Computer Vision with Self-Supervised Learning

Danna Gurari
University of Colorado Boulder
Fall 2021

Review

- Last lecture topic:
  - Vision and sound
- Assignments (Canvas)
  - Final project proposal due earlier today
  - Final project outline due next week
    - Description link: https://home.cs.colorado.edu/~DrG/Courses/RecentAdvancesInComputerVision/FinalProject.html
- Questions?

Self-Supervised Learning: Today's Topics

- Problem
- Idea
- Generation-based methods
- Context-based methods

Self-Supervised Learning: Today's Topics

- Problem
- Idea
- Generation-based methods
- Context-based methods

What Neural Networks Learn

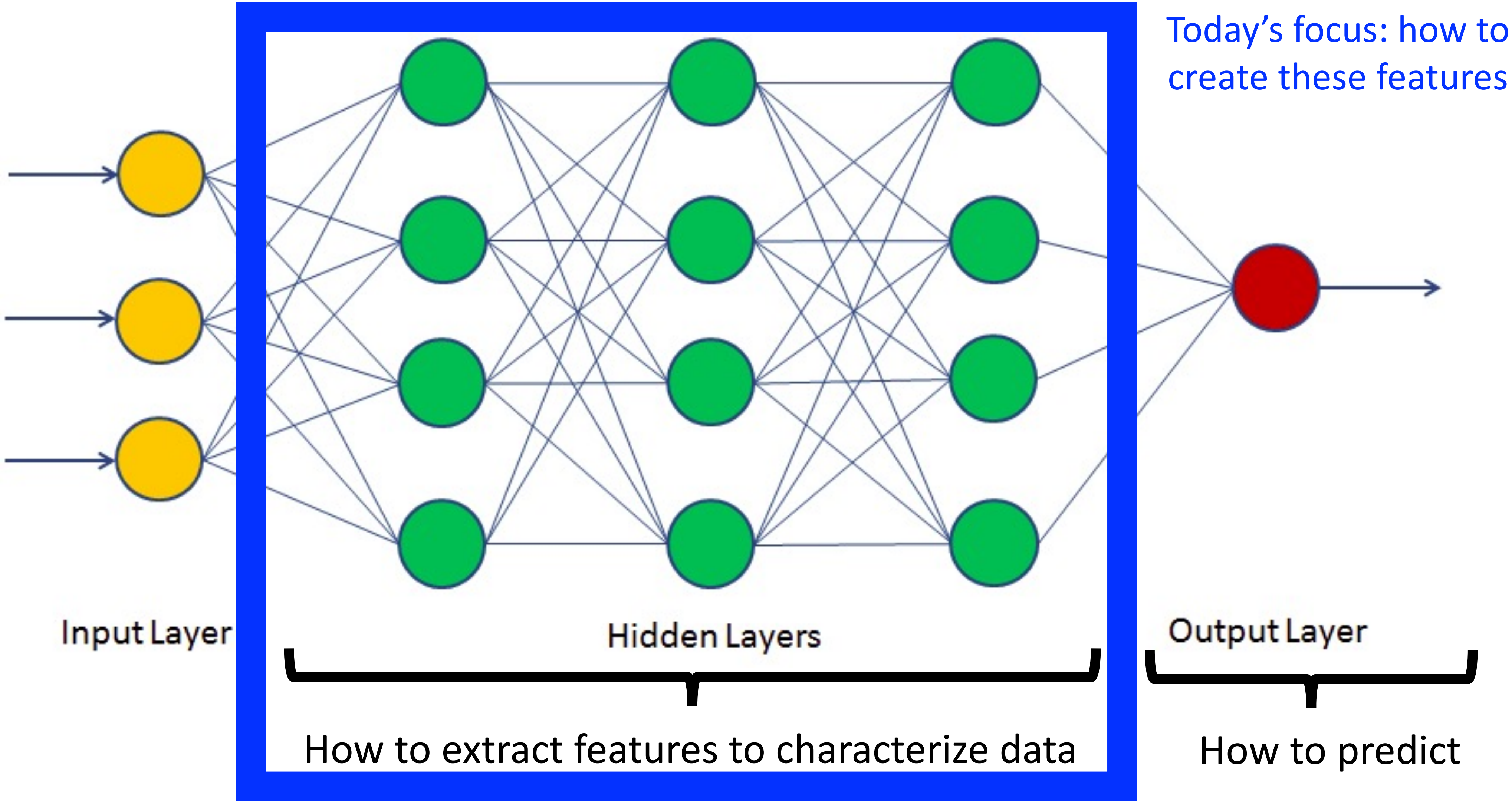


Figure Source: https://www.datacamp.com/community/tutorials/neural-network-models-r

Fine-Tuning (aka, Transfer Learning)

Key observation: features from a pretrained network can be useful for other datasets/tasks

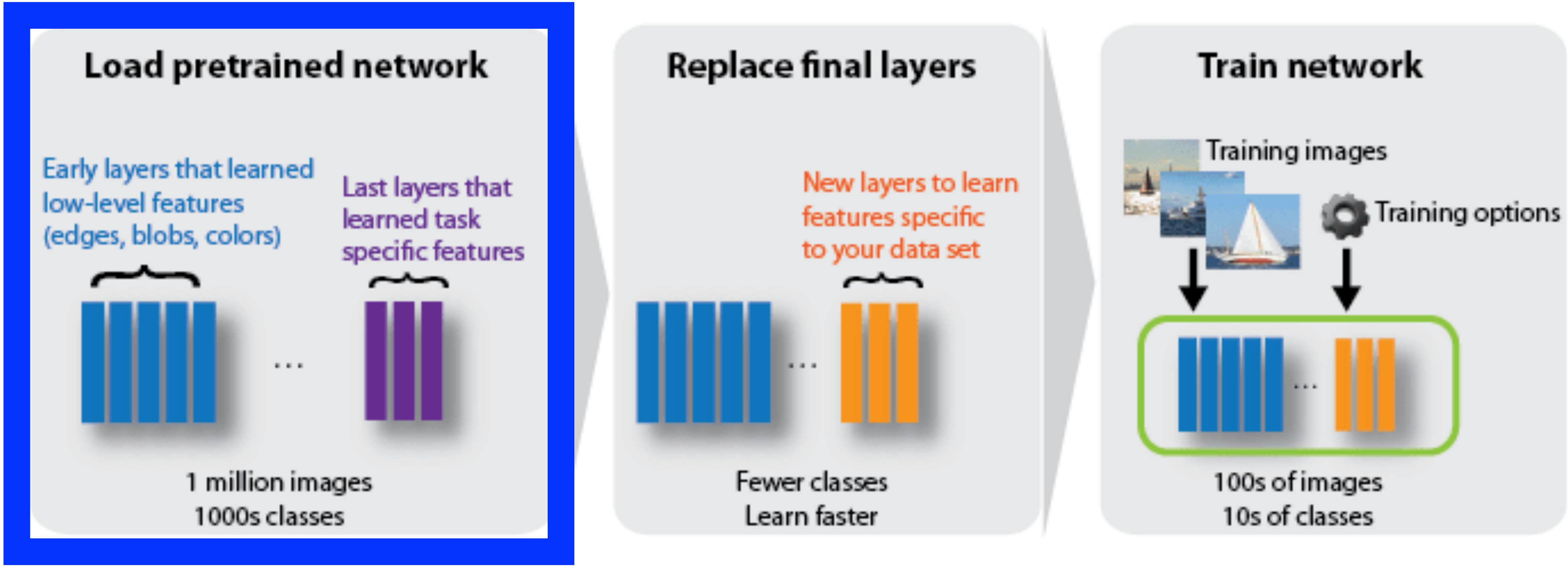
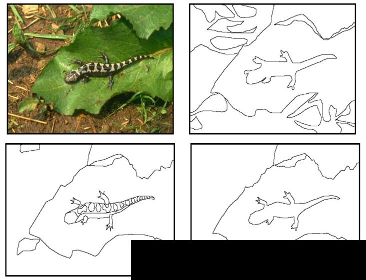
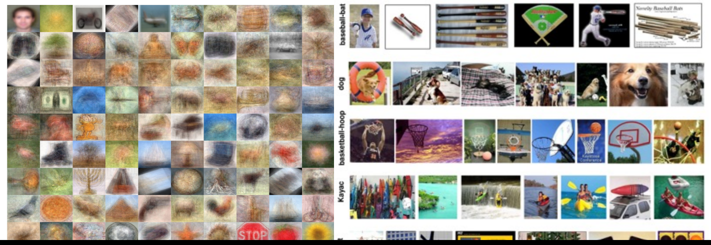


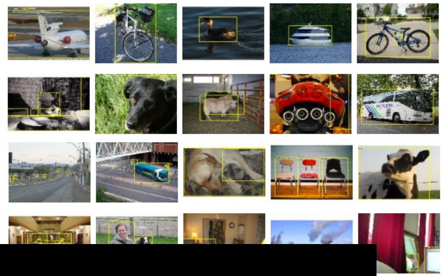
Image Source: https://www.mathworks.com/help/deeplearning/ug/transfer-learning-using-alexnet.html

How Have Pretrained Networks Learned So Far in this Class?



BSD

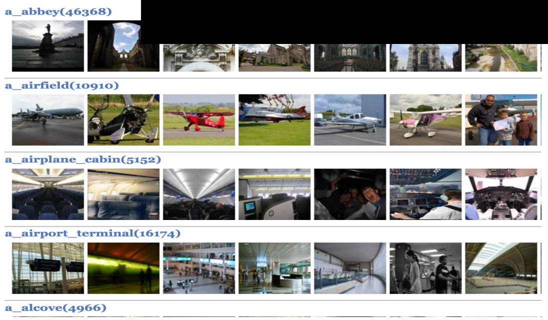




2)

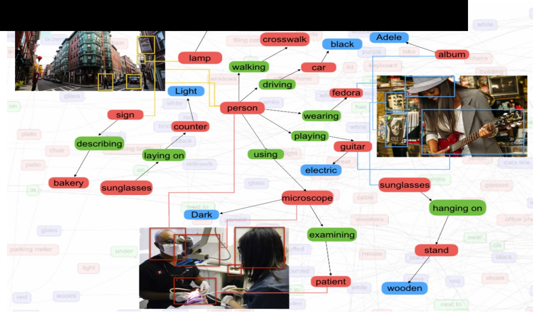
Labe

Large Labelled Datasets



Places (2014)

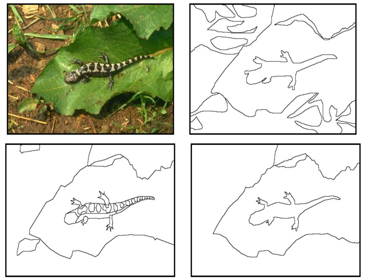
MS COCO (2014)

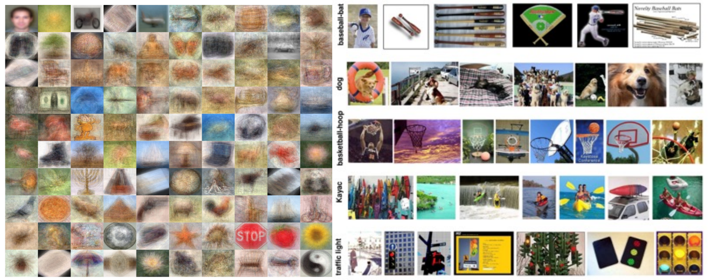


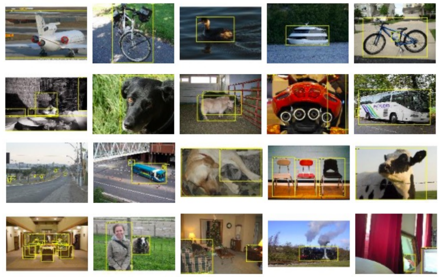
Visual Genome (2016)

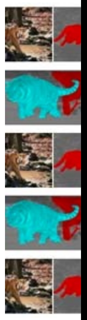
Slide Credit: http://vision.cs.utexas.edu/slides/mit-ibm-august2018.pdf

Why Not Rely On Large Labelled Datasets?



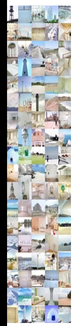


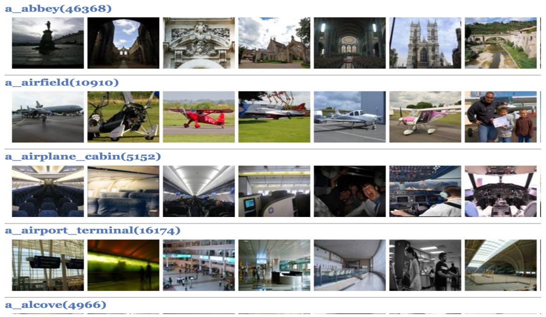




- Expensive

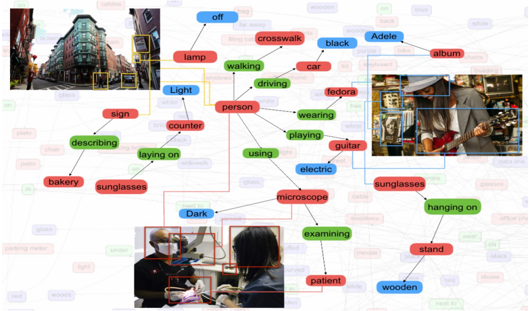
- Relatively Slow to Build Dataset





Places (2014)

MS COCO (2014)



Visual Genome (2016)

Slide Credit: http://vision.cs.utexas.edu/slides/mit-ibm-august2018.pdf

Self-Supervised Learning: Today’s Topics

- Problem
- Idea
- Generation-based methods
- Context-based methods

Intuition: How Do Humans Learn?

With Supervision

Learn from instruction

Unsupervised

Learn from experience

Today's scope

https://pixabay.com/en/toddler-learning-book-child-423227/

https://www.maxpixel.net/Father-Child-Family-Dad-Baby-Daughter-3046495

Self-Supervised Learning

A form of unsupervised learning, where the data itself serves as supervision

Image source; https://lovevery.com/community/blog/child-development/the-surprising-learning-power-of-a-household-mirror/

Idea: Self-Supervised Representation Learning

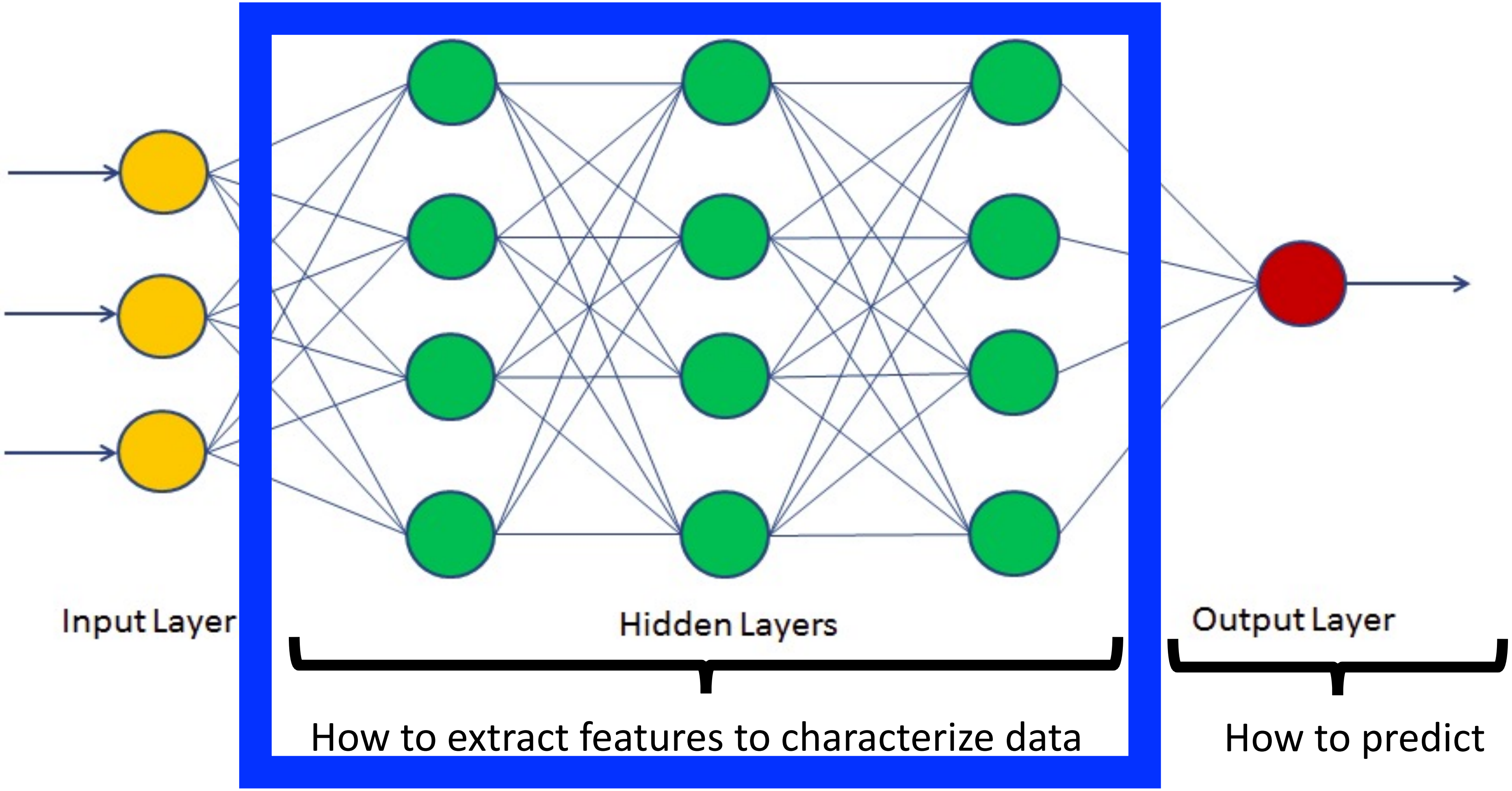


Figure Source: https://www.datacamp.com/community/tutorials/neural-network-models-r

Idea: Self-Supervised Representation Learning

- Approach: add layer after a layer of a pretrained network (fine-tuning) learned with self-supervised learning

- When and why use self-supervised pretraining?
  - Too costly and slow to collect labels for exclusive supervised training
  - Little training data is available

Self-Supervised Learning: Today's Topics

- Problem
- Idea
- Generation-based methods
- Context-based methods

Generative-based Methods

- Autoencoder: predict self
- Colorization: convert grayscale to color
- Video prediction: predict future frames

Generative-based Methods

- Autoencoder: predict self
- Colorization: convert grayscale to color
- Video prediction: predict future frames

Image Autoencoder Architecture

- Learn to copy the input to the output

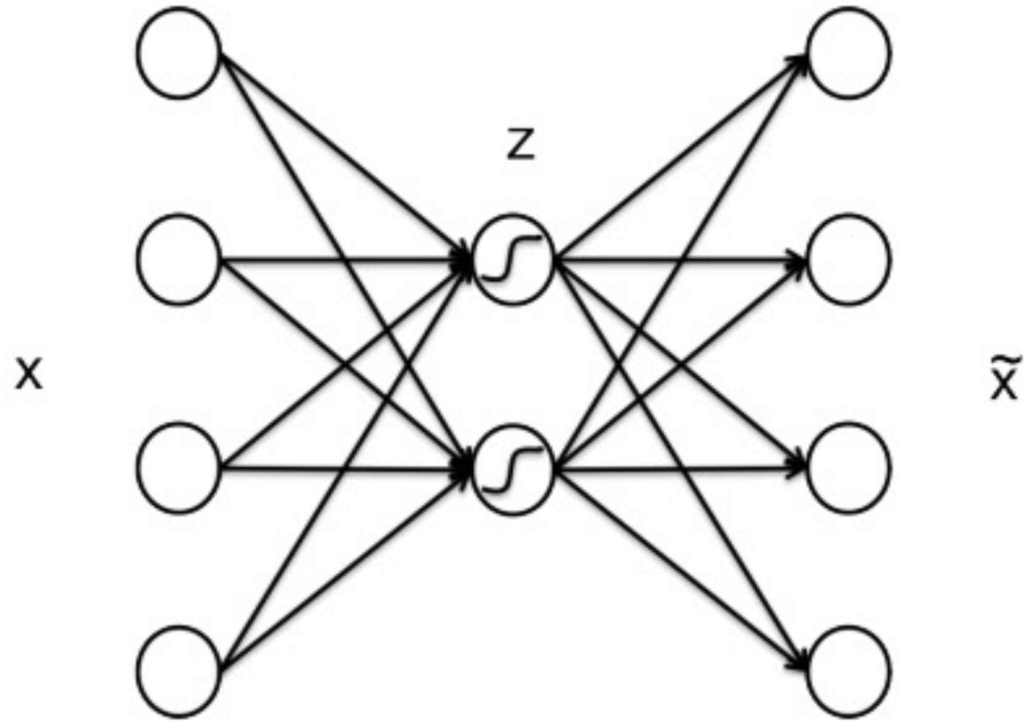


Figure Credit: https://lazyprogrammer.me/a-tutorial-on-autoencoders/

Image Autoencoder Architecture

- Consists of two parts:
  - **Encoder**: compresses inputs to an internal representation
  - **Decoder**: tries to reconstruct the input from the internal representation

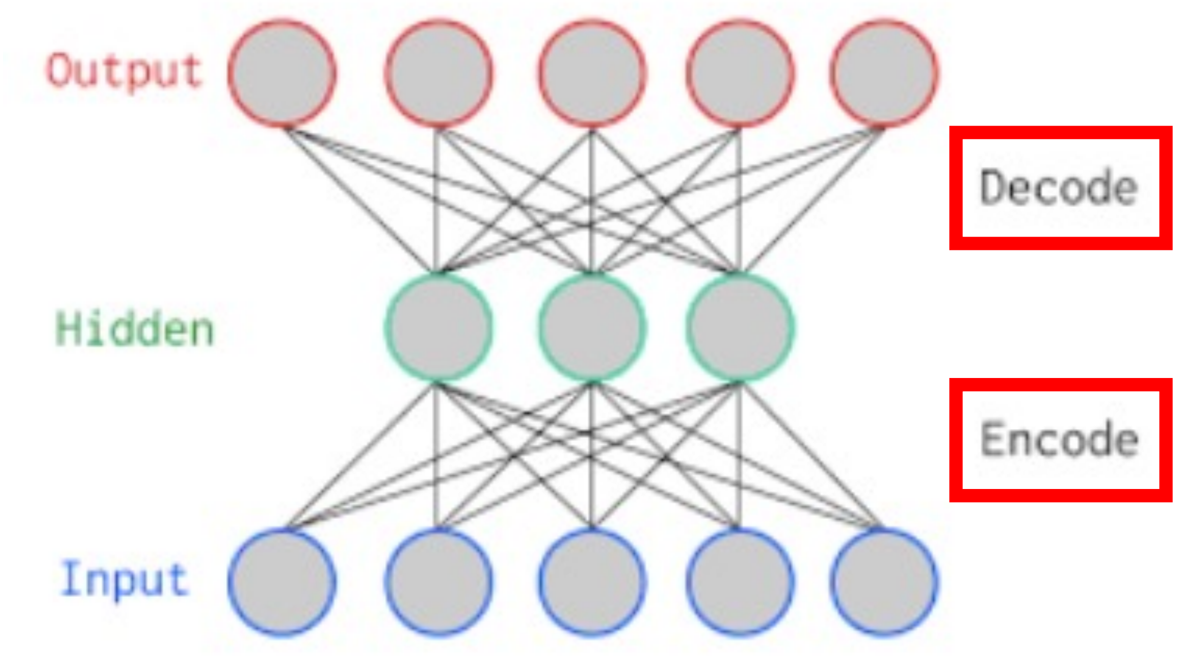
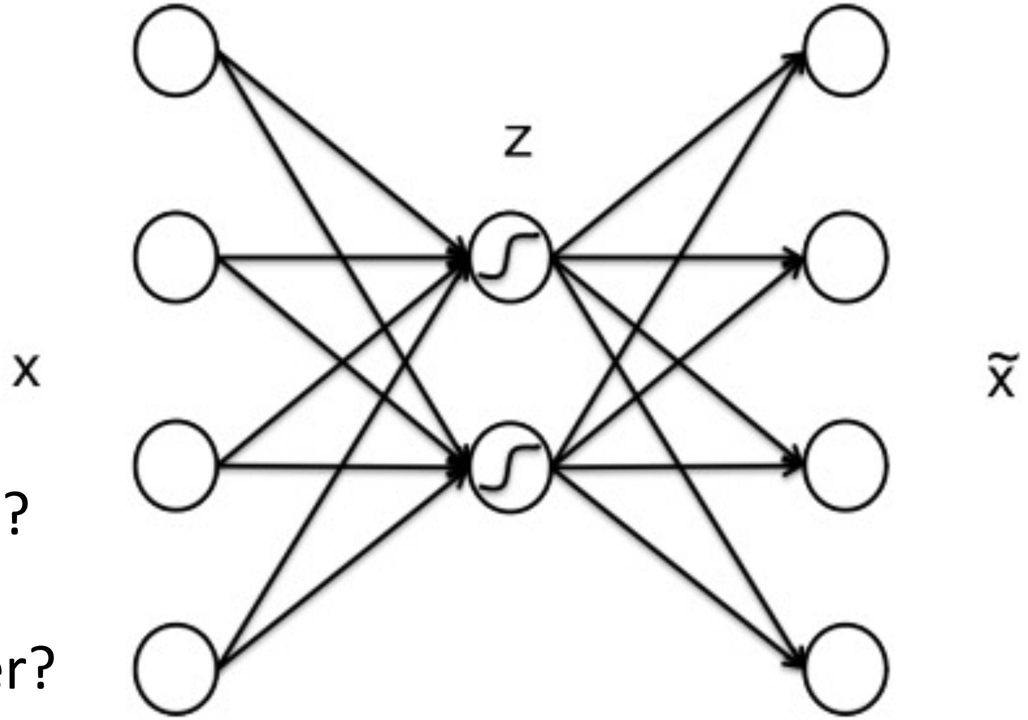


Figure Credit: https://www.datacamp.com/community/tutorials/autoencoder-keras-tutorial

Image Autoencoder Architecture

- Given this input 620 x 426 image (264,120 pixels):



  - What would a perfect autoencoder predict?
    - Itself
  - What number of nodes are in the final layer?
    - 264,120

Figure Credit: https://lazyprogrammer.me/a-tutorial-on-autoencoders/

Image Autoencoders

- Intuition: which number sequence is easier to remember?
  - **A:** 30, 27, 22, 11, 6, 8, 7, 2
  - **B:** 30, 15, 46, 23, 70, 35, 106, 53, 160, 80, 40, 20, 10, 5
- **B:** need learn only two rules
  - If even, divide by 2
  - If odd, multiply by 3 and add 1

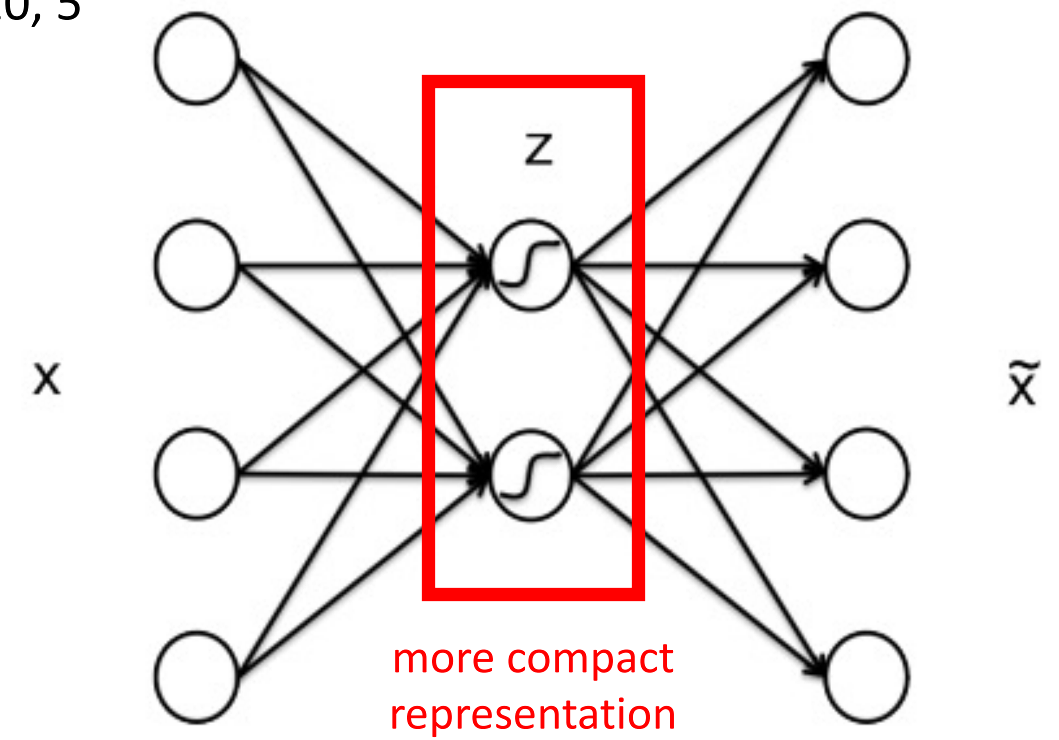


Figure Credit: https://lazyprogrammer.me/a-tutorial-on-autoencoders/

Image Autoencoder Training

How do you train a neural network?

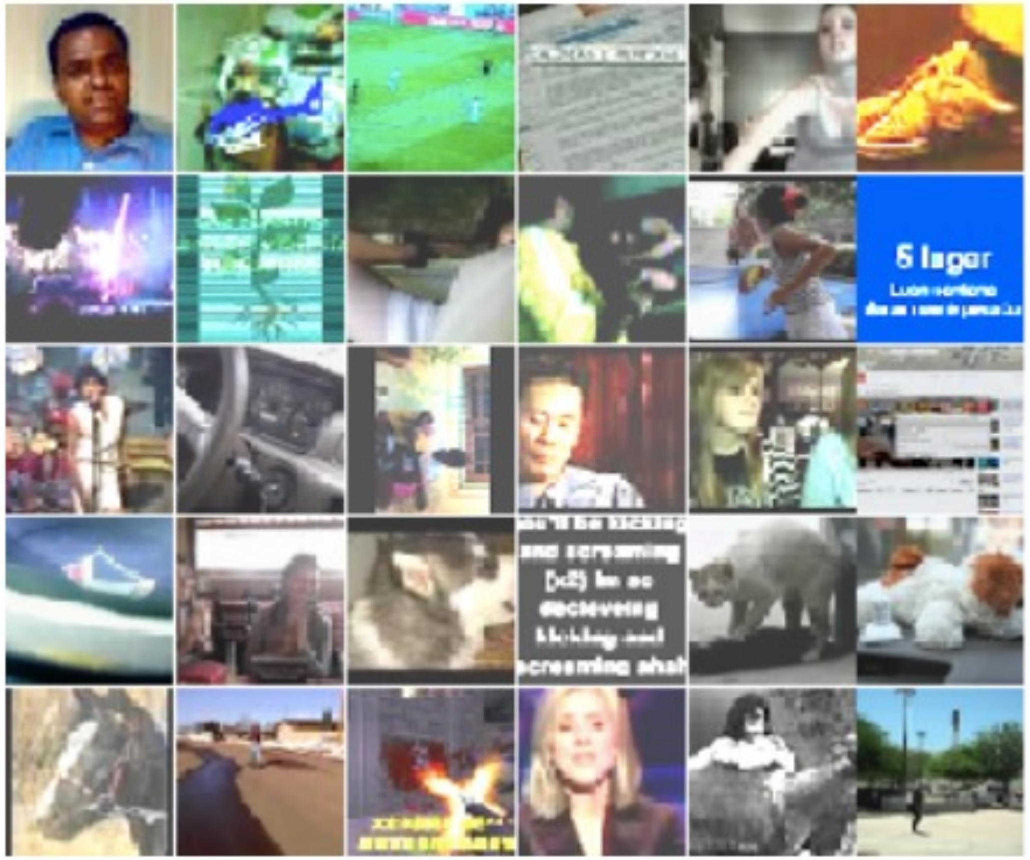
Image Autoencoder Training

Repeat until stopping criterion met:

1. **Forward pass**: propagate training data through network to make prediction
2. **Backward pass**: using predicted output, calculate error gradients backward
3. Update each weight using calculated gradients

Image Autoencoder Features

- e.g., training data:
  - 1 image taken from 10 million YouTube videos
  - Each image is in color and 200x200 pixels

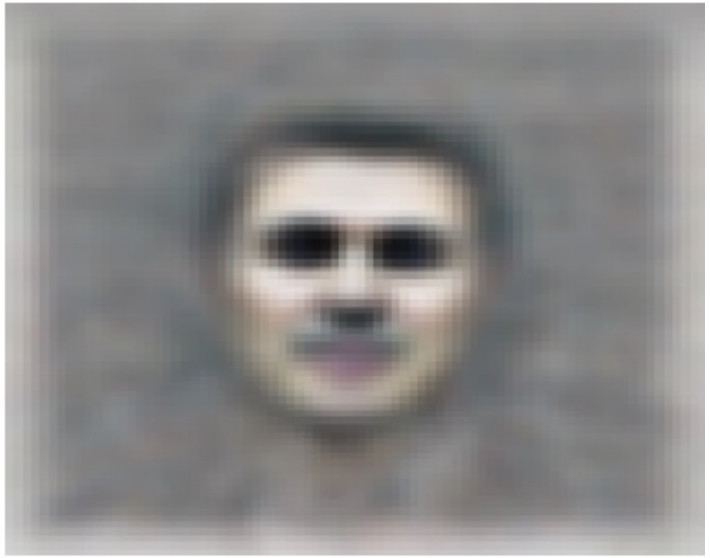


- What features do you think it learned?

Quoc V. Le et al., Building High-level Features Using Large Scale Unsupervised Learning; ICML 2013.

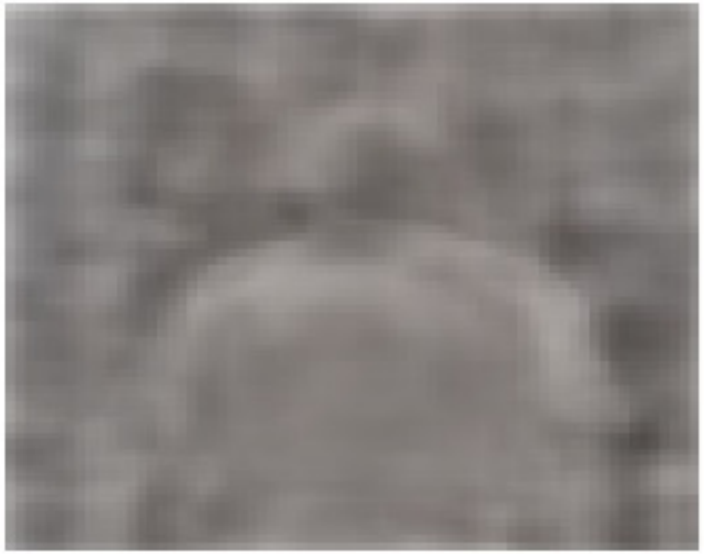
Image Autoencoder Features

- e.g., features learned include:



human face

cat face

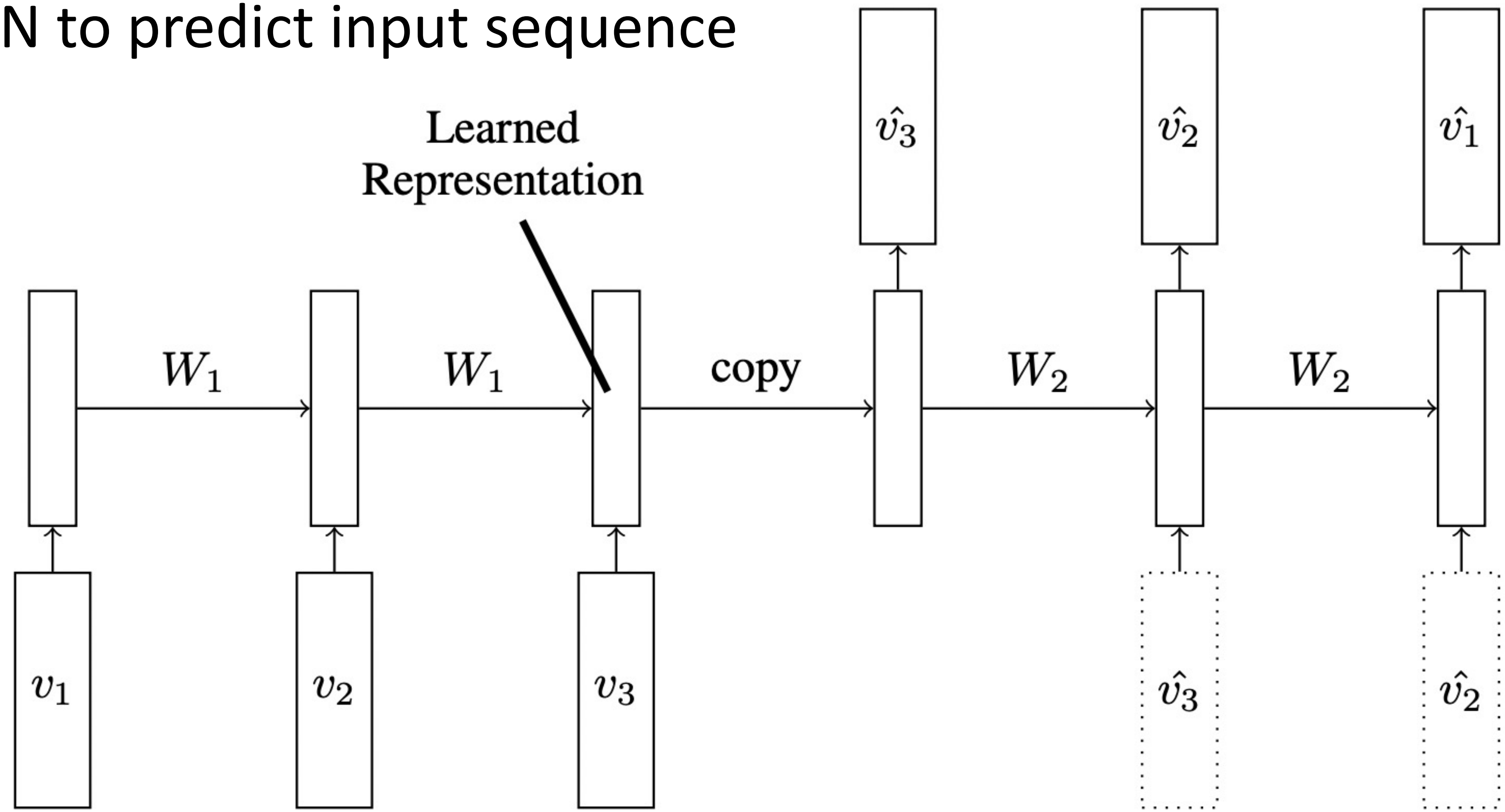


human body

Quoc V. Le et al., Building High-level Features Using Large Scale Unsupervised Learning; ICML 2013.

Video Autoencoder

- Train RNN to predict input sequence

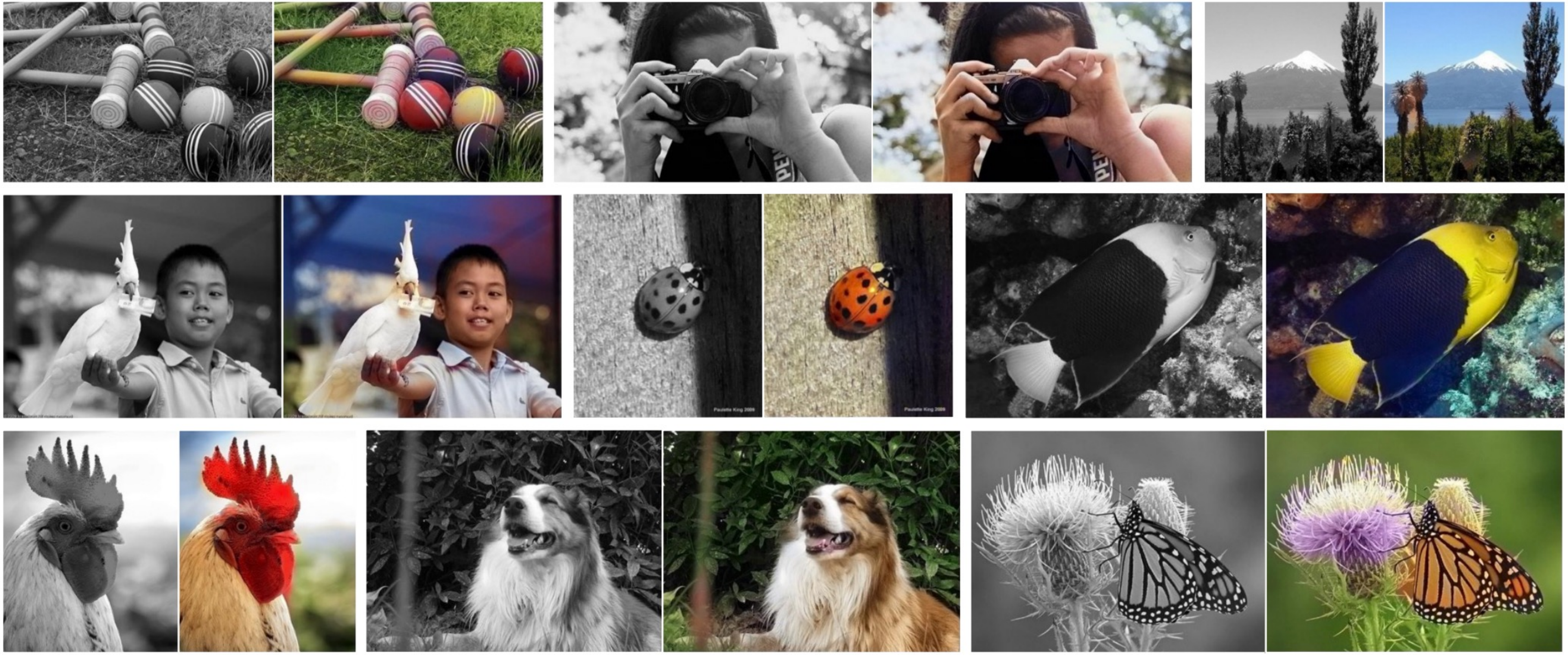


Srivastava et al., Unsupervised Learning of Video Representations using LSTMs; ICML 2015.

Generative-based Methods

- Autoencoder: predict self
- Colorization: convert grayscale to color
- Video prediction: predict future frames

Colorization: *Plausible* Coloring Results

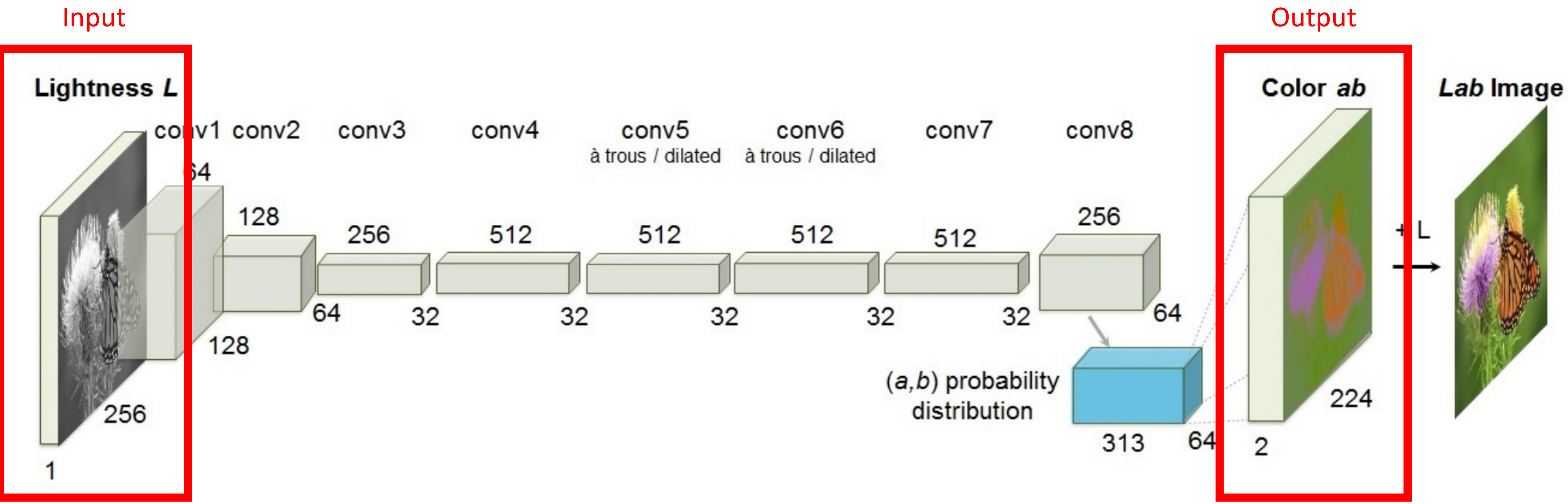


R. Zhang, P. Isoa, and A. A. Efros. Colorful Image Colorization. ECCV 2016.

Colorization: *Plausible* Coloring Results

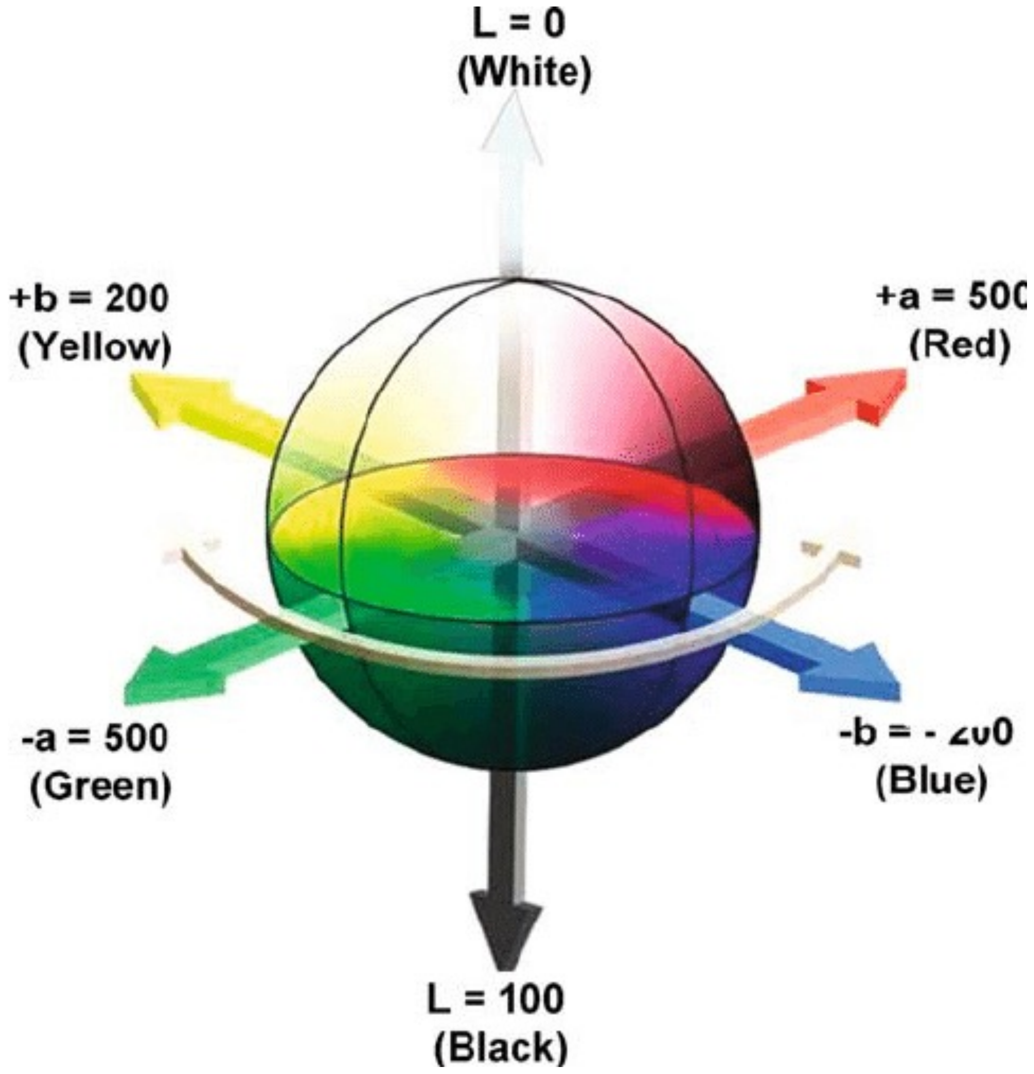
Figure Sources: https://www.flickr.com/photos/applesnpearsau/12197380673/in/photostream/; https://commons.wikimedia.org/wiki/File:JACQUES_VILET_-_1982,_Les_Fruits_du_Jardin.jpg

Image Colorization Architecture



R. Zhang, P. Isoa, and A. A. Efros. Colorful Image Colorization. ECCV 2016.

Image Colorization Architecture: CIE *Lab* Color

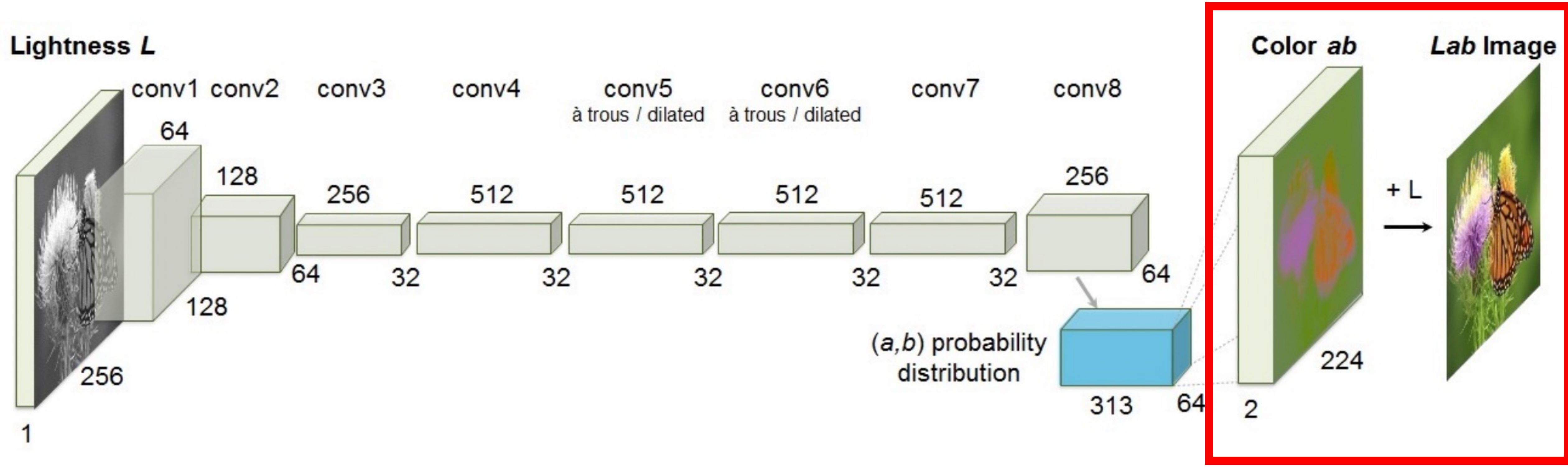


L indicates grayscale information whereas *a* and *b* represent colors

Figure source: https://www.researchgate.net/figure/The-cubical-CIE-Lab-color-space_fig3_23789543

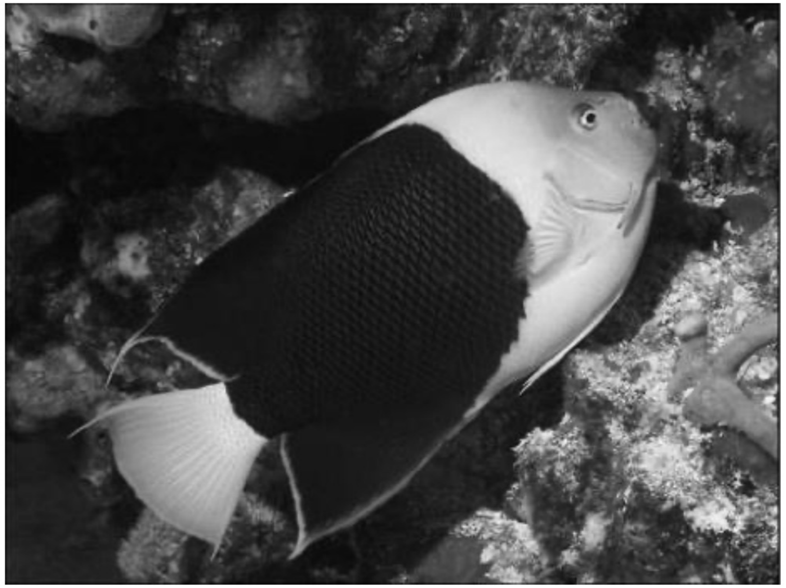
Image Colorization Architecture

Create image by combining predicted *a* and *b* channels with the *L* channel



R. Zhang, P. Isoa, and A. A. Efros. Colorful Image Colorization. ECCV 2016.

Image Colorization Architecture



Grayscale image: *L* channel

$\mathbf{X} \in \mathbb{R}^{H \times W \times 1}$

L

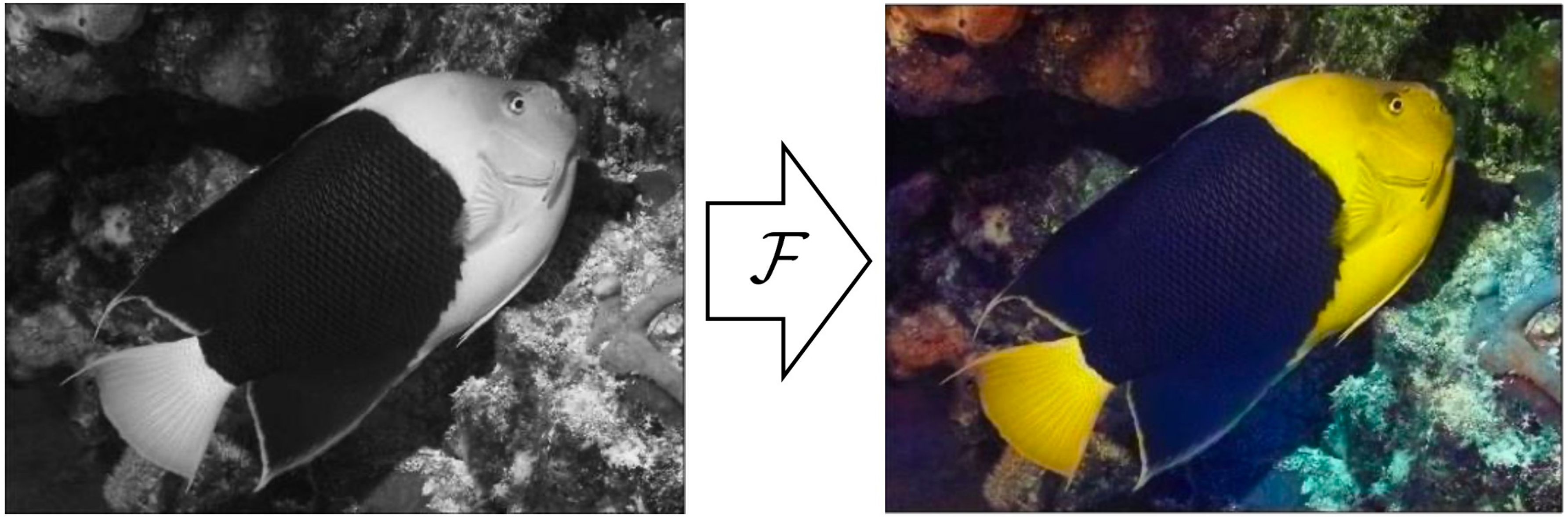
Color information: *ab* channels

$\widehat{\mathbf{Y}} \in \mathbb{R}^{H \times W \times 2}$

ab

Figure source: http://videolectures.net/eccv2016_zhang_image_colorization/

Image Colorization Architecture



Grayscale image: *L* channel
$\mathbf{X} \in \mathbb{R}^{H \times W \times 1}$

Concatenate (L,ab)
$(\mathbf{X}, \widehat{\mathbf{Y}})$

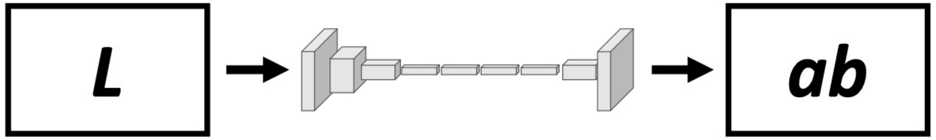


Figure source: http://videolectures.net/eccv2016_zhang_image_colorization/

Image Colorization Training

For 1.3 million ImageNet images, repeat until stopping criterion met:

1. **Forward pass**: propagate training data through network to make prediction
2. **Backward pass**: using predicted output, calculate error gradients backward
3. Update each weight using calculated gradients

R. Zhang, P. Isoa, and A. A. Efros. Colorful Image Colorization. ECCV 2016.

Image Colorization Training: Loss Function

- Regression with L2 loss inadequate

$$\mathrm{L}_2(\widehat{\mathbf{Y}}, \mathbf{Y}) = \frac{1}{2} \sum_{h,w} \|\mathbf{Y}_{h,w} - \widehat{\mathbf{Y}}_{h,w}\|_2^2$$

- Use **multinomial classification**

$$\mathrm{L}(\widehat{\mathbf{Z}}, \mathbf{Z}) = -\frac{1}{HW} \sum_{h,w} \sum_{q} \mathbf{Z}_{h,w,q} \log(\widehat{\mathbf{Z}}_{h,w,q})$$

(captures inherent ambiguity of coloring some objects by allowing the system to predict multimodal distributions)

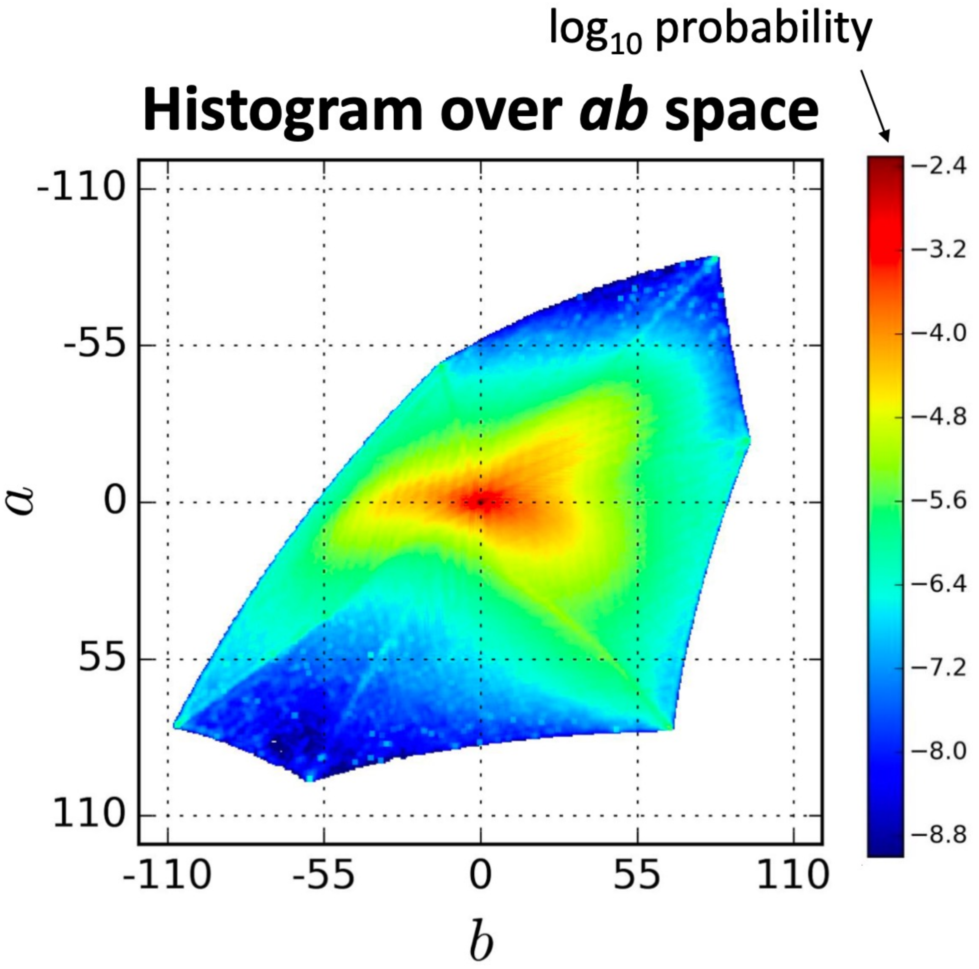


Figure source: http://videolectures.net/eccv2016_zhang_image_colorization/

Image Colorization Training: Loss Function

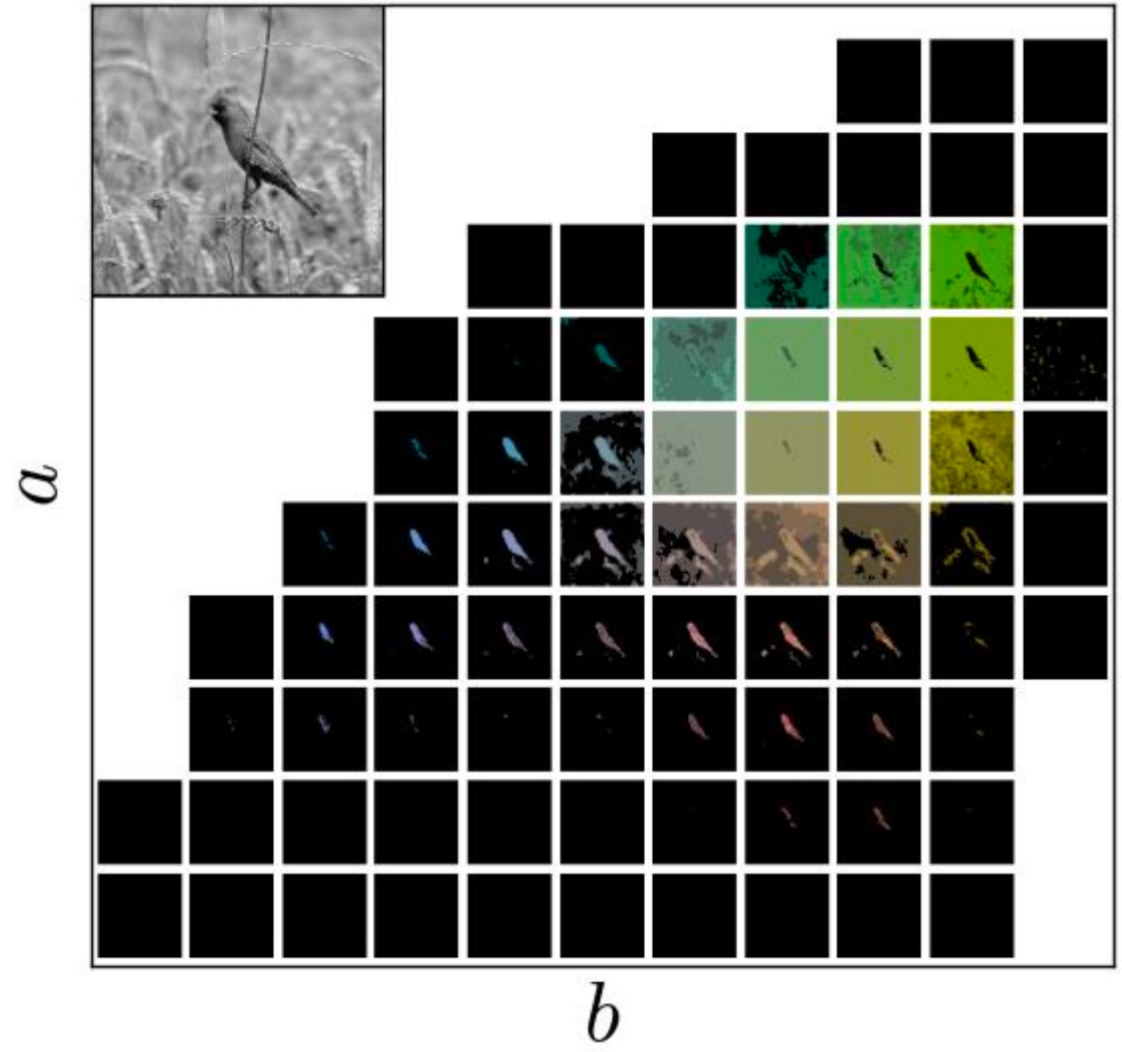


Figure source: http://videolectures.net/eccv2016_zhang_image_colorization/

Image Colorization Training: Loss Function

- Regression with L2 loss inadequate

$$\mathrm{L}_2(\widehat{\mathbf{Y}}, \mathbf{Y}) = \frac{1}{2}\sum_{h,w}\|\mathbf{Y}_{h,w} - \widehat{\mathbf{Y}}_{h,w}\|_2^2$$

- Use **multinomial classification**

$$\mathrm{L}(\widehat{\mathbf{Z}}, \mathbf{Z}) = -\frac{1}{HW}\sum_{h,w}\sum_{q}\mathbf{Z}_{h,w,q}\log(\widehat{\mathbf{Z}}_{h,w,q})$$

- **Class rebalancing** to encourage learning of *rare* colors

$$\mathrm{L}(\widehat{\mathbf{Z}}, \mathbf{Z}) = -\frac{1}{HW}\sum_{h,w}v(\mathbf{Z}_{h,w})\sum_{q}\mathbf{Z}_{h,w,q}\log(\widehat{\mathbf{Z}}_{h,w,q})$$

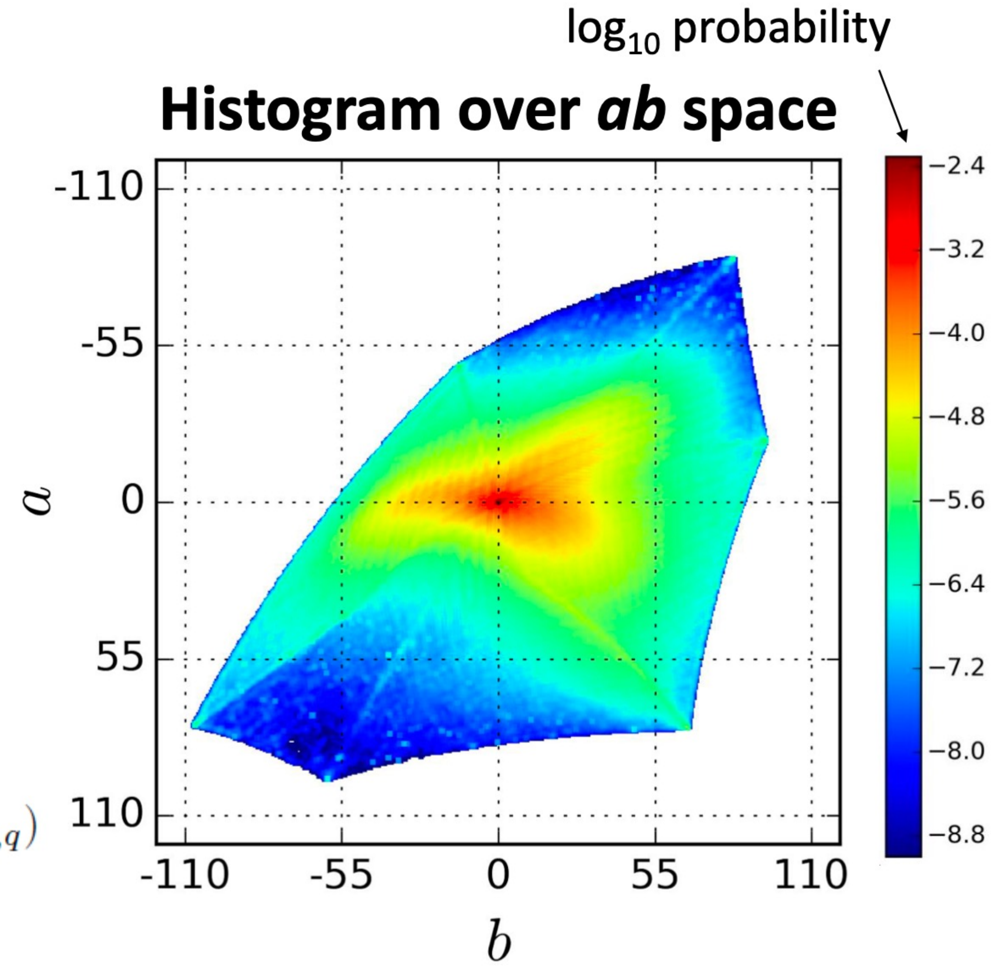


Figure source: http://videolectures.net/eccv2016_zhang_image_colorization/

Image Colorization Features

Task requires understanding an image at the pixel and semantic-level

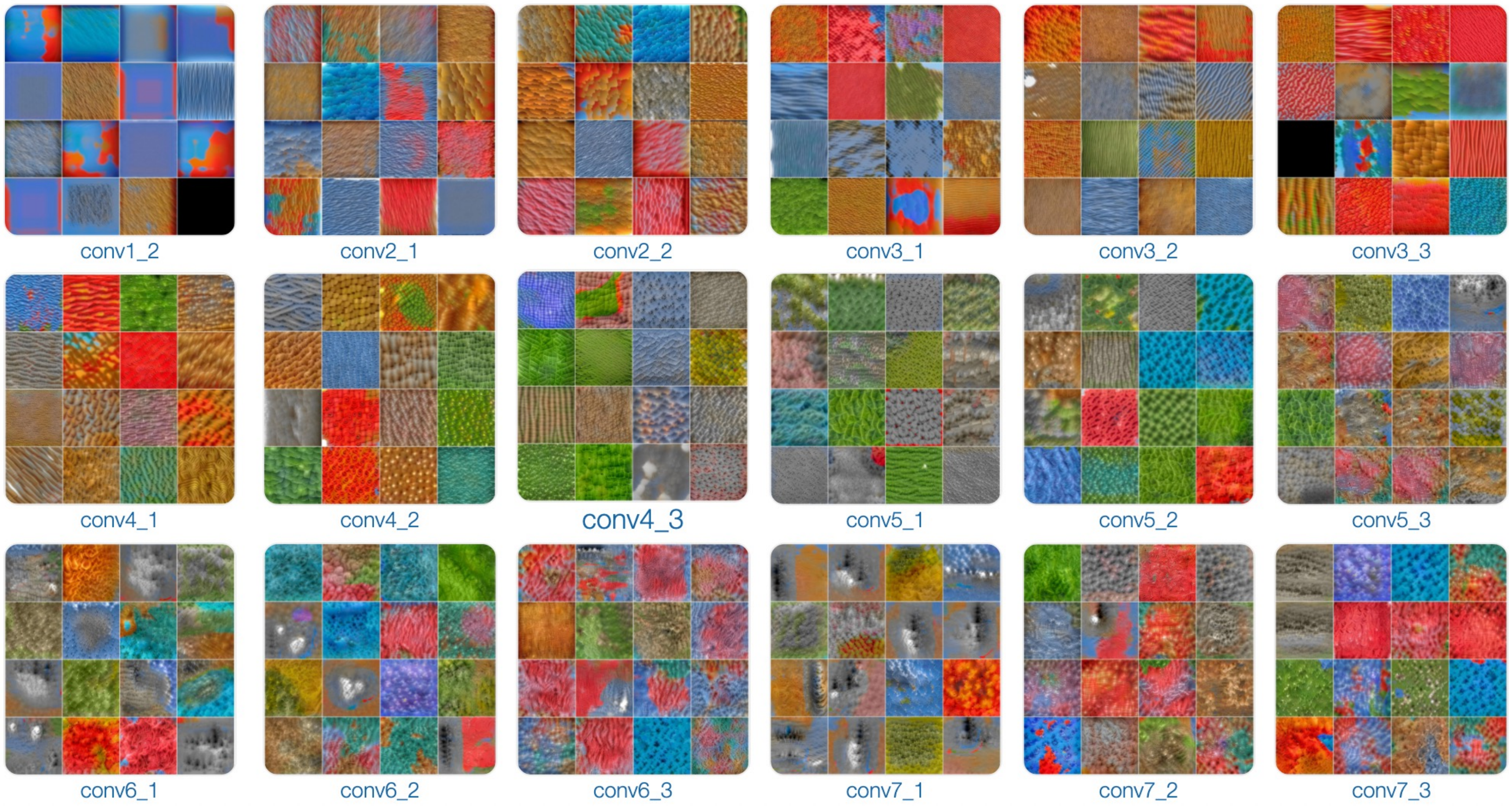
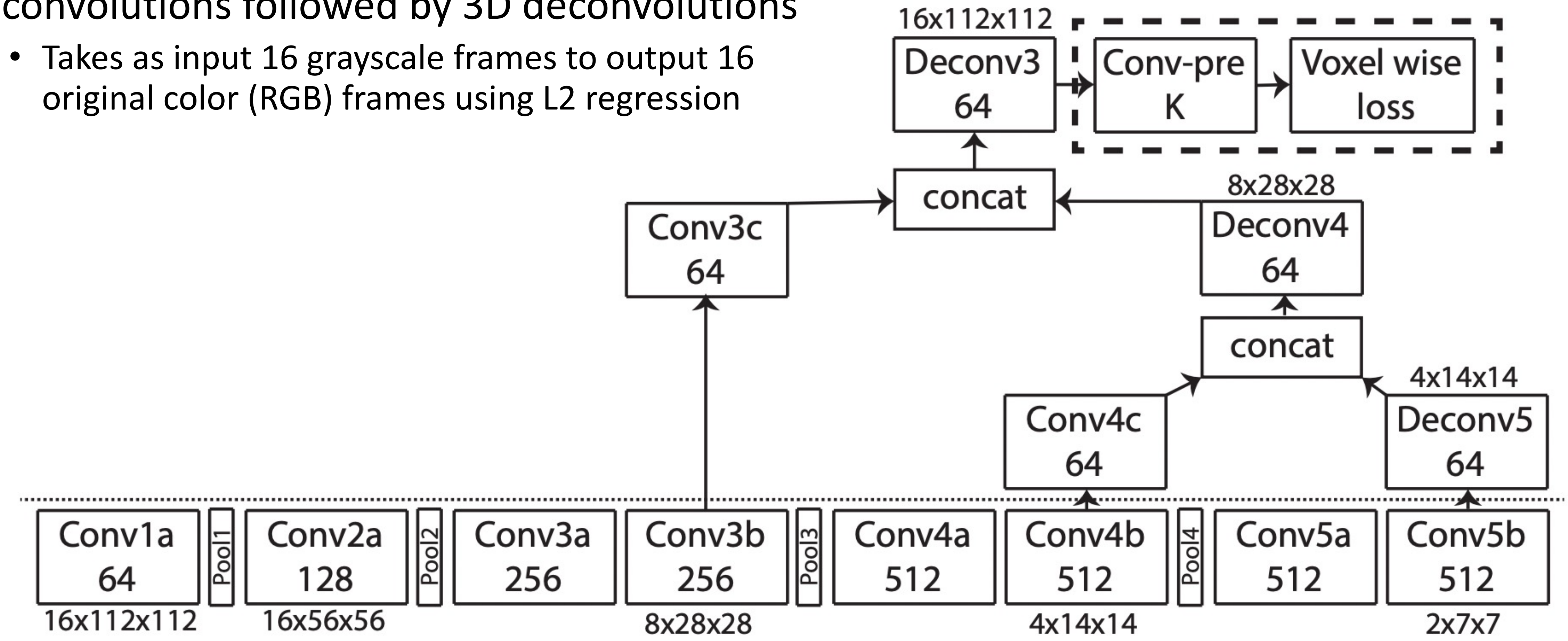


Figure source: http://richzhang.github.io/colorization/

Video Colorization

3D convolutions followed by 3D deconvolutions

- Takes as input 16 grayscale frames to output 16 original color (RGB) frames using L2 regression



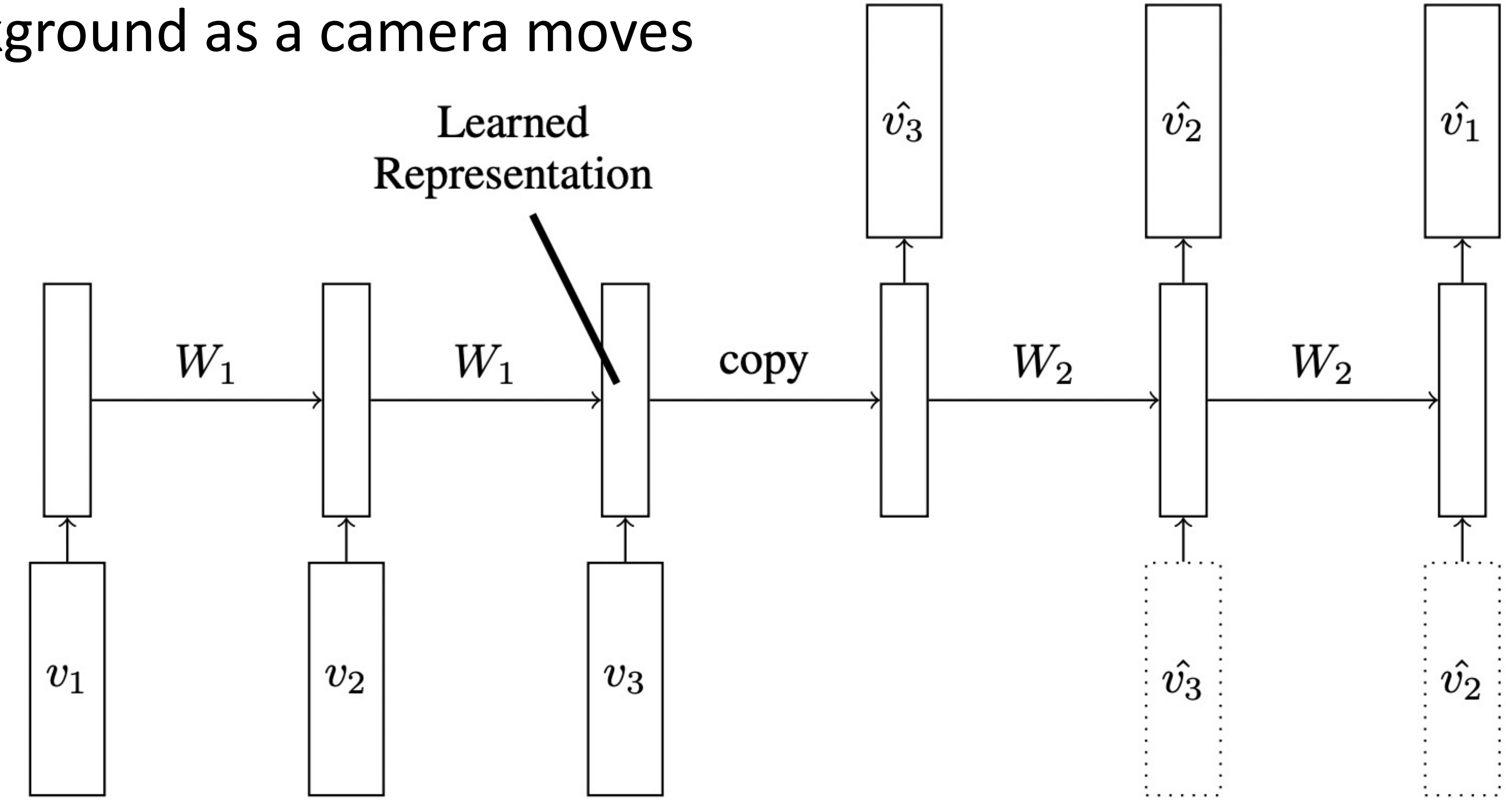
Tran et al. Deep End2End Voxel2Voxel Prediction. CVPR 2016.

Generative-based Methods

- Autoencoder: predict self
- Colorization: convert grayscale to color
- Video prediction: predict future frames

Video Prediction

- Train RNN to predict future frames; limitations for prediction include?
  - Identifying new objects that enter scene
  - Determining background as a camera moves



Srivastava et al., Unsupervised Learning of Video Representations using LSTMs; ICML 2015.

Generative-based Methods

- Autoencoder: predict self
- Image colorization: convert grayscale to color
- Video prediction: predict future frames

Self-Supervised Learning: Today's Topics

- Problem
- Idea
- Generation-based methods
- Context-based methods

Context-based Methods

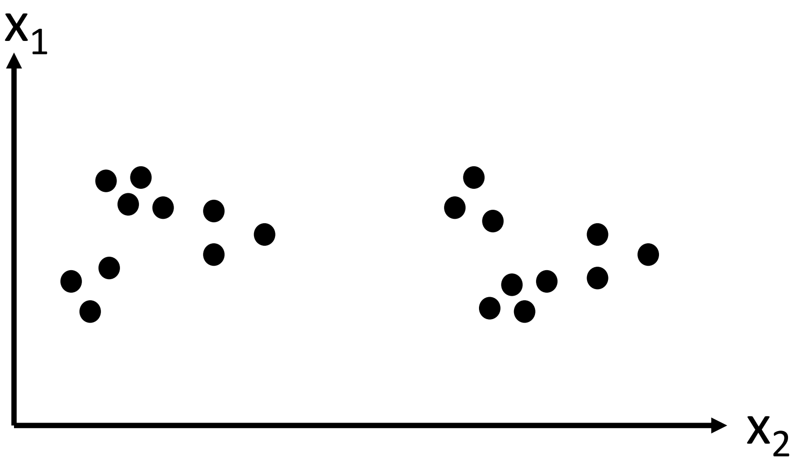
- Similarity context: clustering
- Spatial context: predict relative positions of image patches
- Timing context: predict relative positions of video frames

Context-based Methods

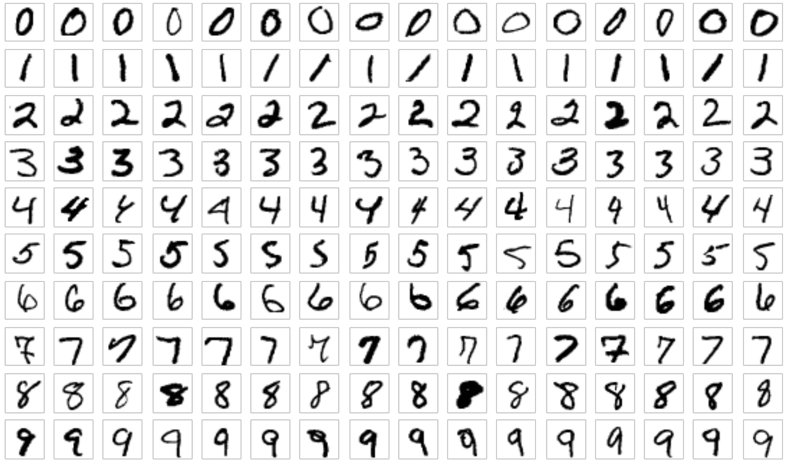
- Similarity context: clustering
- Spatial context: predict relative positions of image patches
- Timing context: predict relative positions of video frames

Clustering

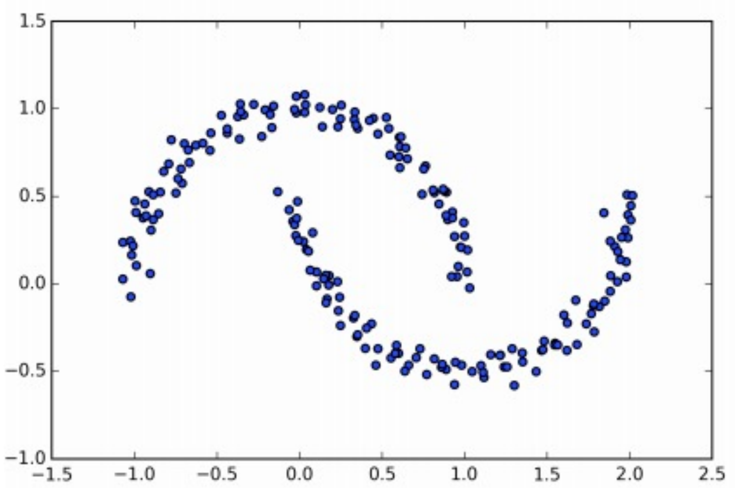
A.



B.



C.

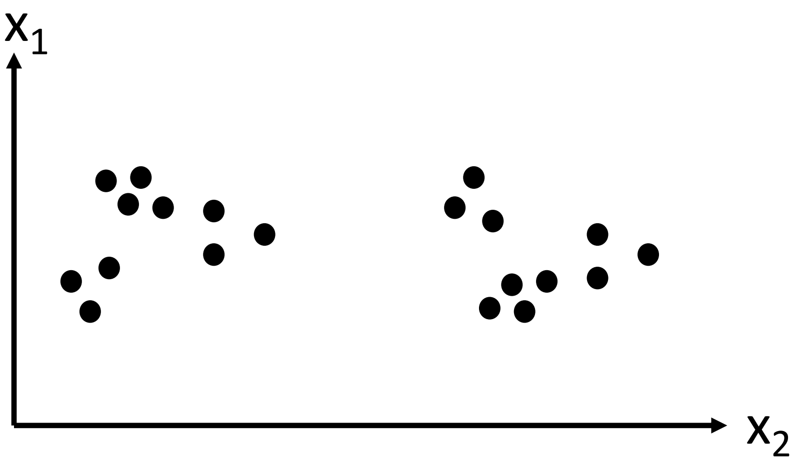


Find groupings such that entities in a group will be similar to each another and different from the entities in other groups.

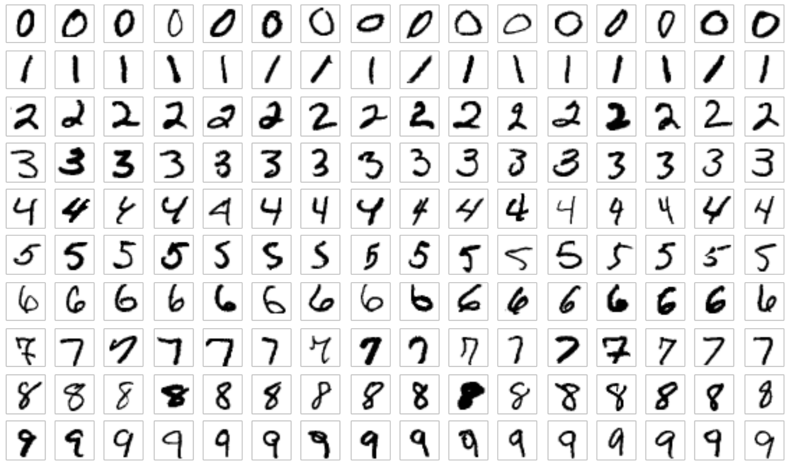
Raschka and Mirjalili; Python Machine Learning

Clustering: Key Questions

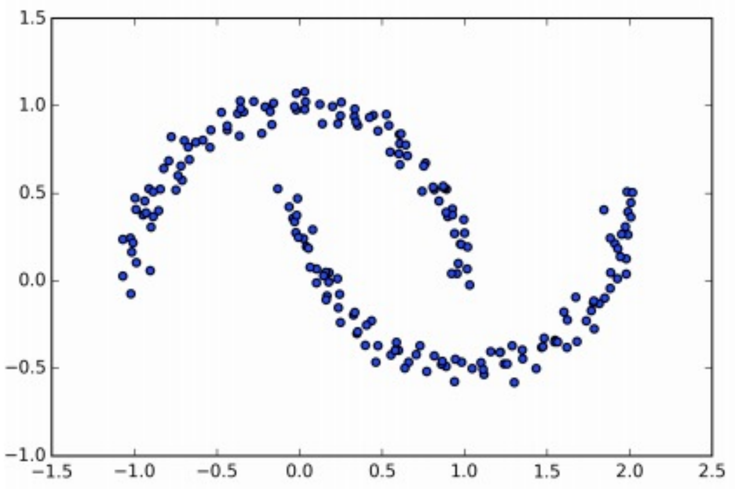
A.



B.



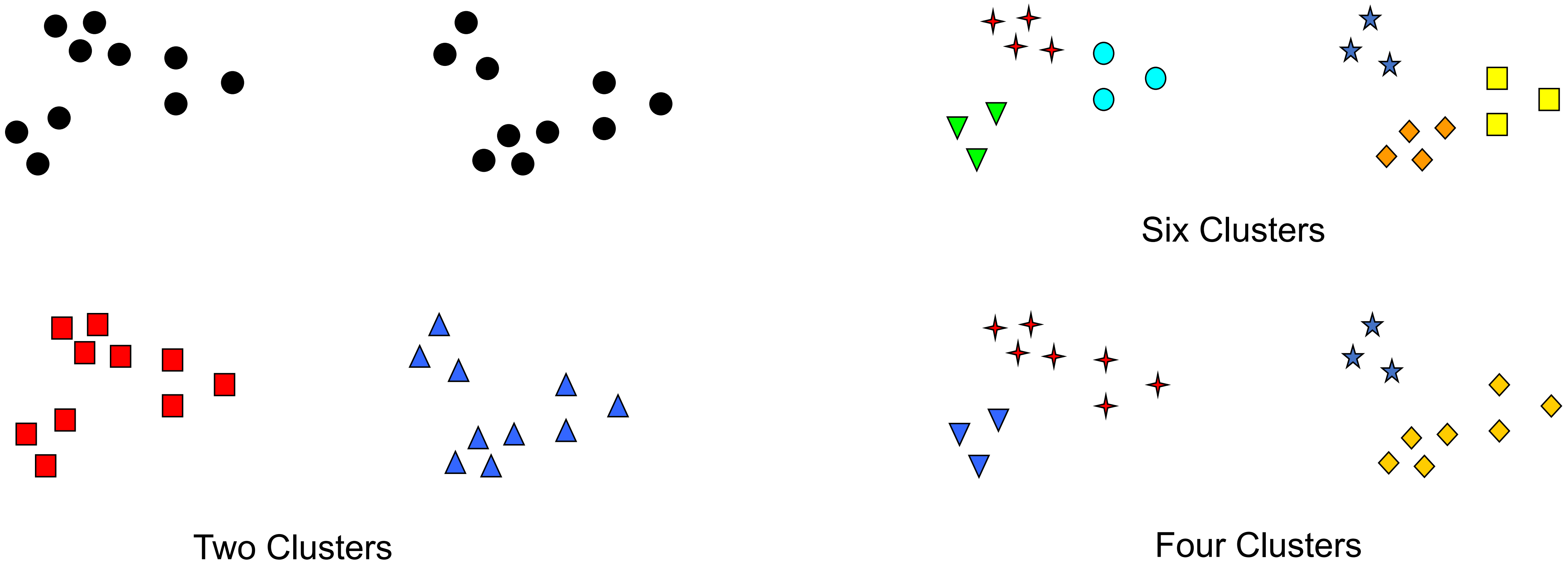
C.



- How many data clusters to create?
- What "algorithm" to use to partition the data?

Raschka and Mirjalili; Python Machine Learning

How Many Clusters?

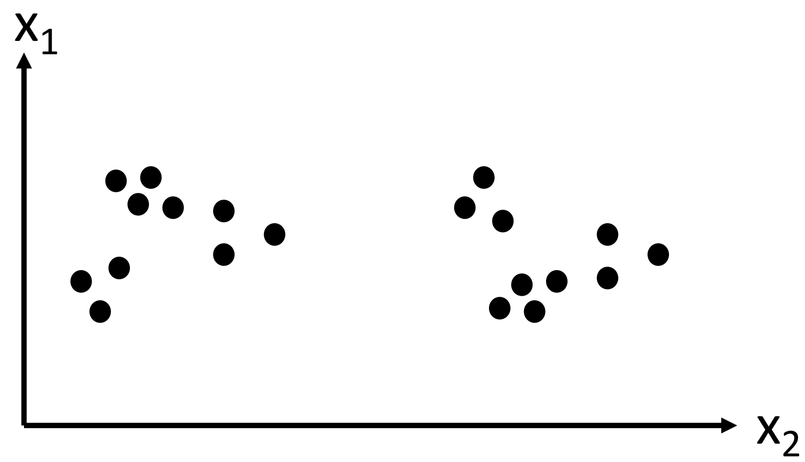


Number of clusters can be ambiguous.

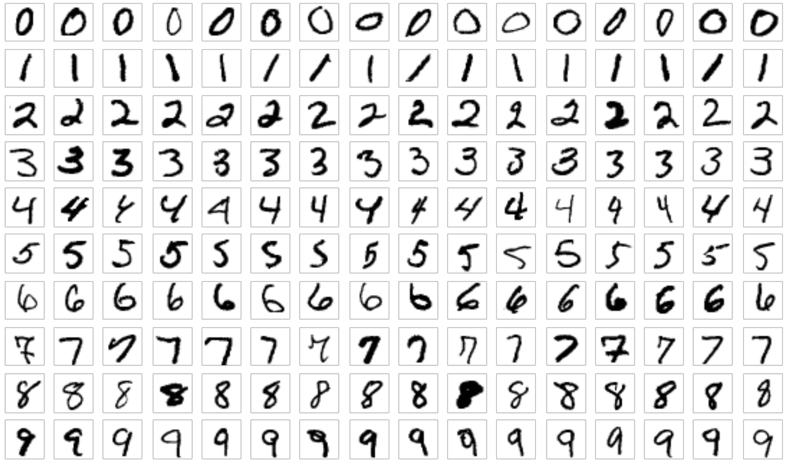
Slide adapted from: https://www-users.cs.umn.edu/~kumar001/dmbook/slides/chap7_basic_cluster_analysis.pdf

Self-Supervised Learning of Clusters

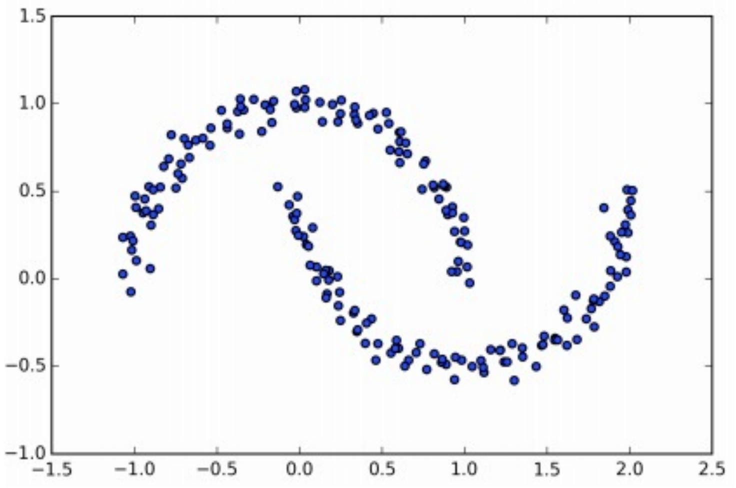
A.



B.



C.



CNNS are trained to identify cluster assignments OR to recognize whether images belong to the same cluster

Raschka and Mirjalili; Python Machine Learning

Context-based Methods

- Similarity context: clustering
- Spatial context: predict relative positions of image patches
- Timing context: predict relative positions of video frames

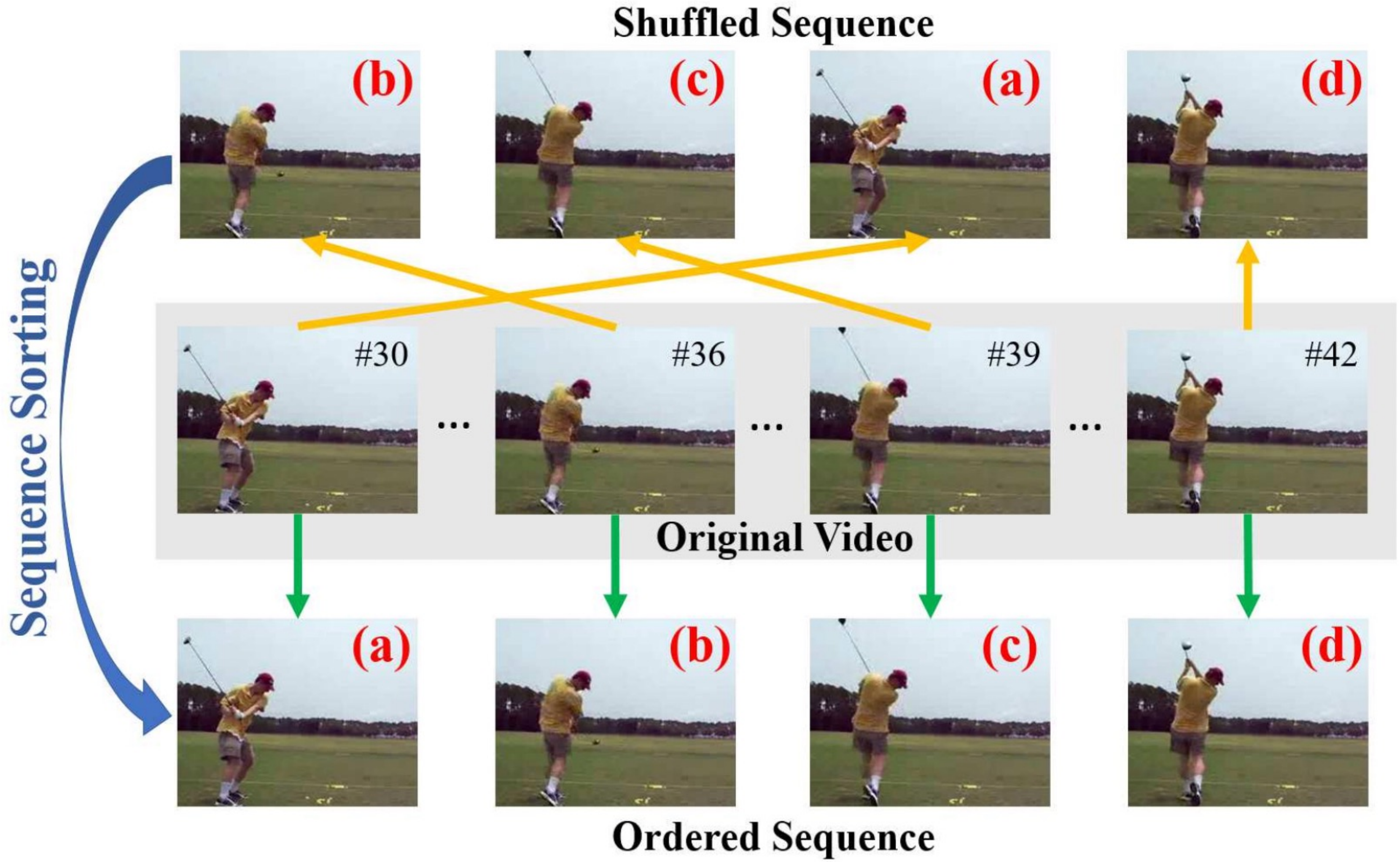
Task: Predict Image Index for Each Patch

Carl Doersch, Abhinav Gupta, and Alexei A. Efros, Unsupervised Visual Representation Learning by Context Prediction; ICCV 2015.

Context-based Methods

- Similarity context: clustering
- Spatial context: predict relative positions of image patches
- Timing context: predict relative positions of video frames

Task: Predict Order of Video Frames



Lee et al., Unsupervised Representation Learning by Sorting Sequences; ICCV 2017.

Context-based Methods

- Similarity context: clustering
- Spatial context: predict relative positions of image patches
- Timing context: predict relative positions of video frames

Can you think of any other self-supervised learning methods that could be used to learn visual features?

Self-Supervised Learning: Today's Topics

- Problem
- Idea
- Generation-based methods
- Context-based methods


The End