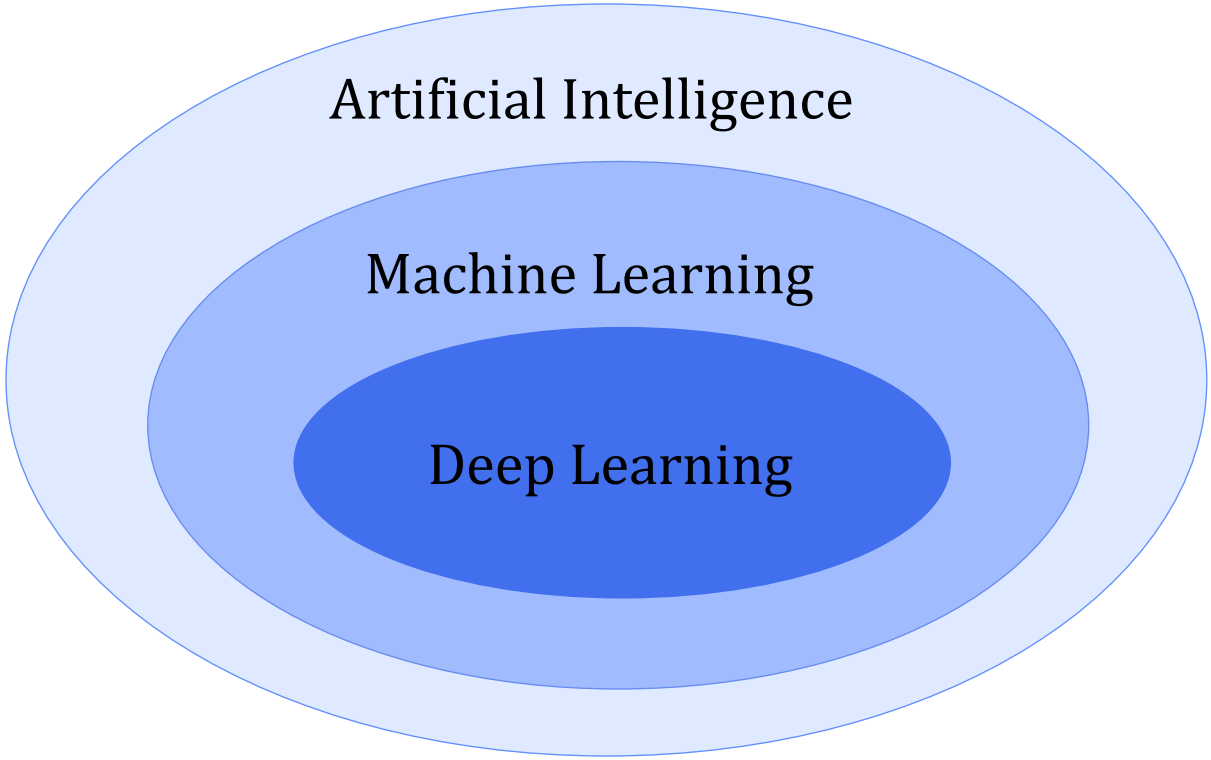
2018 Summer Institute
Deep Learning

SDSC SAN DIEGO SUPERCOMPUTER CENTER

UC San Diego

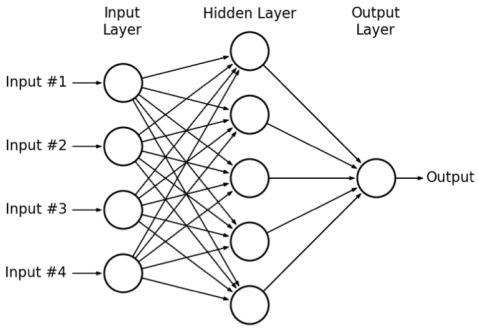
Deep Learning



Deep Learning is a subfield of Machine Learning

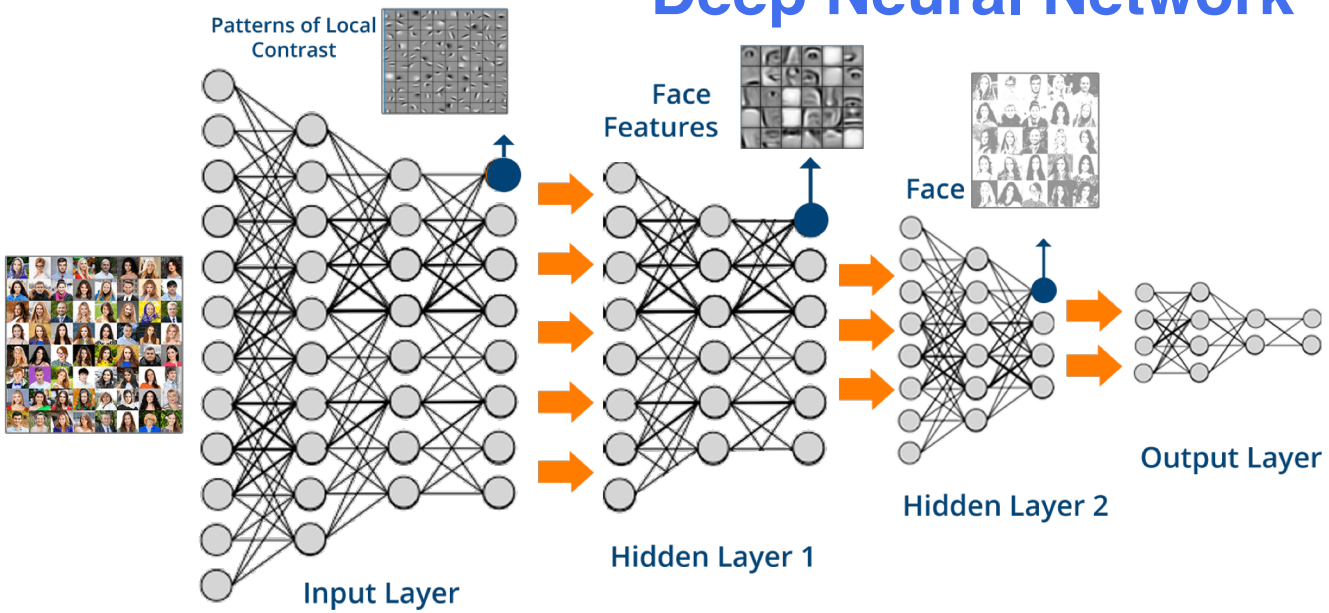
Deep Learning

Neural Network



http://www.astroml.org/book_figures/appendix/fig_neural_network.html

Deep Neural Network



https://cdn.edureka.co/blog/wp-content/uploads/2017/05/Deep-Neural-Network-What-is-Deep-Learning-Edureka.png

- **'Deep' refers to the many layers in model**
  - Allows for learning at different levels of abstraction
  - Leads to automatic feature learning & excellent performance

Applications of Deep Learning

- Image classification
- Speech recognition
- Handwriting recognition
- Self-driving cars
- Drug design
- Precision medicine
- Disease detection
- Targeted ads
- Stock market analysis

Deep Learning Agenda

8:30 - 9:00 – Intro to NN & CNN

9:00 - 9:30 – MNIST Hands-On

9:30 - 10:00 – CNN Transfer Learning

10:00 - 10:15 – Break

10:15 - 10:45 – CNN Transfer Learning (cont.)

10:45 - 11:15 – FasterCNN

11:15 - 11:45 – U-Net & LSTM

11:45 - 12:00 – Wrap-Up

Transfer Learning with CNN

Mai H. Nguyen, Ph.D.

What is Transfer Learning?

- **To overcome challenges of training model from scratch:**
  - Insufficient data
  - Very long training time
- **Use pre-trained model**
  - Trained on another dataset
  - This serves as starting point for model
  - Then train model on current dataset for current task

Transfer Learning Approaches

- **Feature extraction**
  - Remove last fully connected layer from pre-trained model
  - Treat rest of network as feature extractor
  - Use features to train new classifier (“top model”)
- **Fine tuning**
  - Tune weights in some layers of original model (along with weights of top model)
  - Train model for current task using new dataset

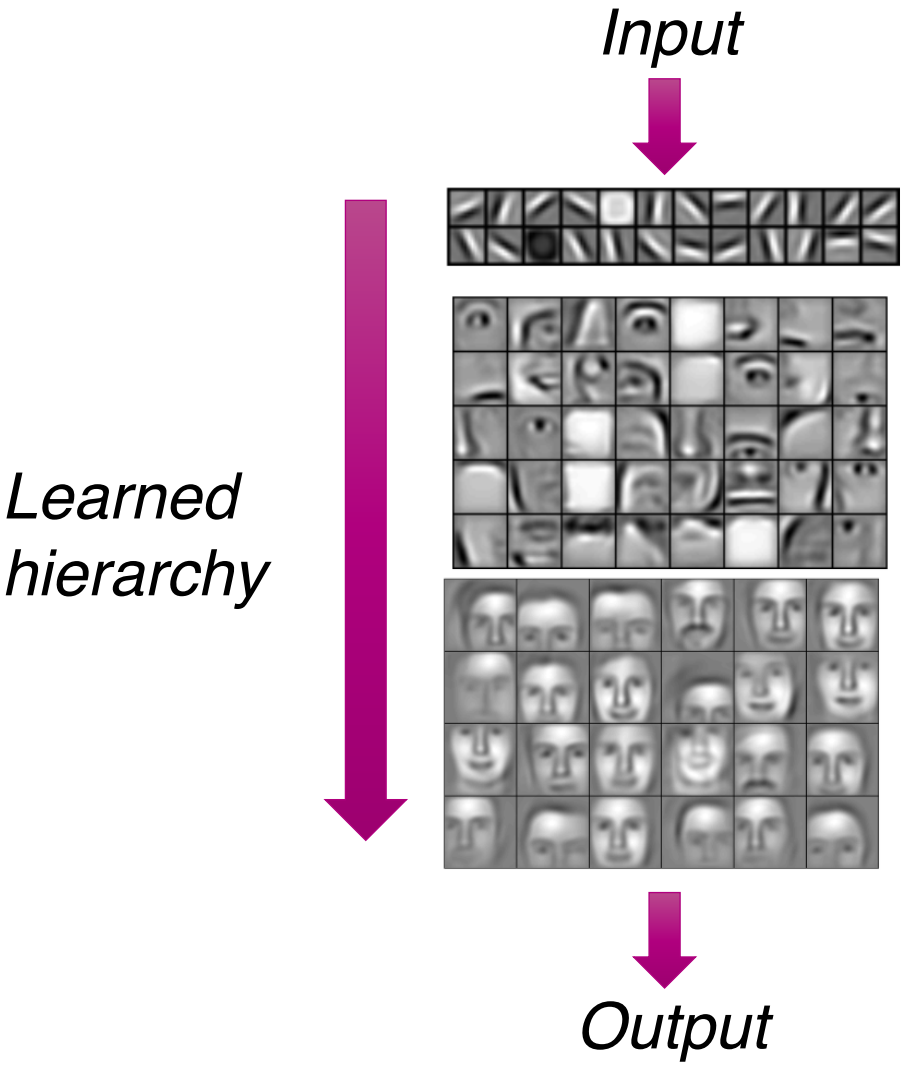
CNNs for Transfer Learning

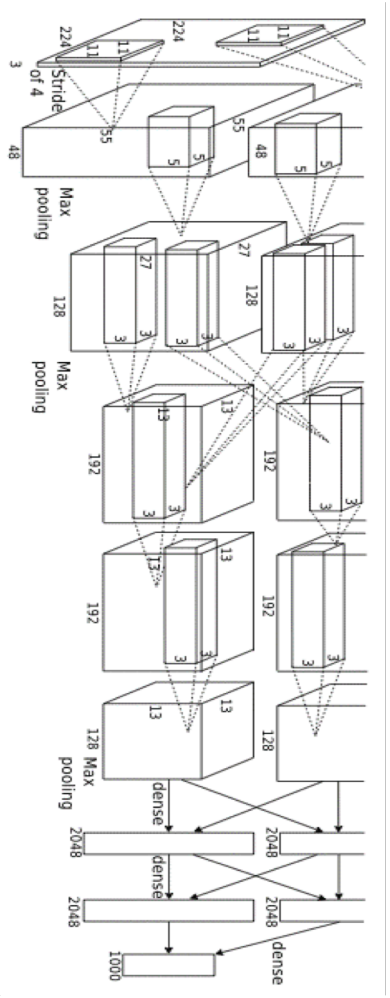
- **Popular architectures**
  - AlexNet
  - GoogLeNet
  - VGGNet
  - ResNet
- **All winners of ILSVRC**
  - ImageNet Large Scale Visual Recognition Challenge
  - Annual competition on vision tasks on ImageNet data

ImageNet

- **Database**
  - Developed for computer vision research
  - > 14,000,000 images hand-annotated
  - > 22,000 categories
- **ILSVRC History**
  - Started in 2010
  - Image classification task: 1,000 object categories
  - Image classification error rate
    - 2011: ~25% (conventional image processing techniques)
    - 2012: 15.3% (AlexNet)
    - 2015: 3.57% (ResNet; better than human performance)
    - 2016: 2.99% (16.7% error reduction)
    - 2017: 2.25% (23.3% error reduction)

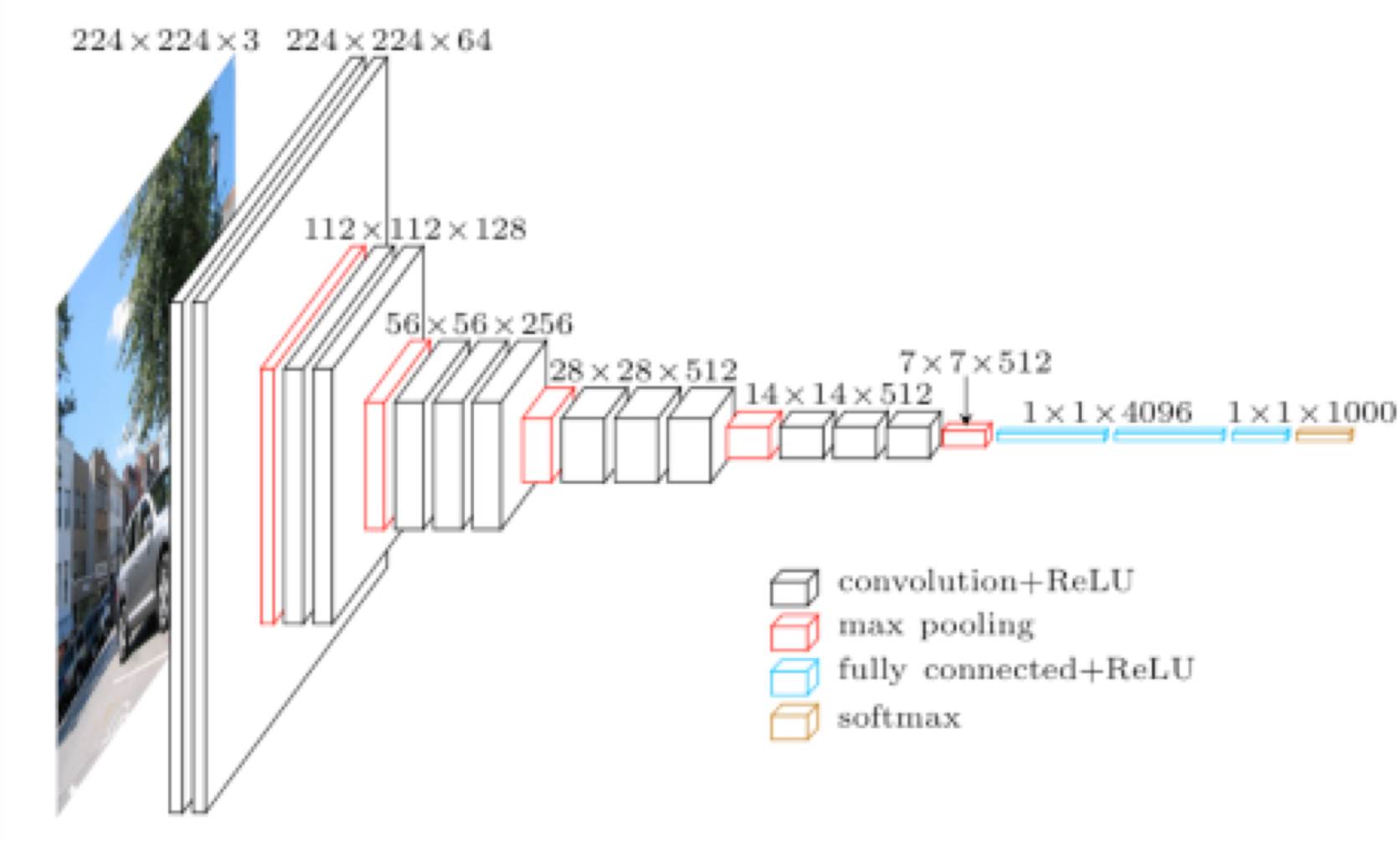
Why Does Transfer Learning Work?





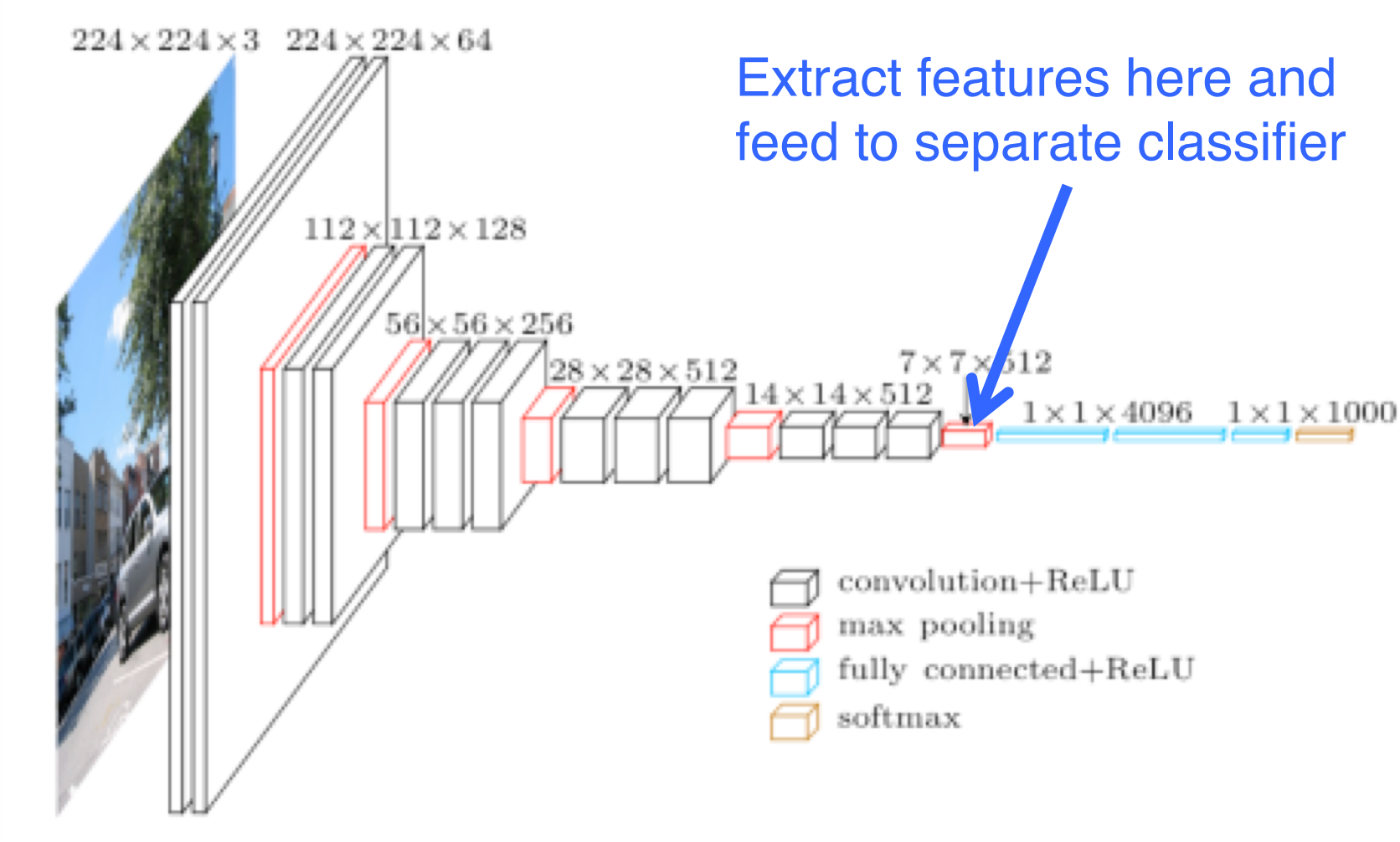
Lee et al. 'Convolutional Deep Belief Networks for Scalable Unsupervised Learning of Hierarchical Representations' ICML 2009

VGG as Pre-Trained Network



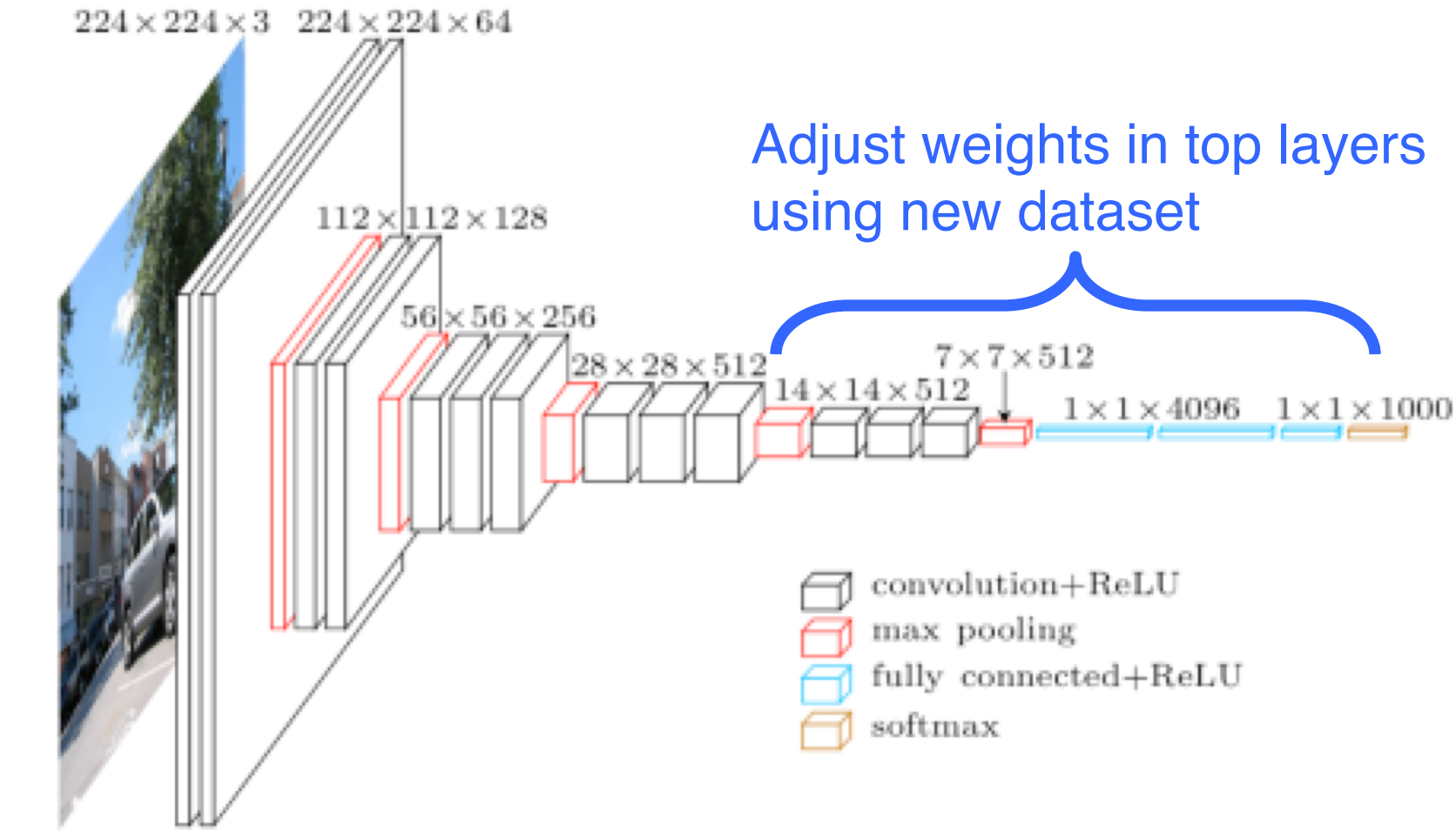
Source: https://www.cs.toronto.edu/~frossard/post/vgg16/

Transfer Learning – Feature Extraction



Source: https://www.cs.toronto.edu/~frossard/post/vgg16/

Transfer Learning – Fine Tuning



Source: https://www.cs.toronto.edu/~frossard/post/vgg16/

When & How to Fine Tune

- **New dataset is small & similar to original dataset**
  - Extract features from higher layer and feed to separate classifier
- **New dataset is large & similar to original dataset**
  - Fine tune top or all layers
- **New dataset is small & different from original dataset**
  - Extract features from lower layer and feed to separate classifier
- **New dataset is large & different from original dataset**
  - Fine tune top or all layers

Other Practical Tips

- **Learning rate**
  - Use very small learning rate for fine tuning. Don't want to destroy what was already learned.
- **Start with properly trained weights**
  - Train top-level classifier first, then fine tune lower layers.
  - Top model with random weights may have negative effects on when fine tuning weights in pre-trained model
- **Data augmentation**
  - Simple ways to slightly alter images
    - Horizontal/vertical flips, random crops, translations, rotations, etc.
  - Use to artificially expand your dataset

References

- F. Chollet. The Keras Blog. https://blog.keras.io/building-powerful-image-classification-models-using-very-little-data.html
- ImageNet. http://www.image-net.org/
- Transfer Learning. http://cs231n.github.io/transfer-learning/

Additional CNN Resources

- Caffe Model Zoo. http://caffe.berkeleyvision.org/model_zoo.html
- CS231n Convolutional Neural Networks for Visual Recognition. http://cs231n.github.io/
- Keras Documentation. https://keras.io/
- TensorFlow Getting Started. https://www.tensorflow.org/get_started/
- TensorFlow Neural Network Playground. http://playground.tensorflow.org/

Questions?