Practical Deep Learning — Intel Course (WEEK — 02) — CNN

medium.com/@aakashgoel12/practical-deep-learning-intel-course-week-02-cnn-19a1d932793e

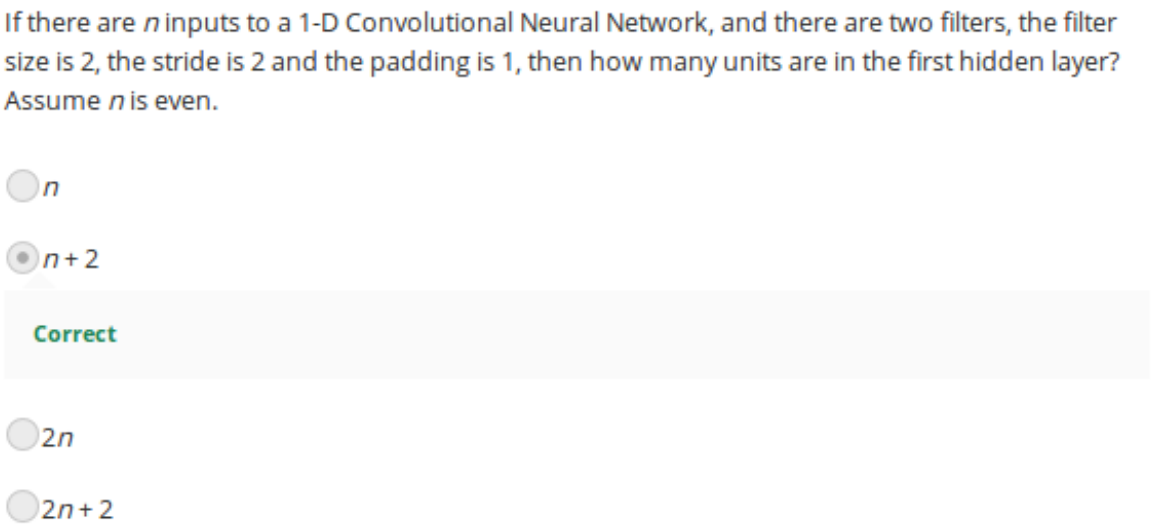
July 13, 2020

> Why CNN and not MLP ?

- Typical image: 256X256 (56,000 pixels), first layer weights (8M Parameters) in MLP — **Unscalable.**
- To take advantage of spacial locality present in images i.e. nearby pixels are more correlated than pixels that are farther away. So, we use this prior knowledge to reduce number of parameters

> 1-D Convolution

- one filter, W1 and W2
- Filter size
- Stride
- Padding — allows us to control the spatial size of the output



Input — n and if include padding @ boundary, input become (n+2). As filter size and stride size is 2, it always take 2 elements of input and move by 2. So, total number of values come out is (n+2)/2. But as no of filters is 2, output will be 2*(n+2)/2.

> Some other points

Pooling — Down sample input of model

Localized max-pooling (stride-2) of the following matrix produces a matrix of what size?

12	20	30	0
8	12	2	0
34	70	37	4
112	100	25	12

- (•) 2 by 2

Correct

- ( ) 2 by 4
- ( ) 4 by 2
- ( ) 4 by 4

What is the largest value in the matrix produced by localized max-pooling (stride-2) of the matrix below?

12	20	30	0
8	12	2	0
34	70	37	4
112	100	25	12

112

Correct Response

What is the smallest value in the matrix produced by localized max-pooling (stride-2) of the matrix below?

12	20	30	0
8	12	2	0
34	70	37	4
112	100	25	12

20

Correct Response

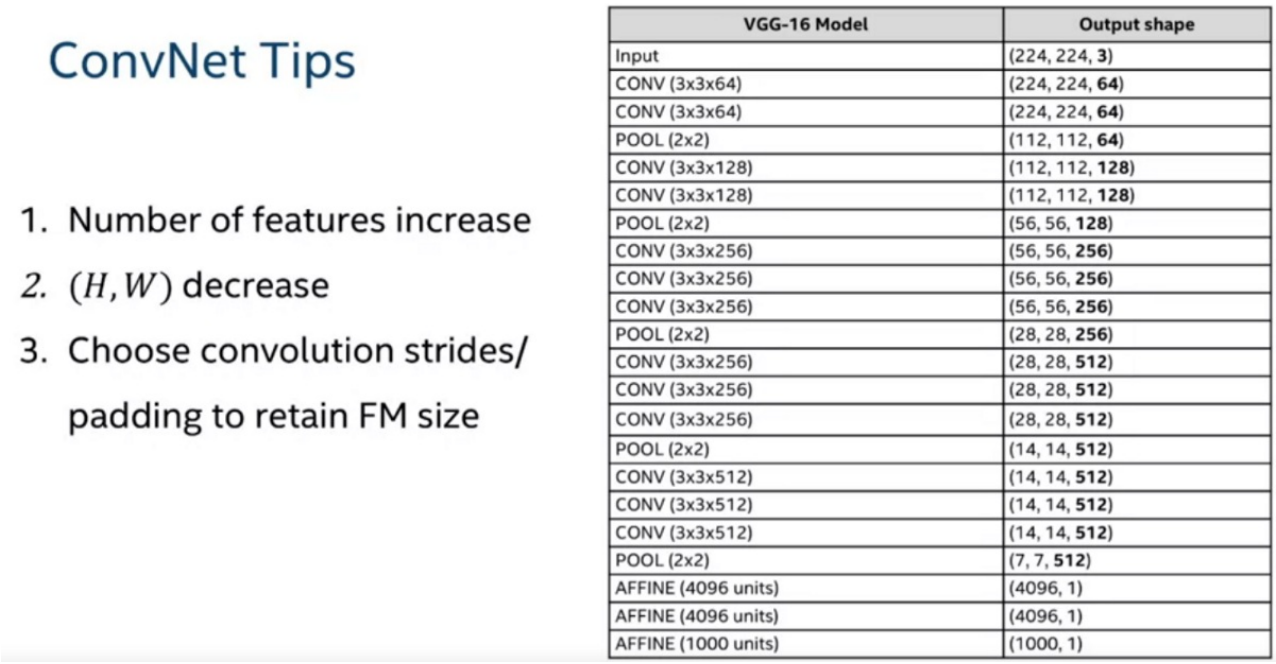
The matrix produced by localized max-pooling (stride-2) is:

20	30
112	37

The largest element is thus 112 and the smallest is 20.

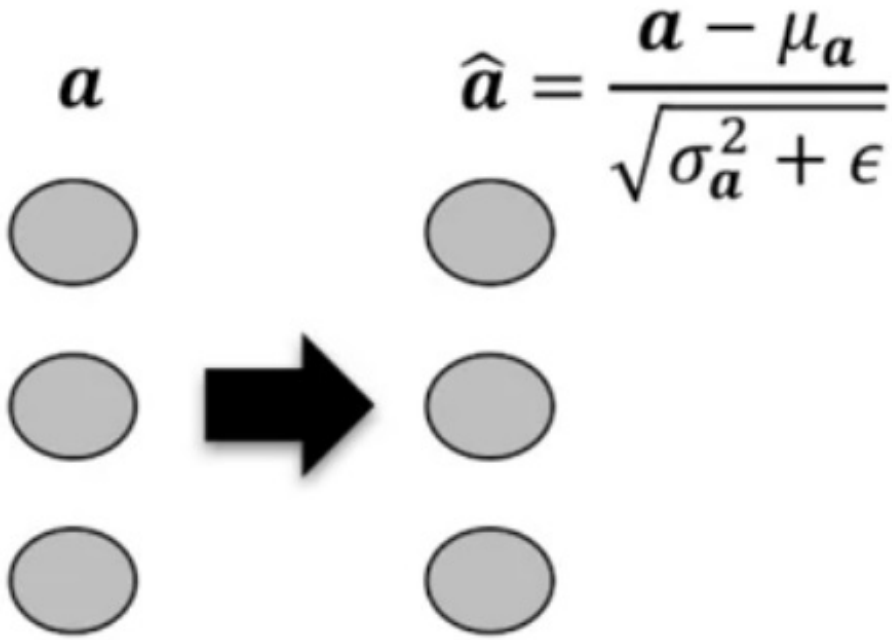
Convnet Tips

Convnet Tips



VGG-16 Model	Output shape
Input	(224, 224, **3**)
CONV (3x3x64)	(224, 224, **64**)
CONV (3x3x64)	(224, 224, **64**)
POOL (2x2)	(112, 112, **64**)
CONV (3x3x128)	(112, 112, **128**)
CONV (3x3x128)	(112, 112, **128**)
POOL (2x2)	(56, 56, **128**)
CONV (3x3x256)	(56, 56, **256**)
CONV (3x3x256)	(56, 56, **256**)
CONV (3x3x256)	(56, 56, **256**)
POOL (2x2)	(28, 28, **256**)
CONV (3x3x256)	(28, 28, **512**)
CONV (3x3x256)	(28, 28, **512**)
CONV (3x3x256)	(28, 28, **512**)
POOL (2x2)	(14, 14, **512**)
CONV (3x3x512)	(14, 14, **512**)
CONV (3x3x512)	(14, 14, **512**)
CONV (3x3x512)	(14, 14, **512**)
POOL (2x2)	(7, 7, **512**)
AFFINE (4096 units)	(4096, 1)
AFFINE (4096 units)	(4096, 1)
AFFINE (1000 units)	(1000, 1)

- **Dropout** — During training, it will **ignore** a **fraction of units**, and that selection is is going to be **randomized** from mini **batch** to min batch. It prevents model from relying on specific features of individual units to drive the output (prevents co-adaptation). Instead it must rely on a distribution of units. **Other way,** because at each mini-batch a different model, slightly different model is being trained because you're silencing different portions of a unit. In that way, you're training different model from mini-batch to mini -batch and mirror like **ensemble methods** that we use in ML to combine information from multiple models together.
- **Batch Normalization** — It allows network to converge faster and achieve lower error. At end of every batch norm layer, output of all neurons are normalized to have zero mean and unit variance. This allows the network to be more robust to bad initialization and reduces internal co-variance shift.



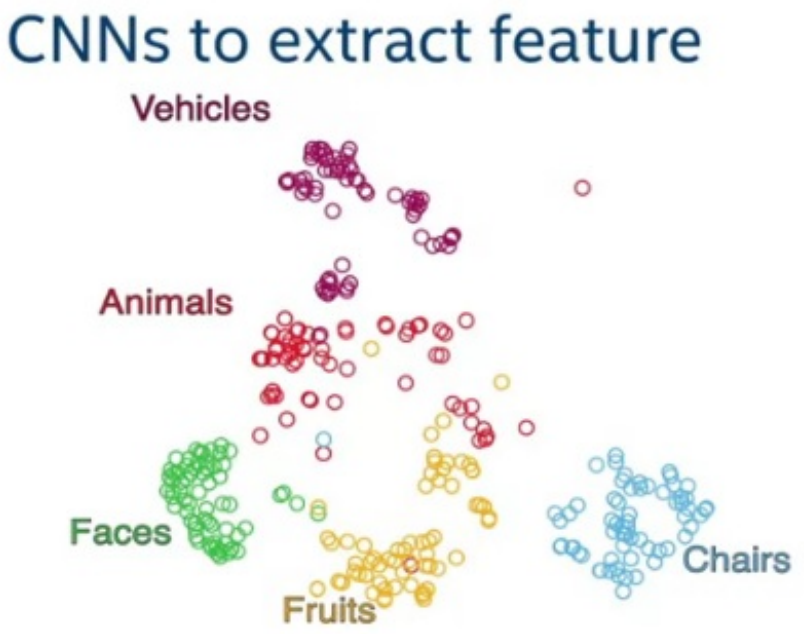
> Additional reading resource

- *Visualizing and Understanding Convolutional Networks*: https://arxiv.org/abs/1311.2901
- *Dropout: A Simple Way to Prevent Neural Networks from Overfitting* : http://www.jmlr.org/papers/volume15/srivastava14a/srivastava14a.pdf
- *Very Deep Convolutional Networks for Large-Scale Image Recognition*: https://arxiv.org/abs/1409.1556
- *Going Deeper with Convolutions*: https://arxiv.org/abs/1409.4842
- *Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift*: https://arxiv.org/abs/1502.03167
- *Delving Deep into Rectifiers: Surpassing Human-Level Performance on ImageNet Classification*: https://arxiv.org/abs/1502.01852

> Fine Tuning and detection

- Earlier layers in CNN act as edge and block detector and later layers identify more complex structures such as object parts
- Images are not linearly separable in pixel space but they may be linearly separable in feature separable

Pixel Space



Feature Space

When extracting features from a Convolutional Neural Network, in what case would a feature representation be useful?

- Images are not linearly separable in the pixel space, and are not linearly separable in the feature space
- Images may be linearly separable in the pixel space, but are not linearly separable in the feature space
- Images are not linearly separable in the pixel space, but may be linearly separable in the feature space

Correct

- Images may be linearly separable in the pixel space, and may be linearly separable in the feature space

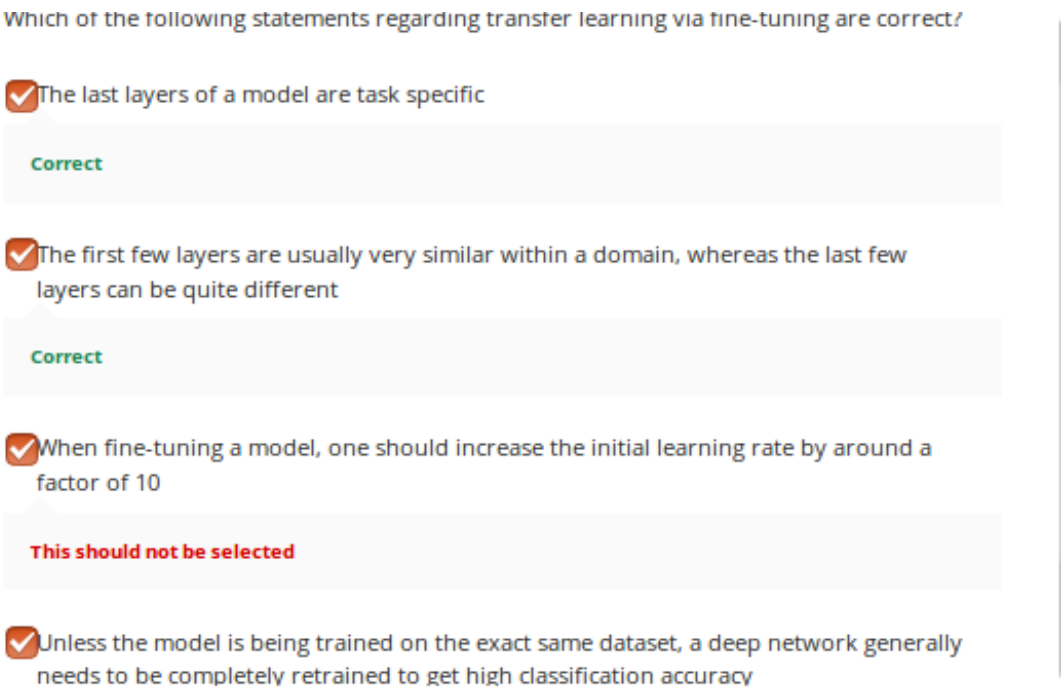
- A good feature representation can be used as input to another machine learning classifier. CNNs can be used to extract features from an image which in turn can be used to generate caption using a language generated RNN
- **Transfer Learning —** TL via fine tuning can be employed using feature representation. Pre train model learns mapping from pixel to feature space and as a result majority of network doesn't have to be re-trained. Instead last layer or layers can be modified and re-learned with smaller dataset in order to fie tune the network to desired application

When fine-tuning a Convolutional Neural Network, which of the following techniques is most effective?

- Start by training a network with a very large dataset, and then modify and re-learn the first layer(s) with a smaller dataset after the network learns a mapping from pixels to a feature space
- Start by training a network with a very large dataset, and then modify and re-learn the last layer(s) with a smaller dataset after the network learns a mapping from pixels to a feature space

Correct

- Start by training a network with a small dataset, and then modify and re-learn the first layer(s) with a very large dataset after the network learns a mapping from pixels to a feature space



Last option is also not selected.

> QUIZ

1. Which of the following can be said about convolutional layers as opposed to full connected layers?

   - [ ] Convolutional layers do not take into account spatial information in input data, while fully connected layers do
   - [x] In general, convolutional layers have less parameters than a corresponding fully connected layer
   - [ ] A fully connected layer can be treated as a convolutional layer with one filter, no matter the filter size.
   - [x] A fully connected layer can be treated as a convolutional layer with one filter with filter size equal to the input size

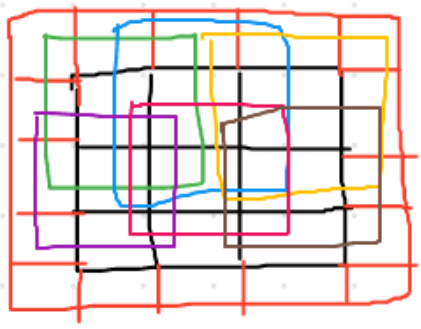
2. Given a convolutional layer whose input is *n* by *n* with a padding of 1 and stride of 1, what filter size must be used to maintain the size of the input (so that the output is also *n* by *n*)?

   n+1

3. When extracting features from a Convolutional Neural Network, in what case would a feature representation be useful?

   - ( ) Images are not linearly separable in the pixel space, and are not linearly separable in the feature space
   - ( ) Images may be linearly separable in the pixel space, but are not linearly separable in the feature space
   - (•) Images are not linearly separable in the pixel space, but may be linearly separable in the feature space
   - ( ) Images may be linearly separable in the pixel space, and may be linearly separable in the feature space

2nd answer isn't correct... Answer is may be 3X3



4. When fine-tuning a Convolutional Neural Network, which of the following techniques is most effective?

- ○ Start by training a network with a very large dataset, and then modify and re-learn the first layer(s) with a smaller dataset after the network learns a mapping from pixels to a feature space
- ◉ Start by training a network with a very large dataset, and then modify and re-learn the last layer(s) with a smaller dataset after the network learns a mapping from pixels to a feature space
- ○ Start by training a network with a small dataset, and then modify and re-learn the first layer(s) with a very large dataset after the network learns a mapping from pixels to a feature space
- ○ Start by training a network with a small dataset, and then modify and re-learn the last layer(s) with a very large dataset after the network learns a mapping from pixels to a feature space

5. Which of the following statements is true regarding CNN architectures?

- ☐ Generally, a CNN begins with a few fully connected layers followed by alternating convolutional and pooling layers
- ☑ Generally, a CNN begins with alternating convolutional and pooling layers, followed by a few fully connected layers
- ☐ With each layer in the network, the number of filters (the depth) generally decreases
- ☑ With each layer in the network, the input size to a layer generally decreases

6. Localized max-pooling (stride-2) of the following matrix produces a matrix of what size? Enter your answer as "# by #"

11	4	7	12
3	1	4	19
33	54	17	24
0	43	2	8

2 by 2

7. What is the largest value produced by localized max-pooling (stride-2) of the following matrix?

11	4	7	12
3	1	4	19
33	54	17	24
0	43	2	8

54