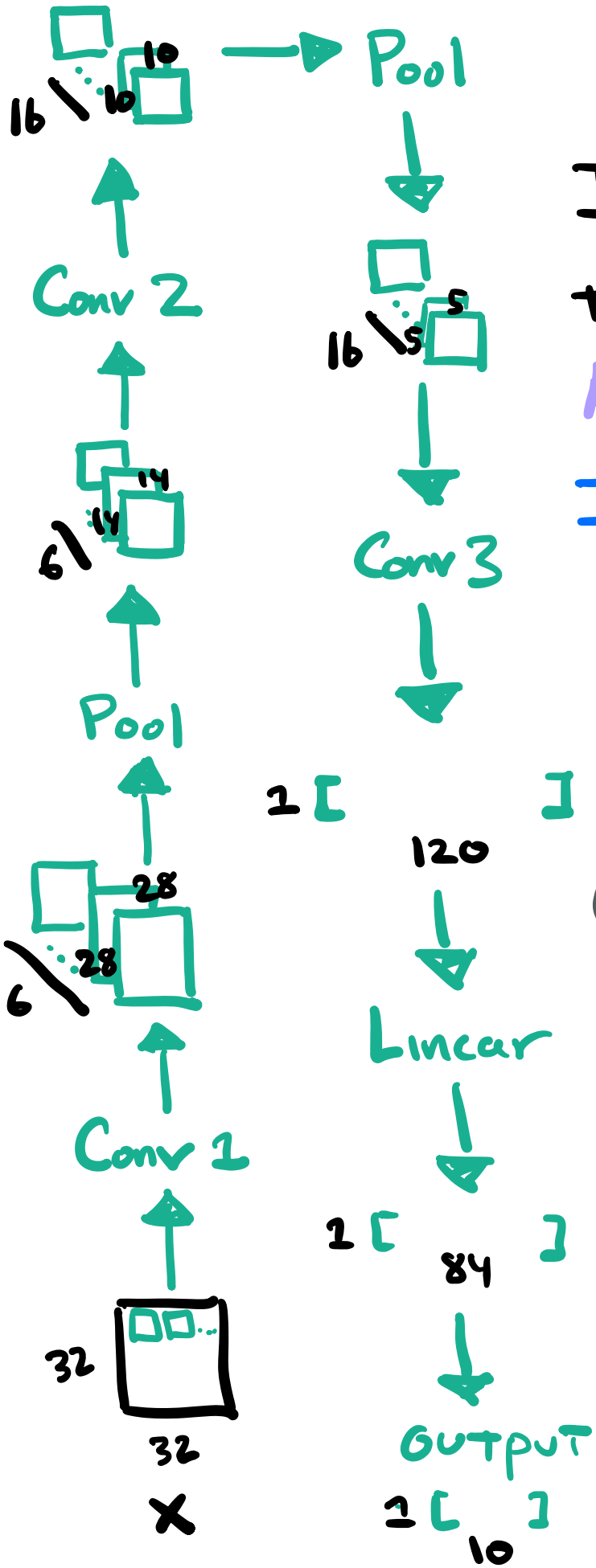
Design decisions

- filter sizes
- padding
- # filters
- Stride

In practice: Deep models dominate.

Some history: LeNet (LeCun, '98)

But hardware & data had to catch up.



LeNet.

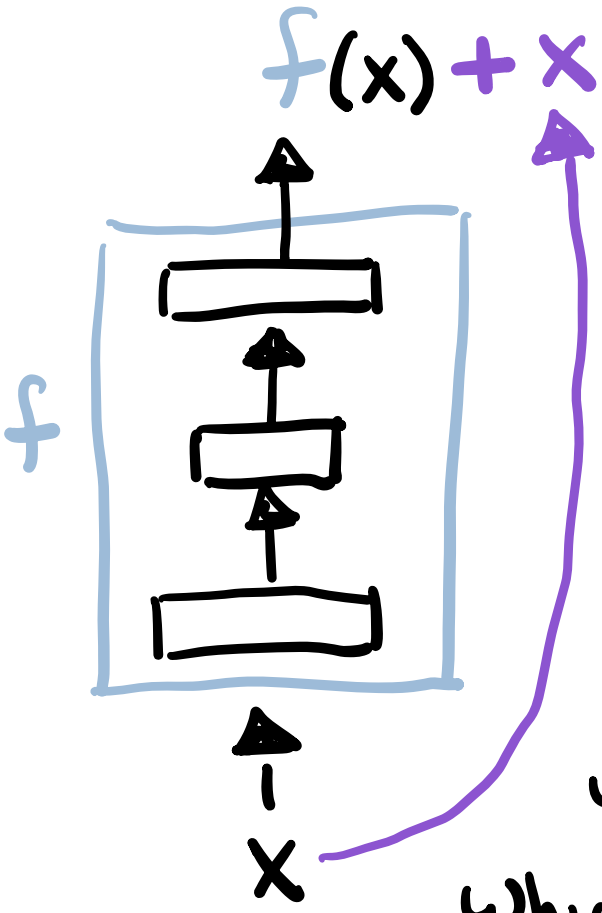
In 2012, the time came — AlexNet won the ImageNet challenge.

Was deeper and used ReLu.

(See CoLab Notebook.)

Residual Connections

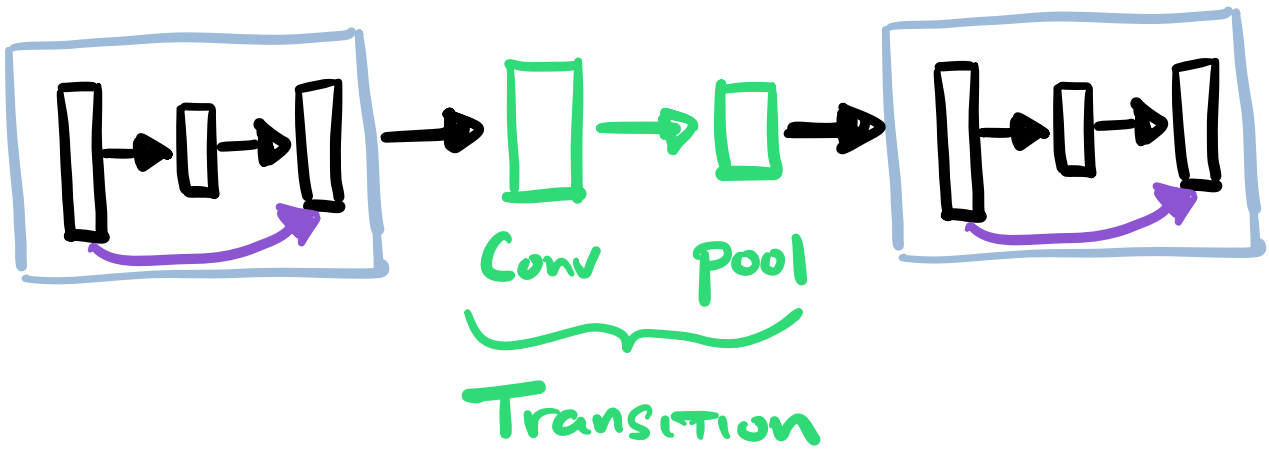
Training gets tricky with deeper nets. Adding paths that skip layers can help.



Can add or concat.
$+$ $\oplus$

If we $\oplus$, dimensions may get out of hand.

DenseNet and similar use transition layers which downsample feature maps (e.g., via pooling)



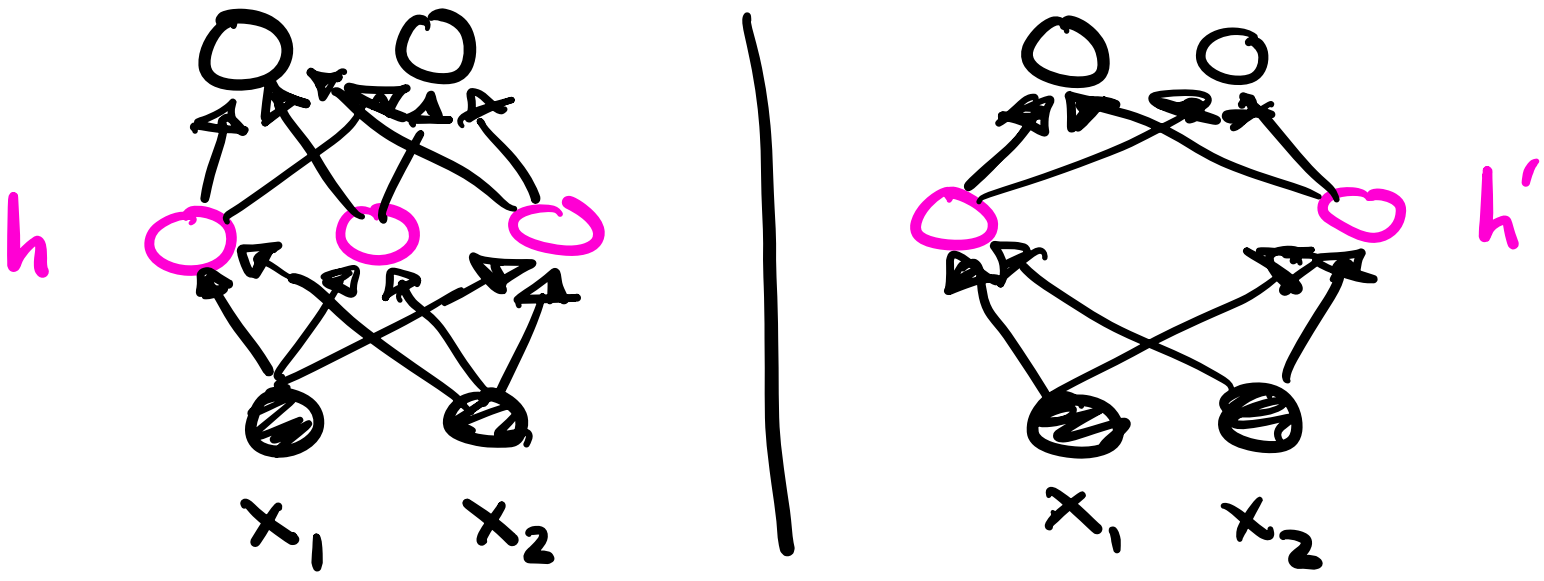
Dropout!

Another standard trick used while training (deep) NETS:

Inject Noise during Training as a regularization Technique.

$$h' = \begin{cases} 0 & \text{with prob. } p \\ \frac{h}{1-p} & \text{otherwise} \end{cases}$$

$$P\left(\frac{h}{1-p}\right) = \frac{ph}{1-p}$$

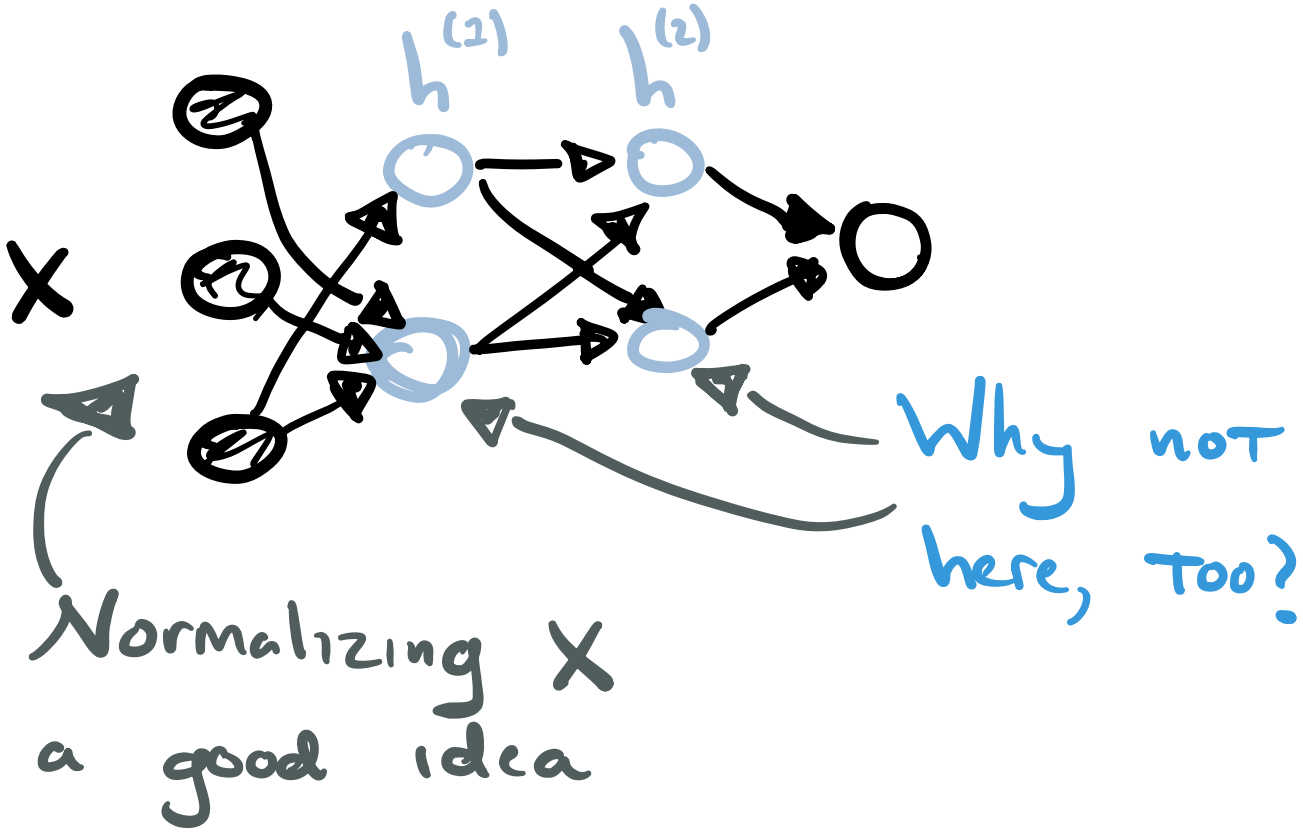


Standard

Dropout

Batch Norm

When we covered optimizers we discussed Normalizing inputs for gradient descent.



With batch norm we do this Per batch.

Assume pre-activation outputs z^l at layer l - so

$$h^l = a(z^l).$$

Then

$$\mu^l = \frac{1}{m}\sum_{j=1}^{m} z_j^l \qquad \sigma^{l^2} = \frac{1}{m}\sum_{j=1}^{m}(\mu^l - z_j^l)^2$$

and

$$z^l_{Norm} = \frac{z^l - \mu^l}{\sqrt{\sigma^{l^2} + \epsilon}}$$

For numerical reasons

0 mean & Unit variance; but BN provides additional flexibility by allowing This to be Shifted

Per layer l

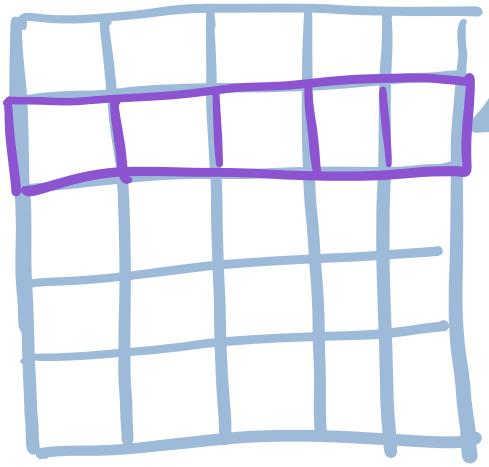
$$z^l_{Norm} = \hat{\gamma}^l \odot \frac{z^l - \mu^l}{\sqrt{\sigma^{l^2} + \epsilon}} + \hat{\beta}^l$$

To be learned.

CNNs for Text!

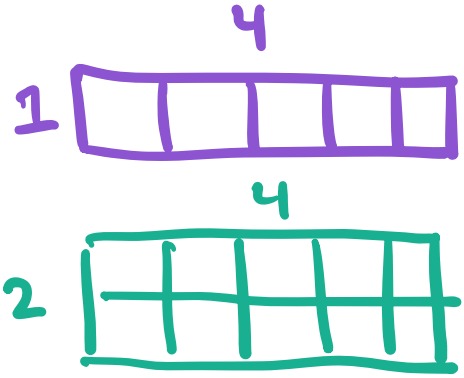
ConvNets mostly used for images

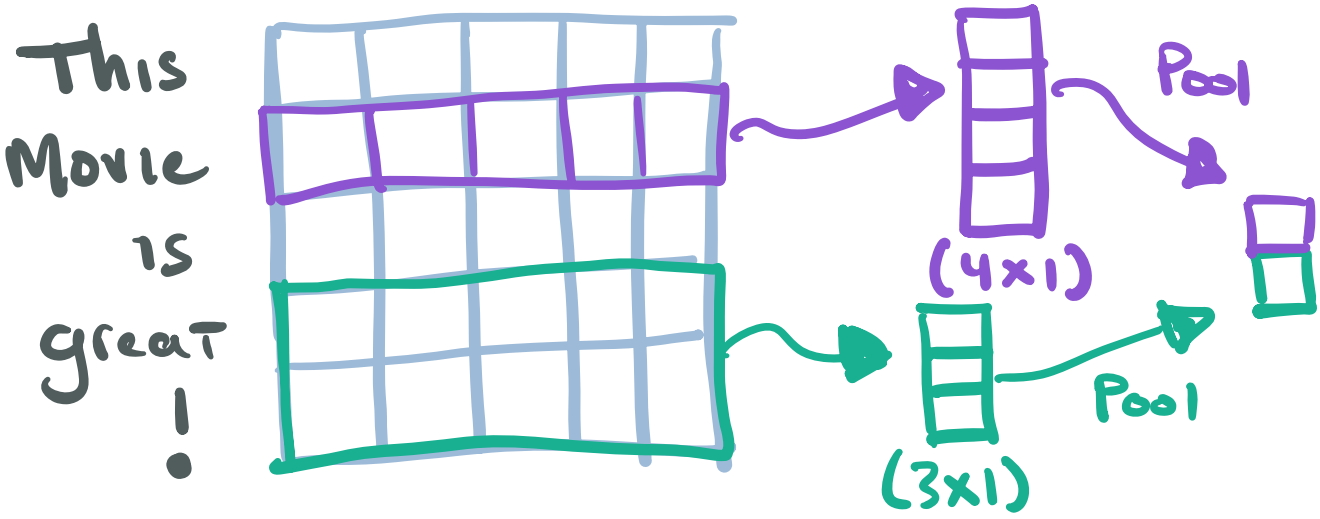
This Movie is great !



Embeddings (9/30 lecture)!

Q. What should our filter size be?





(See Colab)