Lecture 3: computer vision, convolutional networks, transfer learning

Machine Learning - 2

Data Science and Business Analytics Program (DSBA) at HSE & LSE, 22/23

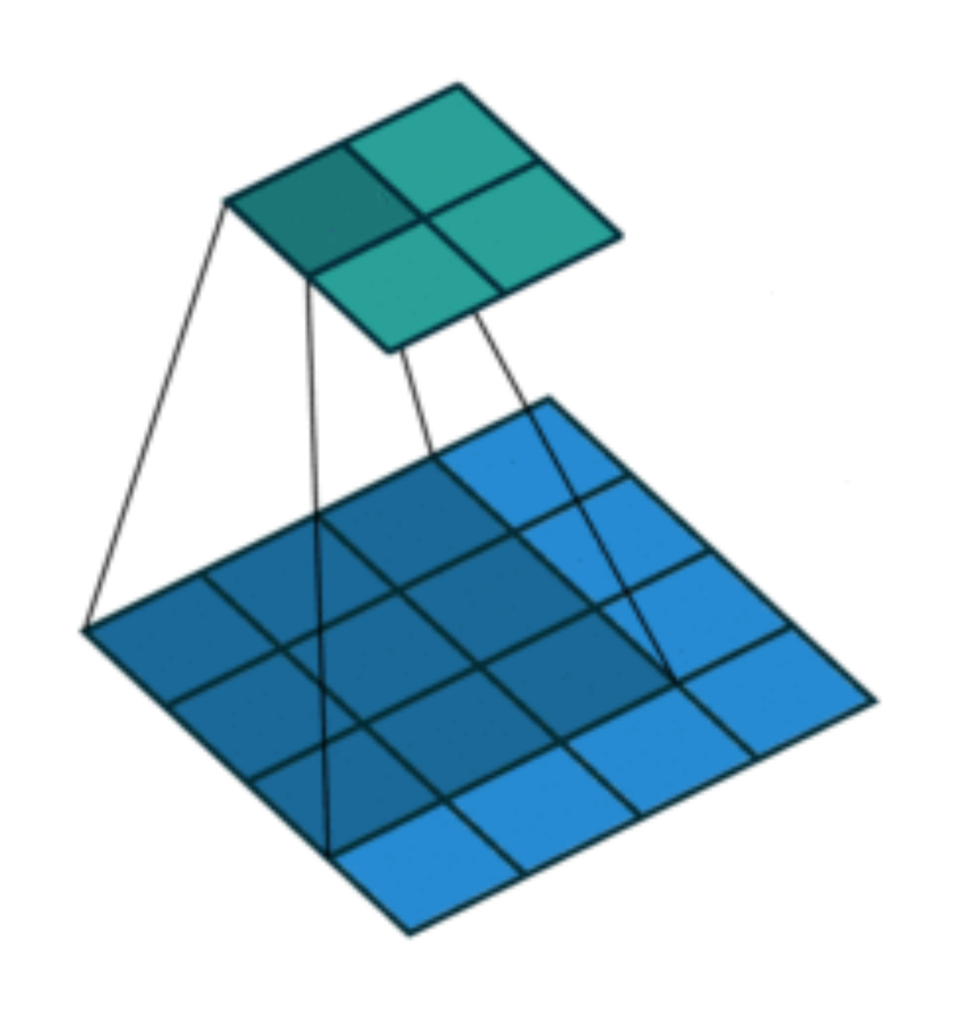
Akim Tsvigun

Convolutions, Pooling, Computer Vision Tasks

Source:

- https://stanford.edu/~shervine/teaching/cs-230/cheatsheet-convolutional-neural-networks
- https://stanford.edu/~shervine/teaching/cs-230/cheatsheet-deep-learning-tips-and-tricks
- https://cs231n.github.io/convolutional-networks/

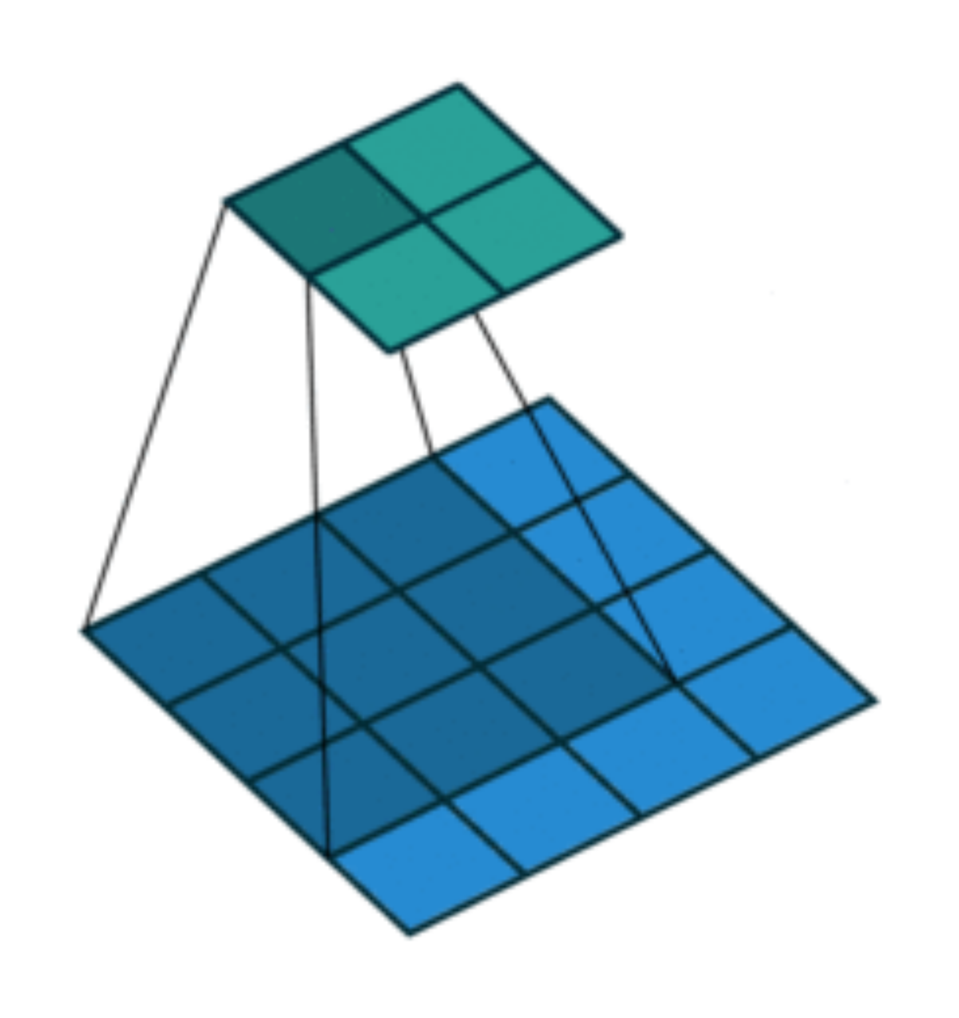
Convolution examples



F - ?
Padding - ?
Stride - ?

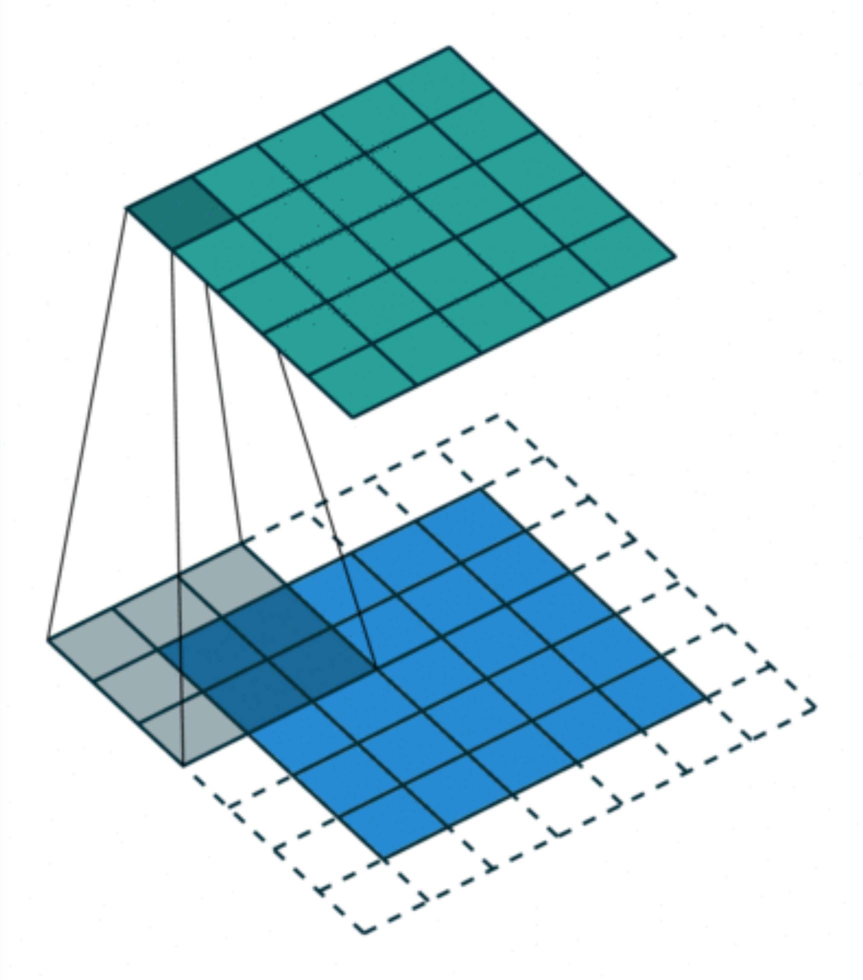
NATIONAL RESEARCH UNIVERSITY

Convolution examples



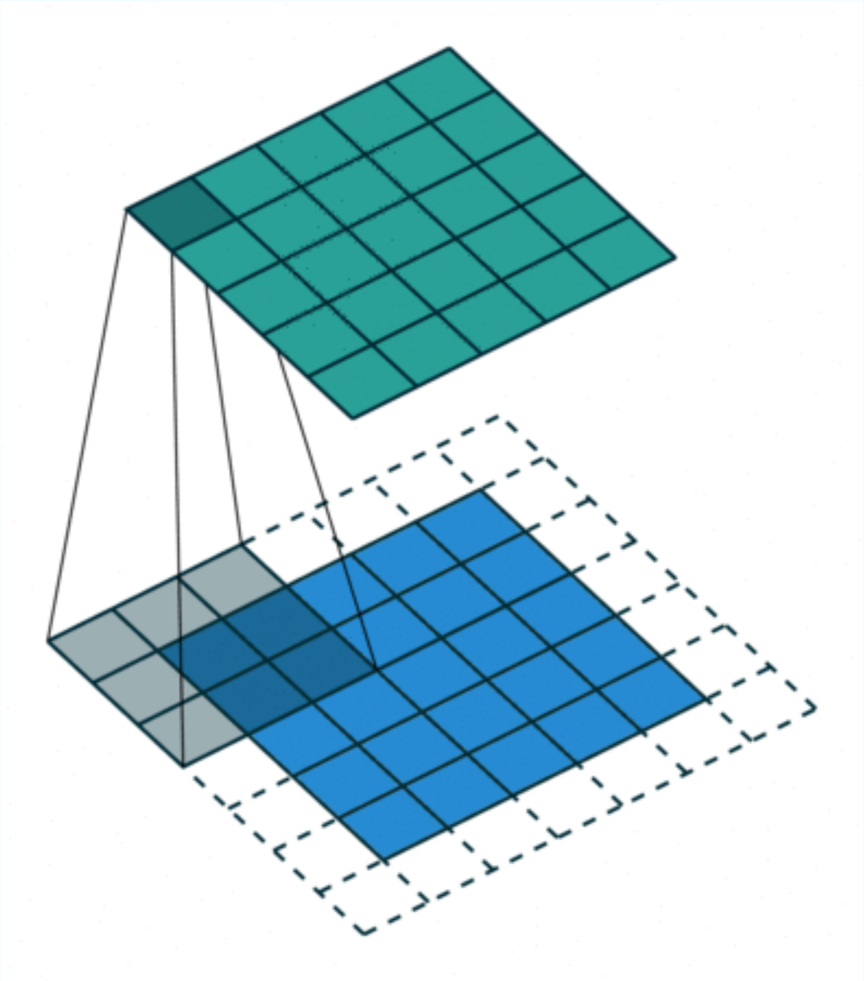
F = 3
Padding = 0
Stride = 1

Convolution examples



F - ?
Padding - ?
Stride - ?

Convolution examples



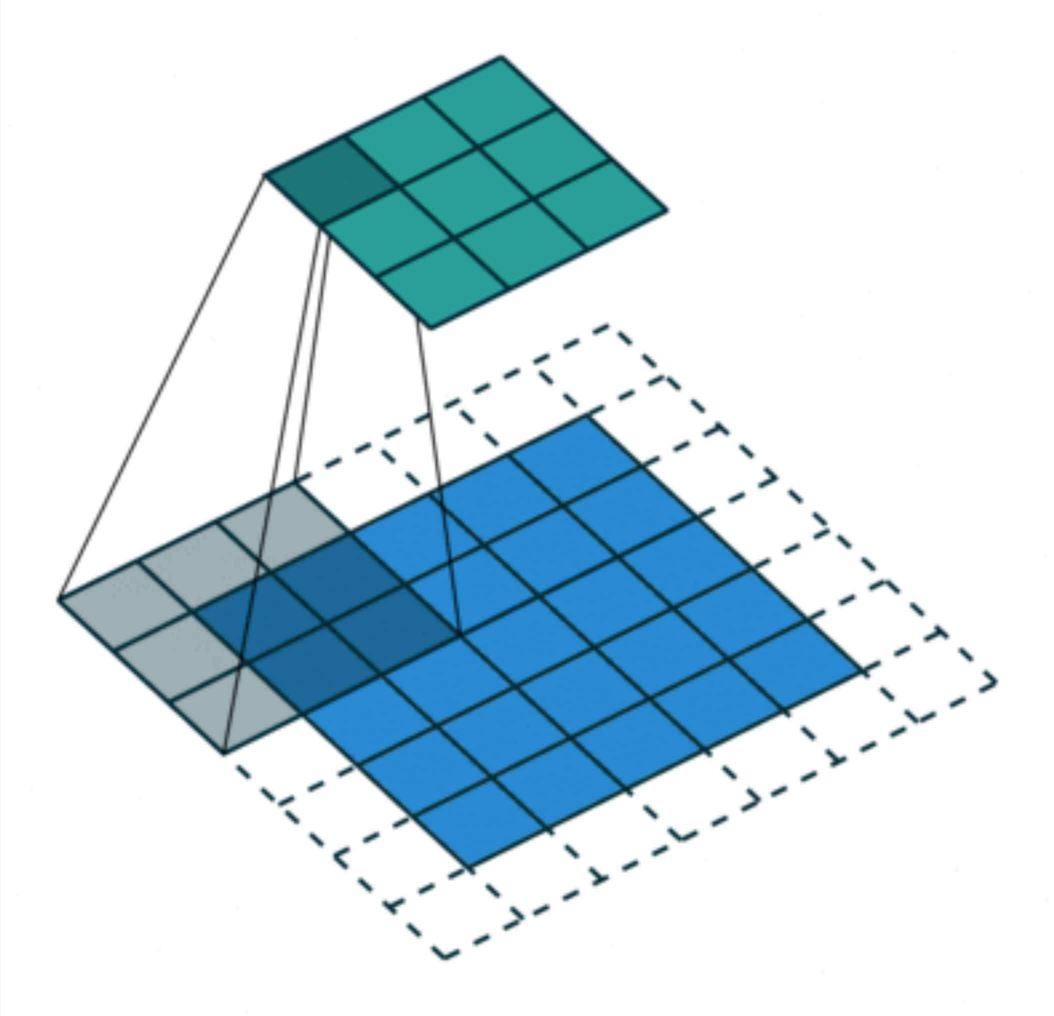
F = 3

Padding = same (half)

Stride = 1

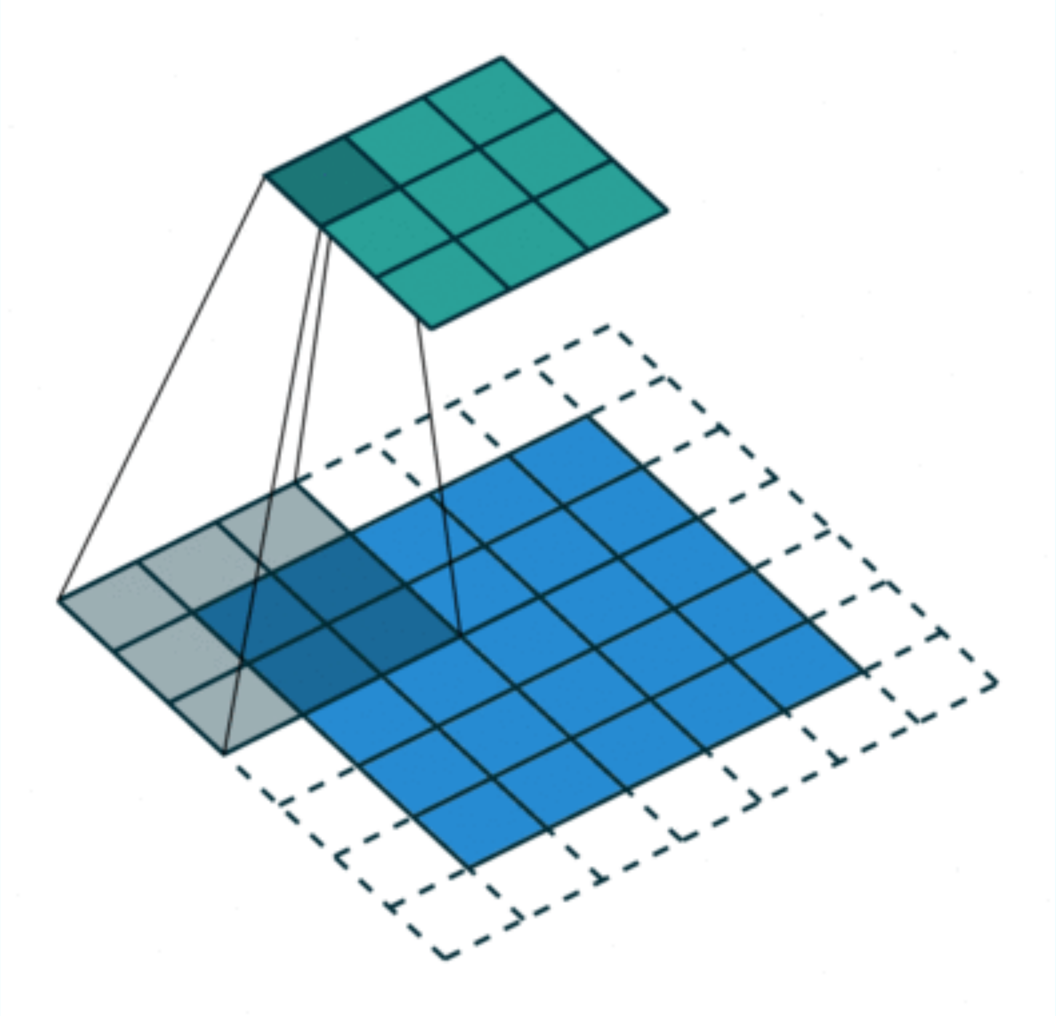
NATIONAL RESEARCH UNIVERSITY

Convolution examples



F - ?
Padding - ?
Stride - ?

Convolution examples

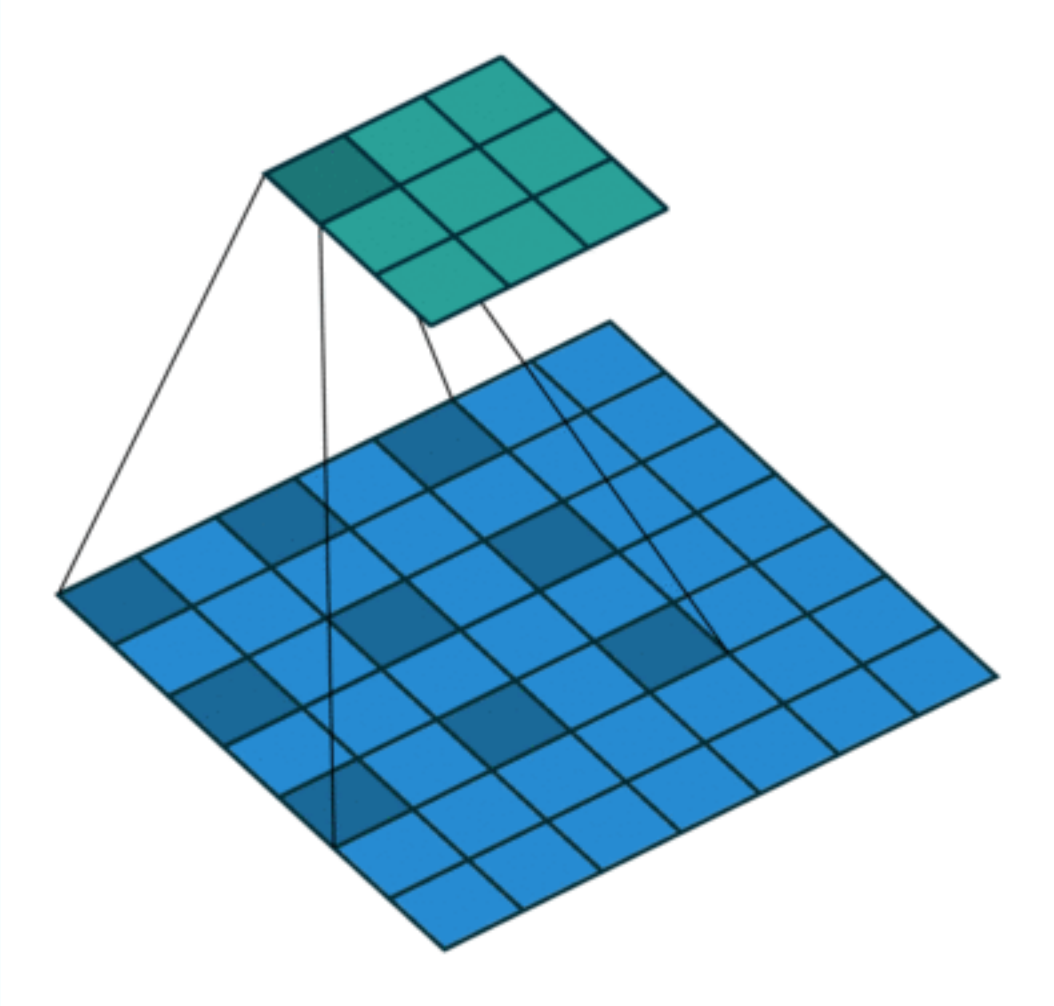


F = 3

Padding = same (half)

Stride = 2

Convolution examples

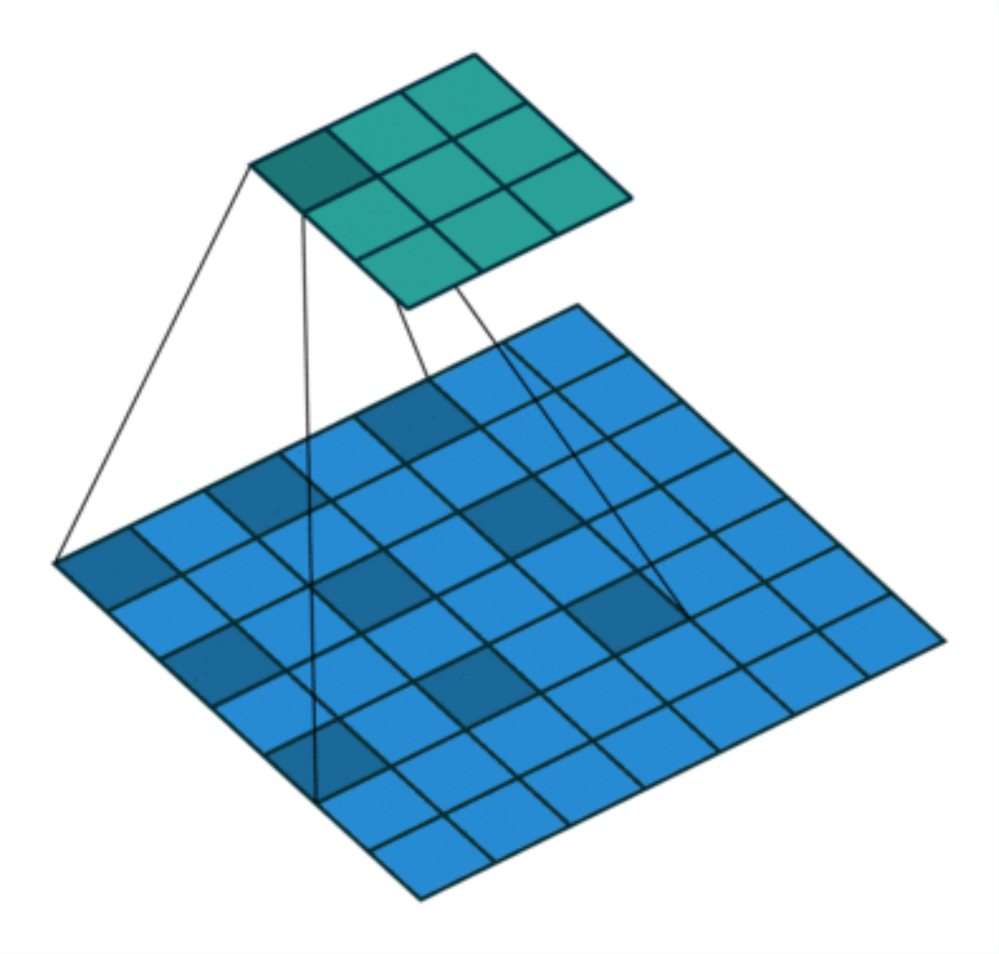


F - ?

Padding - ?

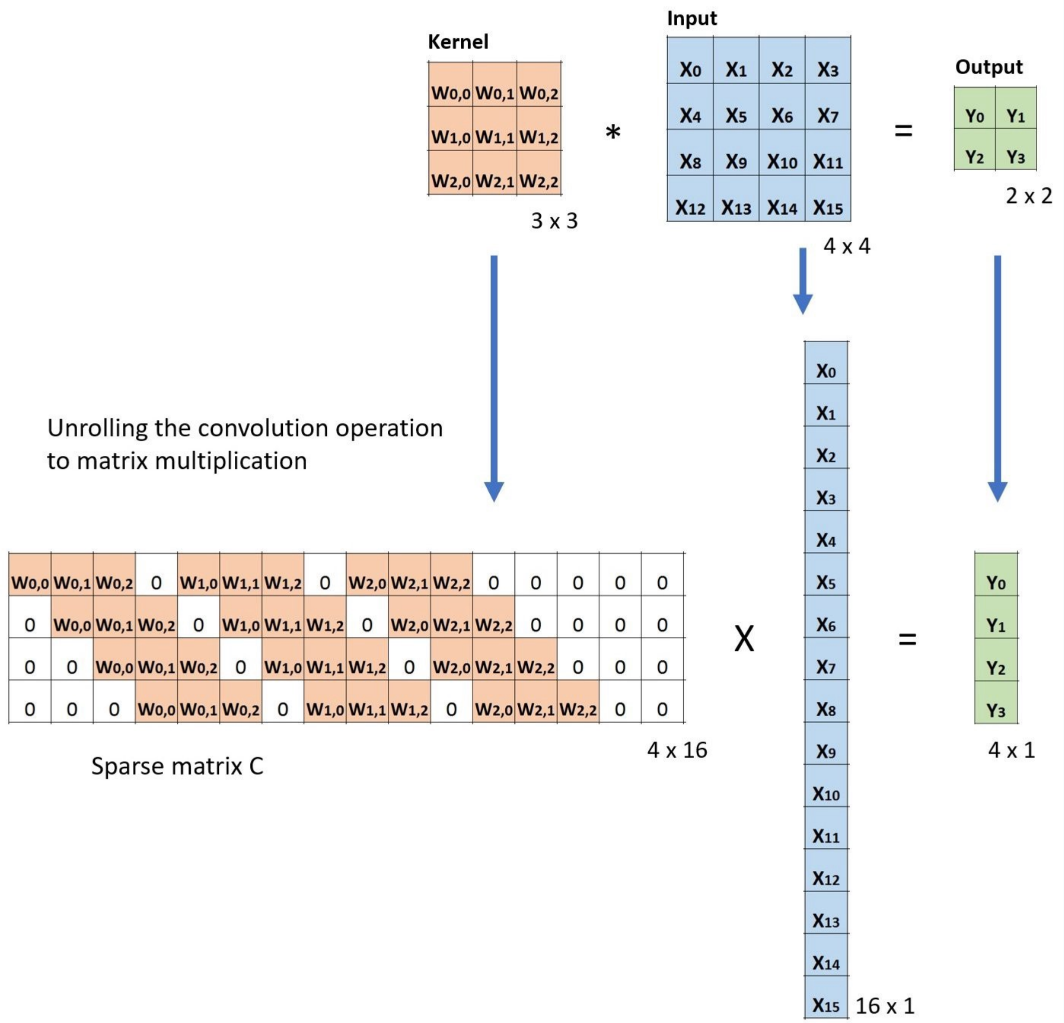
Stride - ?

Convolution examples



F = 3
Padding = 0
Stride = 1
Dilation = 2

Convolution as matrix multiplication



- Construct a sparse matrix which can be directly multiplied!

NATIONAL RESEARCH UNIVERSITY

Data Augmentation

- Train set:

- Test instance:

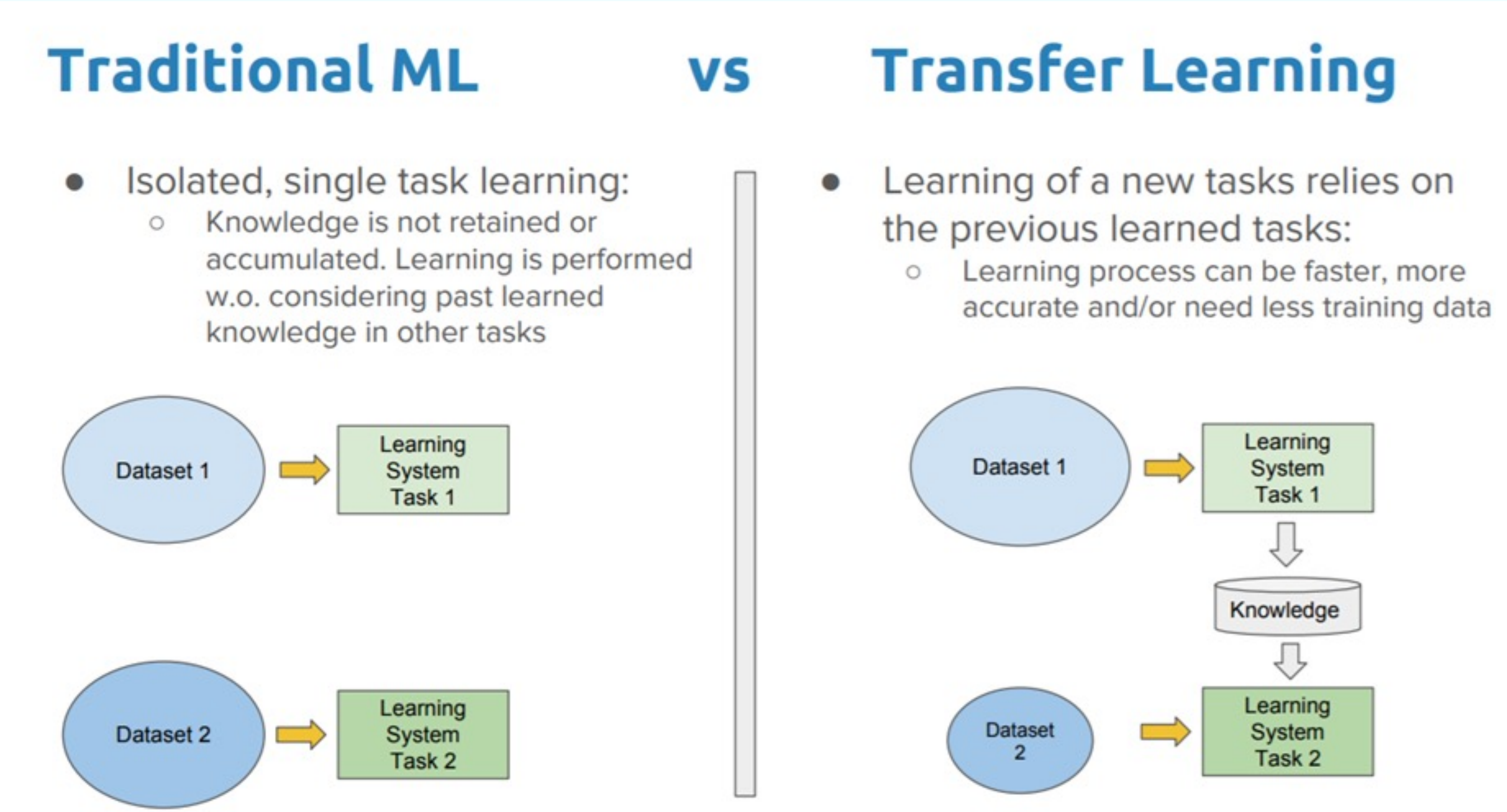
- Which class?

Data Augmentation

Source:

- **https://stanford.edu/~shervine/teaching/cs-230/cheatsheet-deep-learning-tips-and-tricks**

Transfer Learning & Fine-Tuning



Transfer Learning & Fine-Tuning

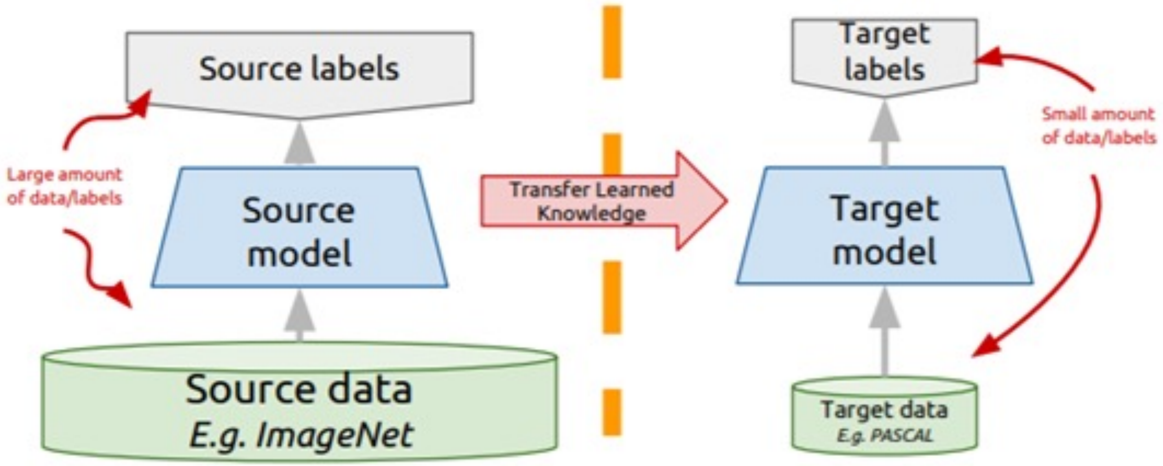
Transfer learning: idea

Instead of training a deep network from scratch for your task:

- Take a network trained on a different domain for a different **source task**
- Adapt it for your domain and your **target task**

Variations:

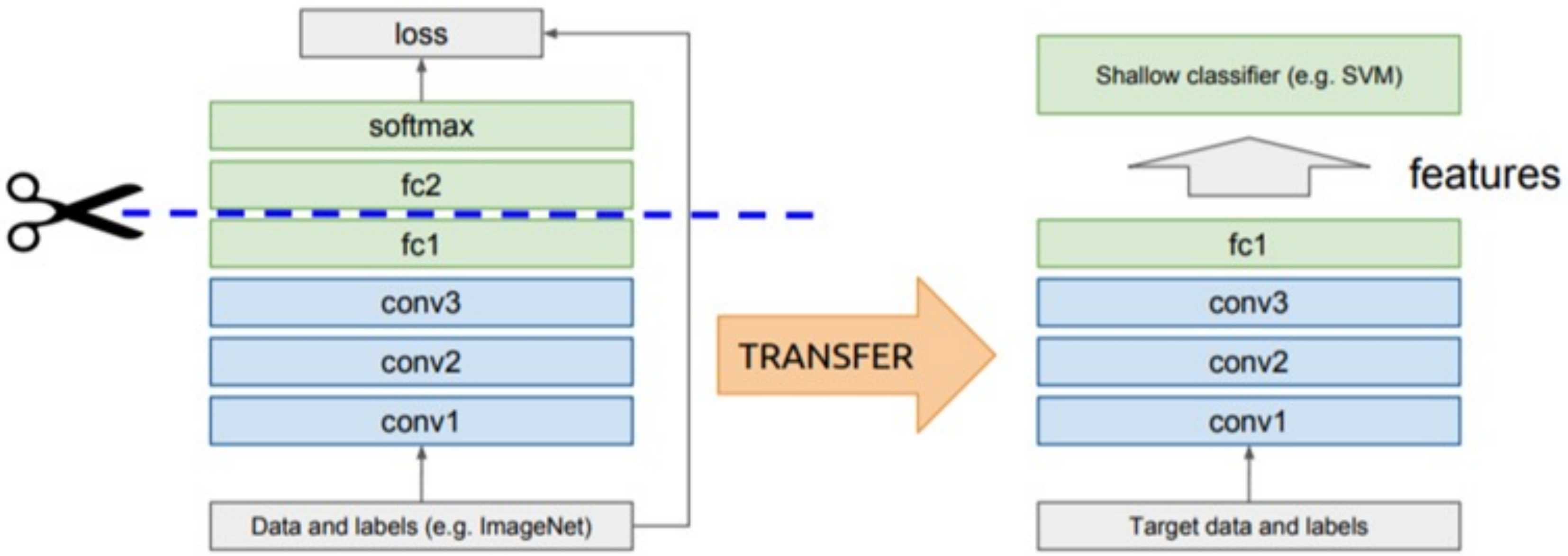
- Same domain, different task
- Different domain, same task



Transfer Learning & Fine-Tuning

Idea: use outputs of one or more layers of a network trained on a different task as generic feature detectors. Train a new shallow model on these features.

Assumes that $D_S = D_T$



Transfer Learning & Fine-Tuning

Freeze or fine-tune?

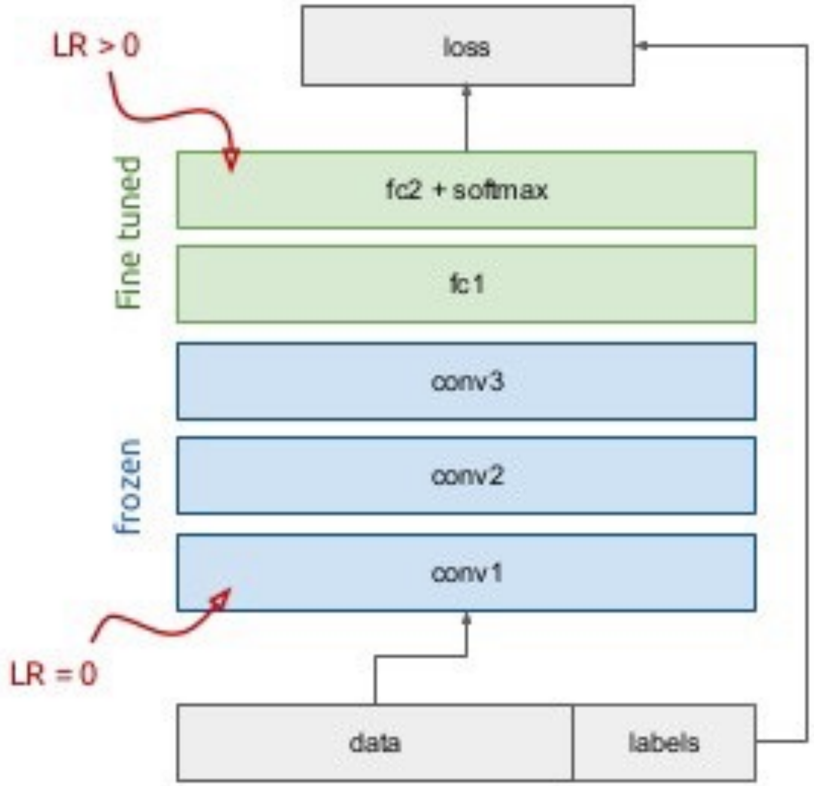
Bottom *n* layers can be frozen or fine tuned.

- **Frozen**: not updated during backprop
- **Fine-tuned**: updated during backprop

Which to do depends on target task:

- **Freeze**: target task labels are scarce, and we want to avoid overfitting
- **Fine-tune**: target task labels are more plentiful

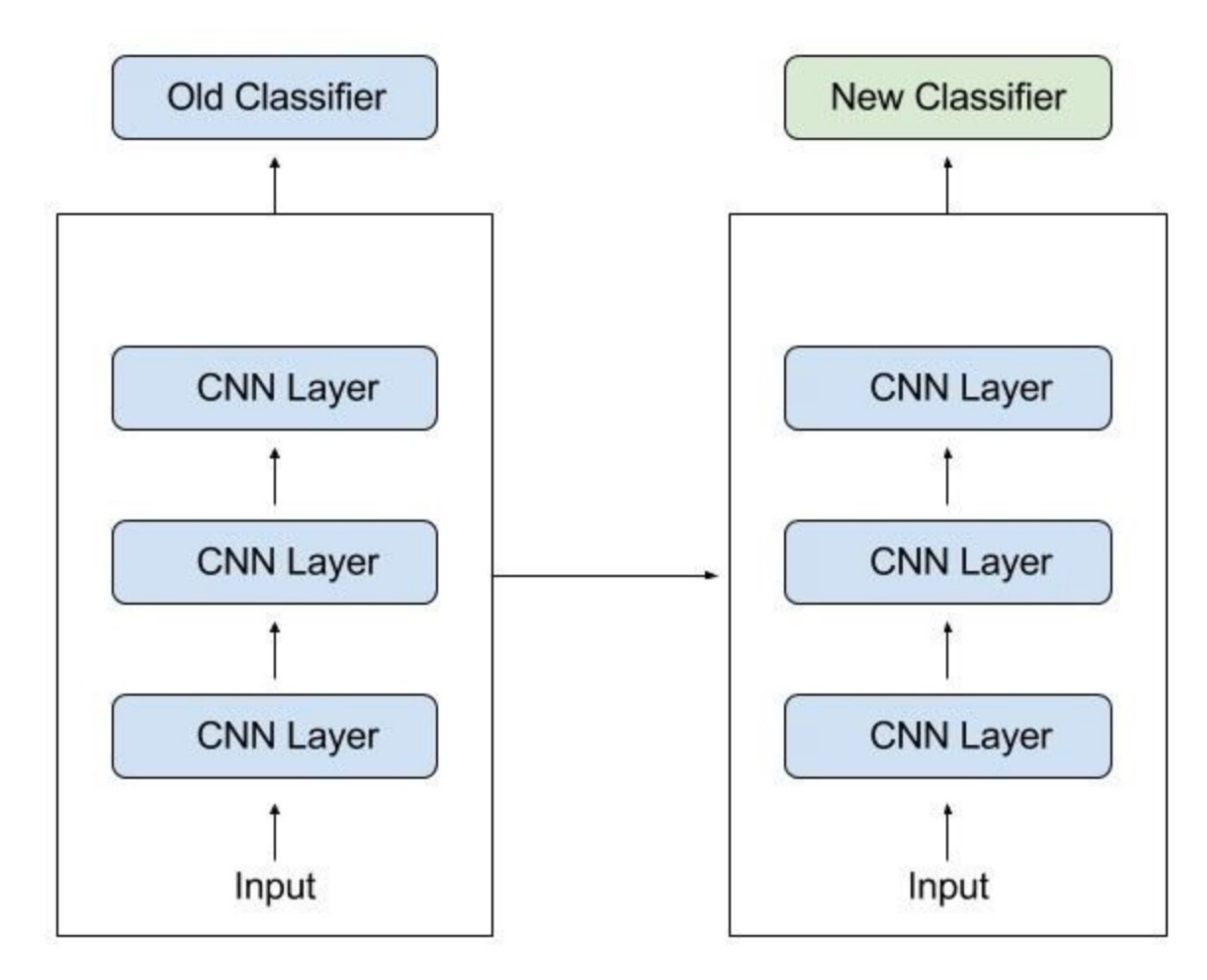
In general, we can set learning rates to be different for each layer to find a tradeoff between freezing and fine tuning



10

NATIONAL RESEARCH UNIVERSITY

Transfer Learning & Fine-Tuning



NATIONAL RESEARCH UNIVERSITY

Self-learning

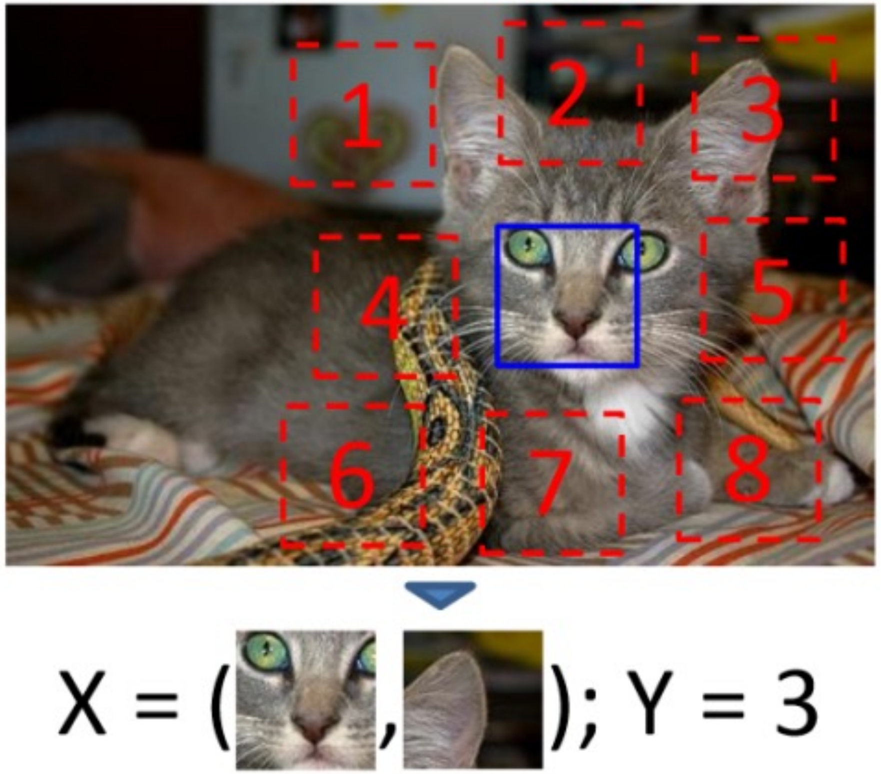
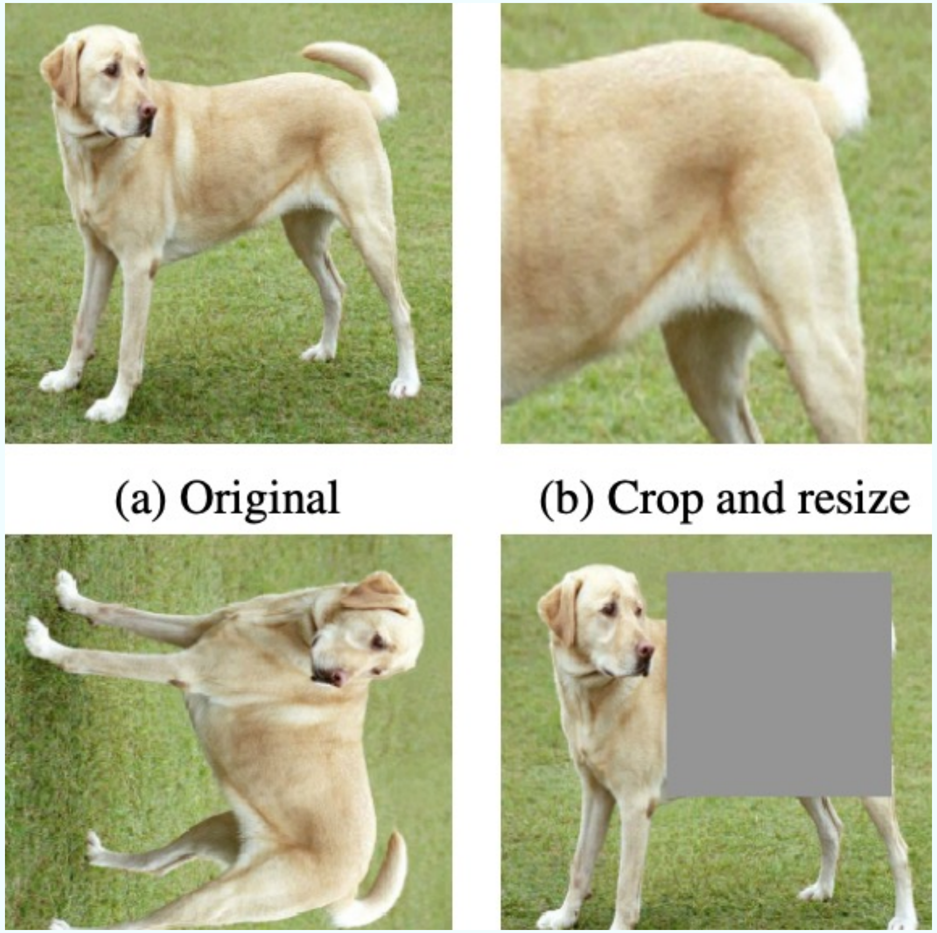


Figure 2. The algorithm receives two patches in one of these eight possible spatial arrangements, without any context, and must then classify which configuration was sampled.



NATIONAL RESEARCH UNIVERSITY

Summary

- Convolutional networks are mostly used in Computer Vision due to the benefits of convolutional & pooling layers
- Data augmentation allows to increase the performance of the model
- Transfer learning & fine-tuning significantly improve the performance of the models in CV / NLP by leveraging knowledge obtained while solving a large pre-training task onto the target task

AIRI

airi.net