Exemplar Deep Learning Applications

Image Classification

Ira A. Fulton Schools of Engineering
Arizona State University

Objective

Objective

Describe an example network for image classification

.

Objective

Explain the parameters defining the network

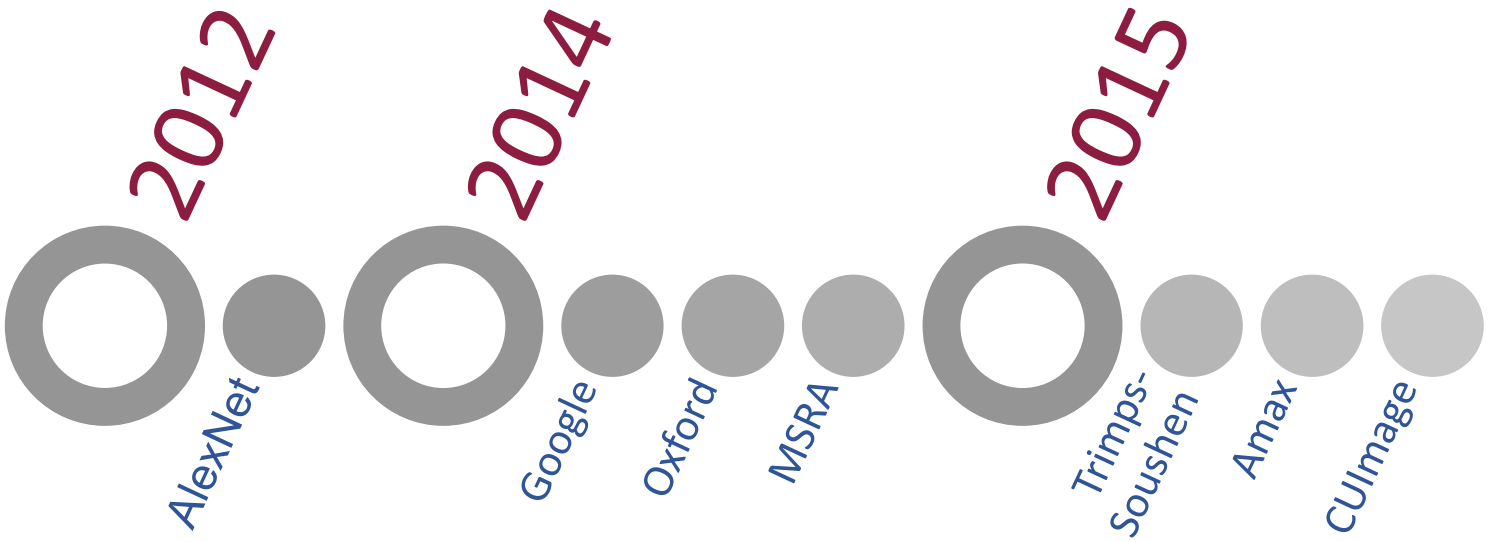
Objective

Identify common tricks for improving classification performance

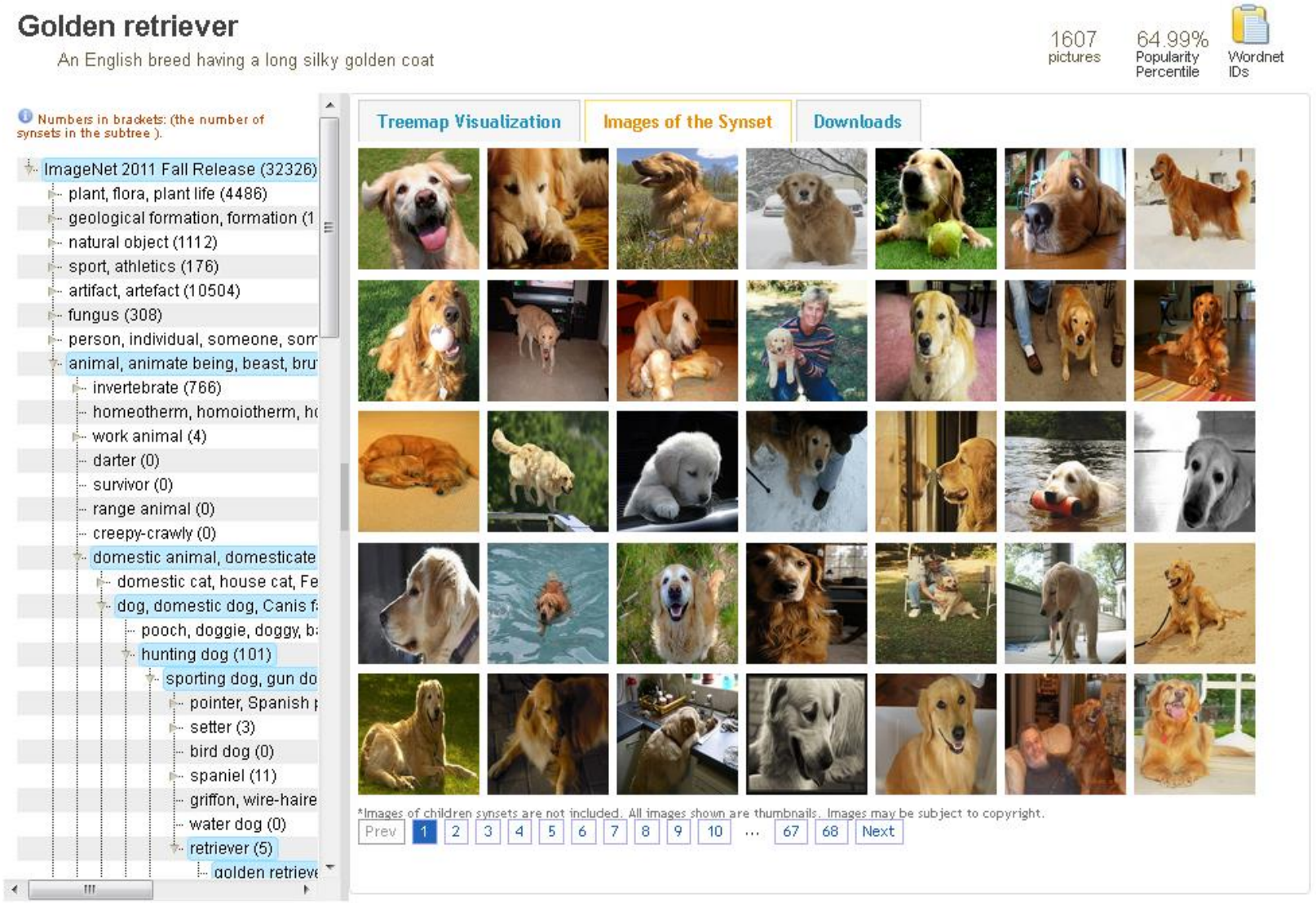
Deep Learning for Image-based Recognition

- Visual recognition is an important part of human intelligence.
- ILSVRC (ImageNet Large-scale Visual Recognition Challenge) illustrates such a task.
- Many ImageNet images are difficult for conventional algorithms to classify.

Success Stories



ImageNet.org Samples



SOURCE: ImageNet.org

Success Stories: 2014 – Top Three

Rank	Team	Error
1	Google	0.06656
2	Oxford	0.07325
3	MSRA	0.08062

SOURCE: ImageNet.org

Success Stories: 2015 – Top Three

Team Name	Entry Description	Description of Outside Data Used	Localization Error	Classifica-tion Error
Trimps-Soushen	Extra annotations collected by ourselves	Extra annotations collected by ourselves	0.122285	0.04581
Amax	Validate the classification model we used in DET Entry1	Share proposal procedure with DET for convenience	0.14574	0.04354
CUImage	Average multiple models – validation accuracy is 79.78%	3000-class classification images from ImageNet are used to pre-train CNN	0.198272	0.05858

SOURCE: ImageNet.org

Example Application 1: DR Detection

DR: Diabetic Retinopathy

A recent work: Gulshan *et al.* "Development and validation of a deep learning algorithm for detection of diabetic retinopathy in retinal fundus photographs." *JAMA* 316.22 (2016): 2402-2410

- Employed large datasets
- A specific CNN architecture (Inception-v3) taking the entire image as input (as opposed to lesion/structure-specific CNNs)
- High performance: Comparable to a panel of 7 board-certified ophthalmologists

Example Application 2: Visual Aesthetics

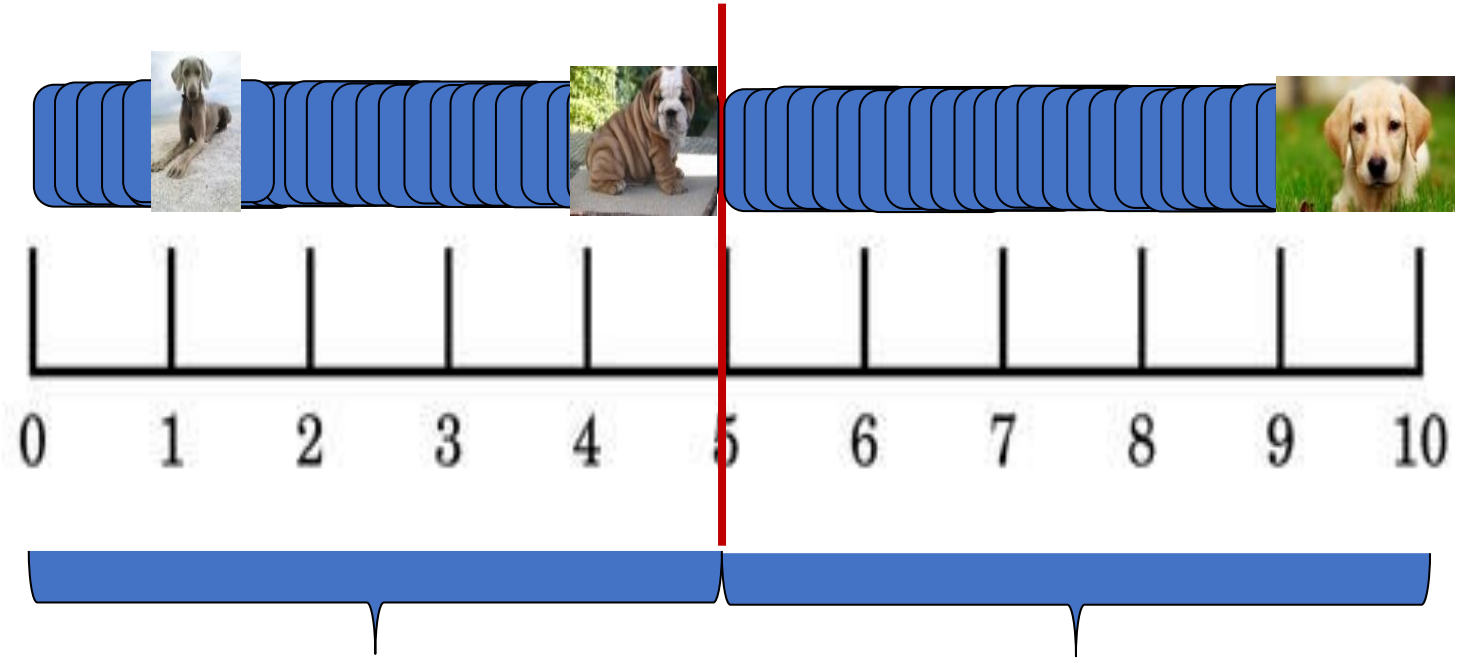
- **While being subjective, computational modes are possible since there are patterns in visually-appealing pictures.**
  - E.g., photographic rules.
- **Huge on-line datasets available. If ratings are also available, the problem becomes supervised learning.**
  - Conventional approaches still face the bottleneck of feature extraction.

Example Application 2: Visual Aesthetics

- **While being subjective, computational modes are possible since there are patterns in visually-appealing pictures.**
  - E.g., photographic rules.
- **Huge on-line datasets available. If ratings are also available, the problem becomes supervised learning.**
  - Conventional approaches still face the bottleneck of feature extraction.

Related Approaches

Solving the task as binary classification

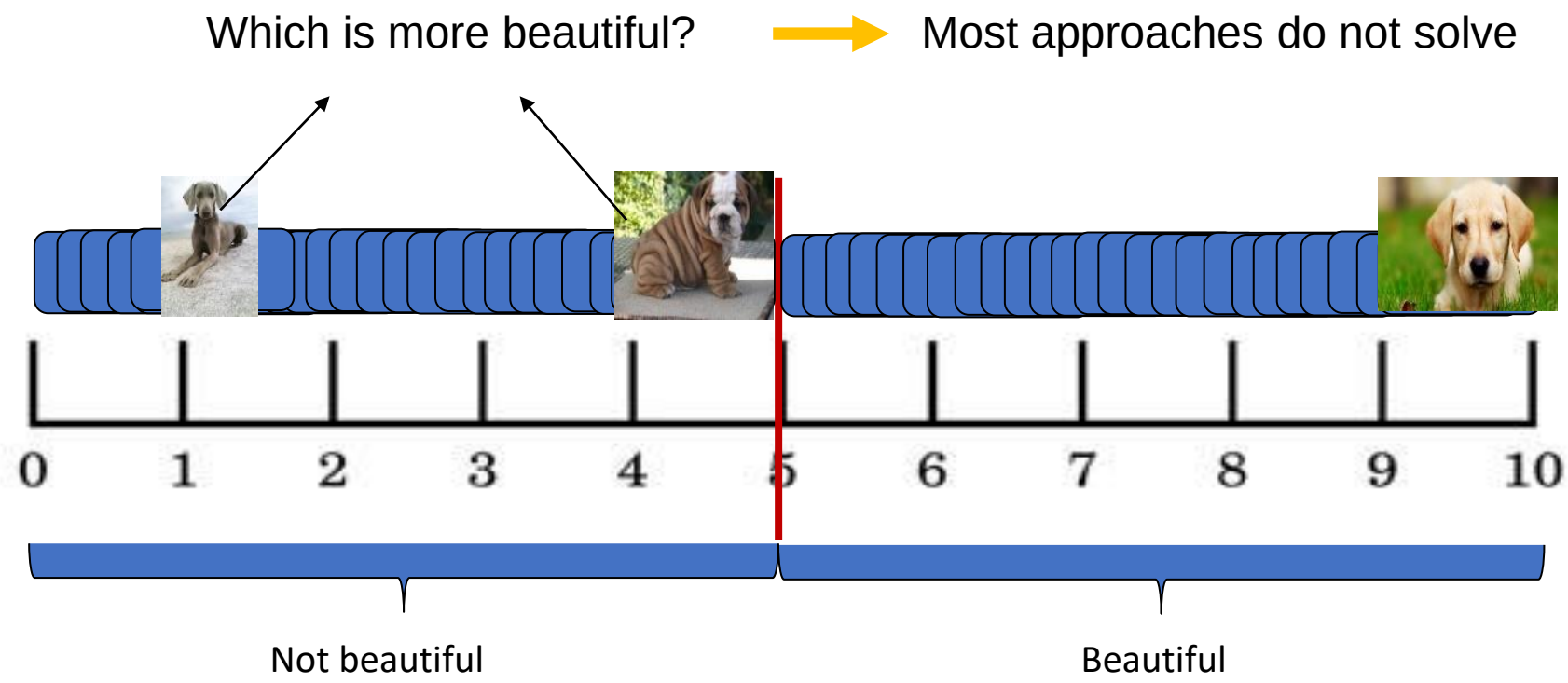


Related Approaches: Examples

- ***RAPID: Rating Pictorial Aesthetics using Deep Learning*** **(Lu *et al.*)**
- ***Deep Multi-Patch Aggregation Network for Image Style, Aesthetics, and Quality Estimation*** **(Lu *et al.*)**
- ***Image Aesthetic Evaluation Using Paralleled Deep Convolution Neural Network*** **(Guo & Li)**

A New Task: Relative Aesthetics

- Image retrieval
- Image enhancement



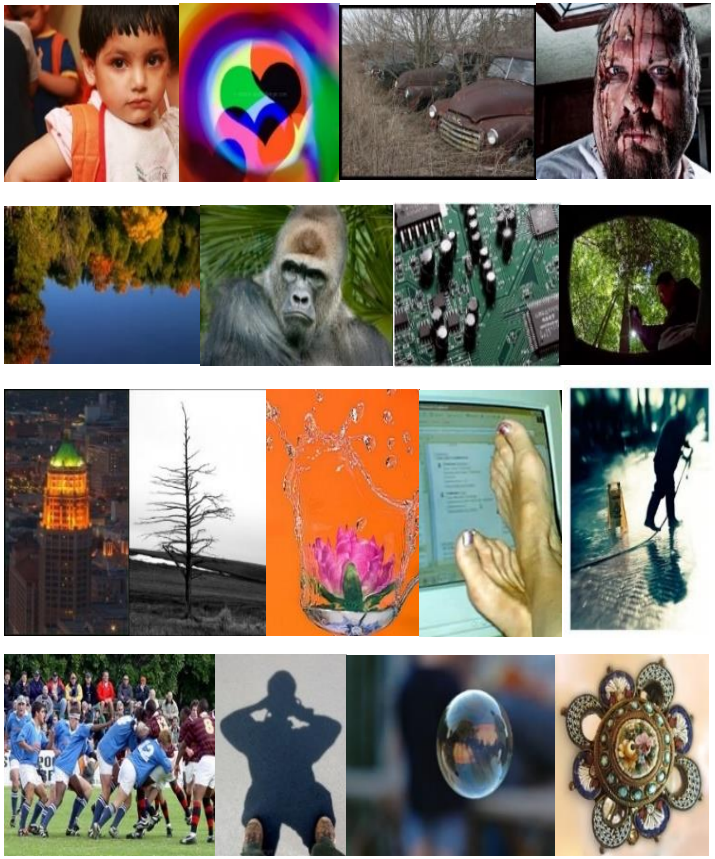
A Deep Learning Approach

- Dual-channeled CNN trained using relative learning
- Siamese Network characteristics (weight sharing) and hinge-loss function
- A custom data-set with relative labels – pairs formed based on aesthetic rating

SOURCE: Gattupalli *et al.* "A Computational Approach to Relative Aesthetics", *International Conference on Pattern Recognition* (ICPR) 2016.

Constructing a Useful Data Set 1/2

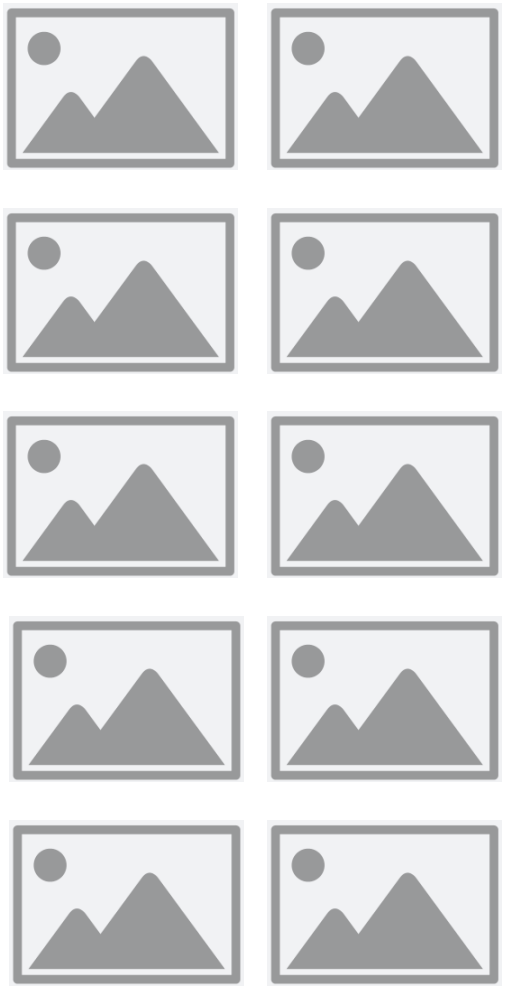
- Total of 250,000 images extracted from dpchallenge.com
- Challenges under which users post their submission
- Peers rate and a final winner is selected based on the average rating
- Belong to a wide variety of semantic categories



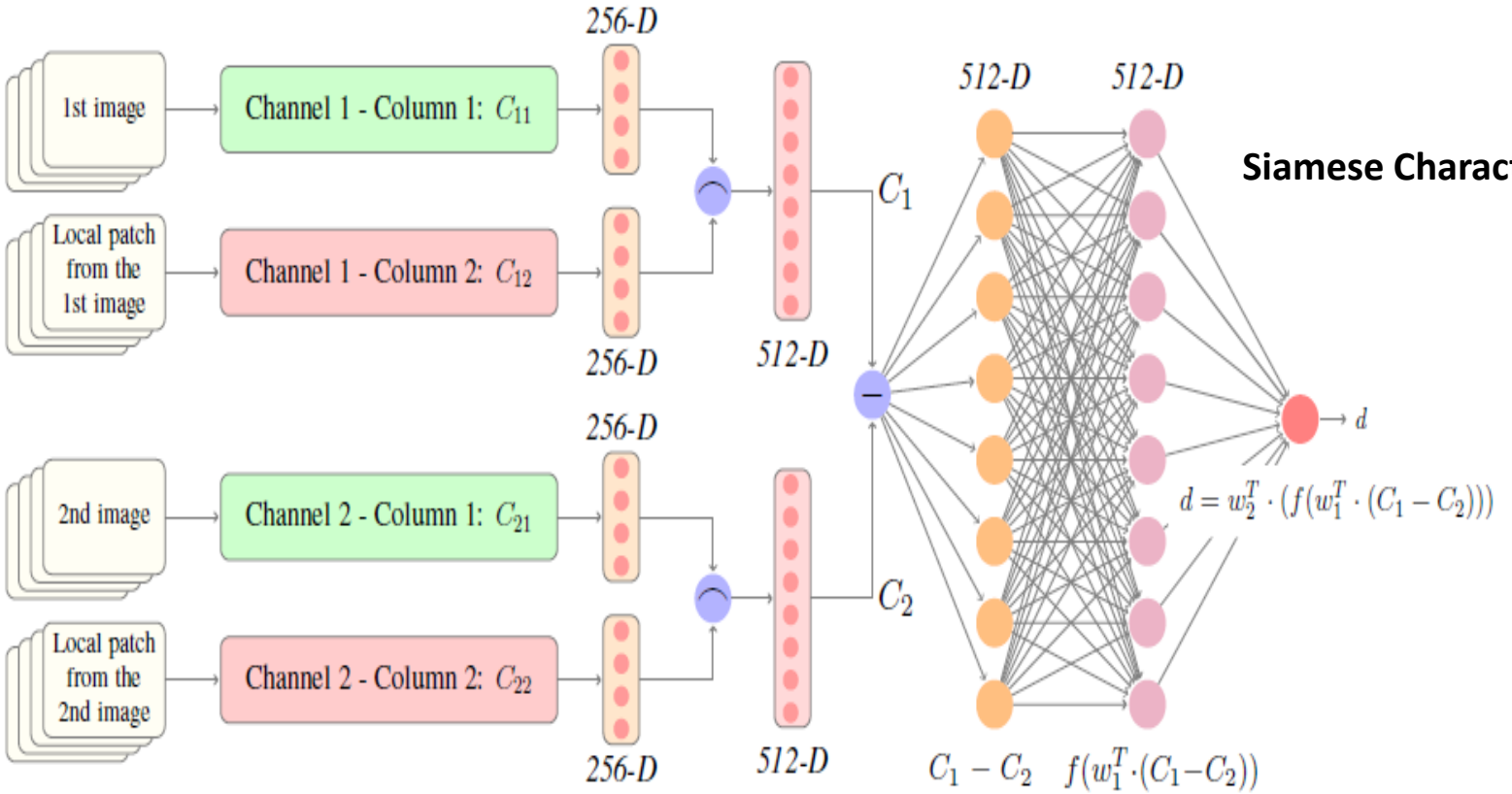
SOURCE: AVA Dataset

Constructing a Useful Data Set 2/2

- **The minimum gap between the average rating of the two images is one**
  - e.g., 3.4 and 4.5, 6.3, and 7.8
- **The maximum variance allowed between the ratings of different voters is 2.6**
- **Pick pairs from the same category only**
  - e.g., cannot compare an image of a car and a building



The Network Architecture & Other Characteristics



Siamese Characteristics

Padded Input	Conv	Max-pooling	Conv	Max-pooling	Conv	Conv	Dropout	Dense	Dropout	Dense	Dropout
$3 \times 230 \times 230$	$2, 64, 11, 2$	2×2	$1, 64, 5, 1$	2×2	$1, 64, 3, 1$	$-, 64, 3, 1$	0.5	1000	0.5	256	0.5

Further Implementation Details

- **Each channel contains two streams of processing: column 1 for global, and column 2 for local**

- **Global Patch**
  - e.g., rule of thirds, golden ration

- **Local Patch**
  - e.g., smoothness/graininess

The Loss Function

$$L = \max(0, \delta - y \cdot d(I_1, I_2)) \longrightarrow \text{Hinge Loss}$$

$$d(I_1, I_2) = f(C_1 - C_2)$$

where,

y = True label of the image pair,

i.e., 1 if $I_1 > I_2$ and

-1 otherwise

C_1, C_2 = Outputs of channel 1 and channel 2 respectively

Sample Results

Two ways of training

- Using binary labels
- Using relative labels

Tested for two tasks

- For Binary Classification task
- For Ranking task

Eight Experiments Total

	Ranking (custom test-set)	Ranking (standard test-set)	Classification (custom test-set)	Classification (standard test-set)
Base-line	62.21	65.87	**59.92**	69.18
Relative aesthetics	**70.51**	**76.77**	59.41	**71.60**

Exemplar Deep Learning Applications

Video-Based Inference

Ira A. Fulton Schools of Engineering
Arizona State University

Objective

Objective

Describe unique challenges in using deep networks for sequential data

Objective

Describe the difference between image-based and video-based classification tasks

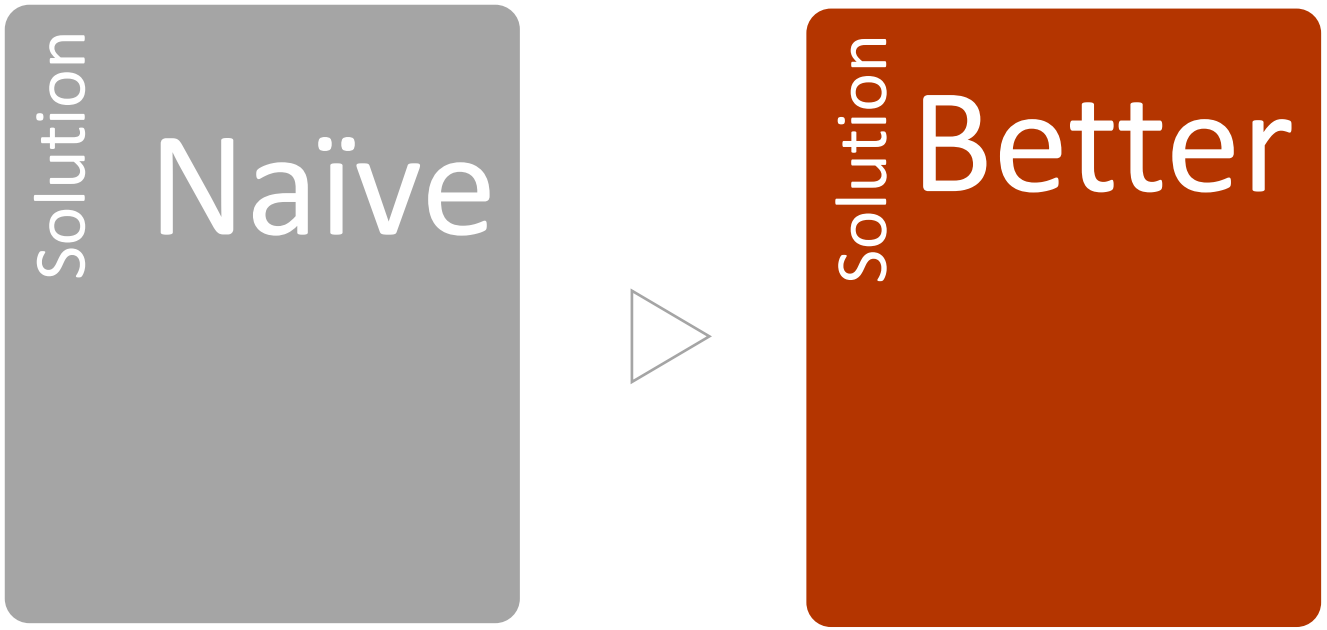
Objective

Explain the value of using video action recognition to contrast the difference between image-based and video-based classification tasks

Objective

Evaluate a video-based classification example using deep learning

Going from Image to Video



- Processing each frame of a video as an independent image and then aggregating the frame-level results
- Extracting spatio-temporal features and an inference task will be based on such features

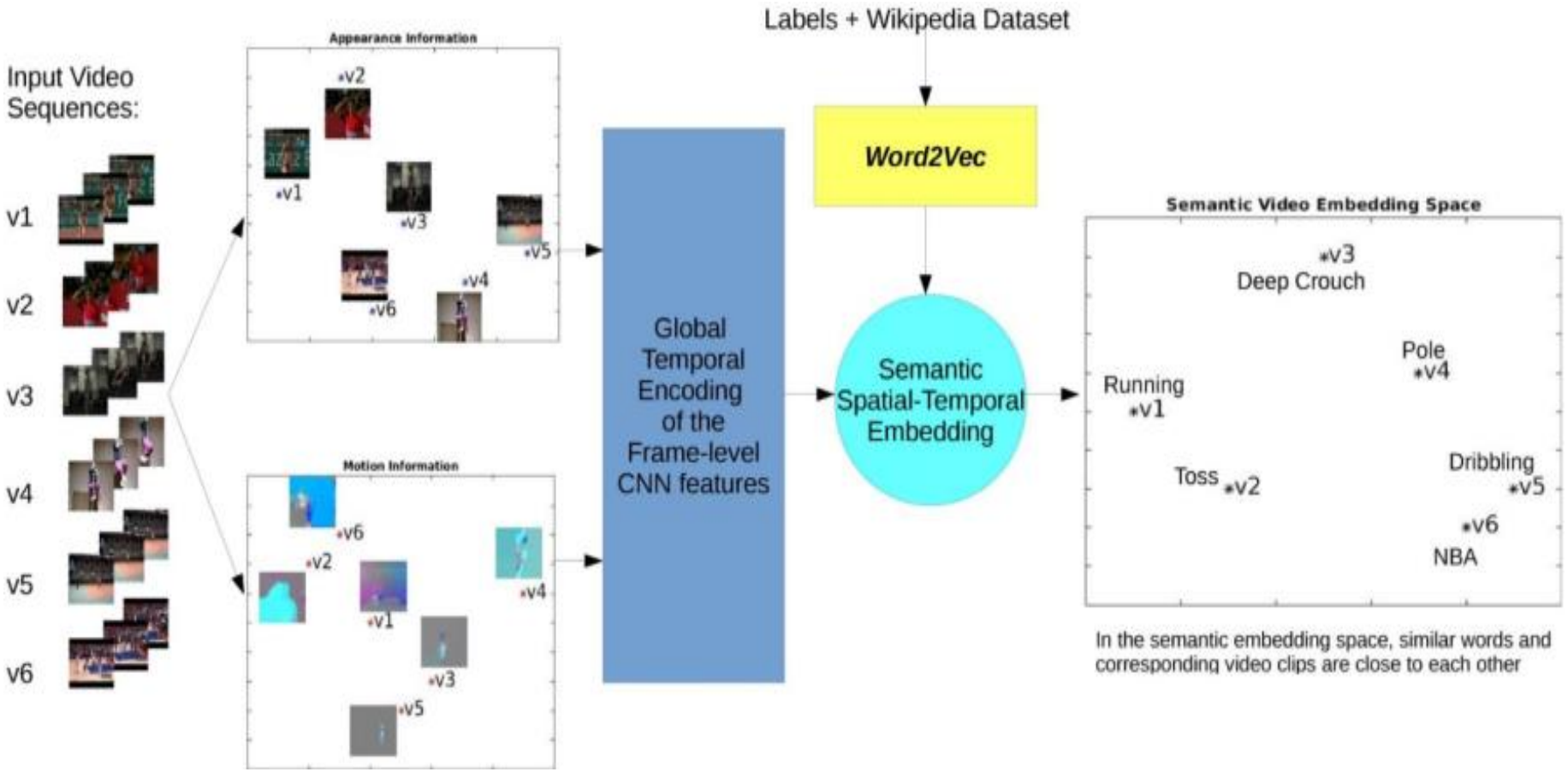
Video2Vec: Sample Applications

We examine a deep learning approach for finding video representations that naturally encode spatial-temporal semantics.

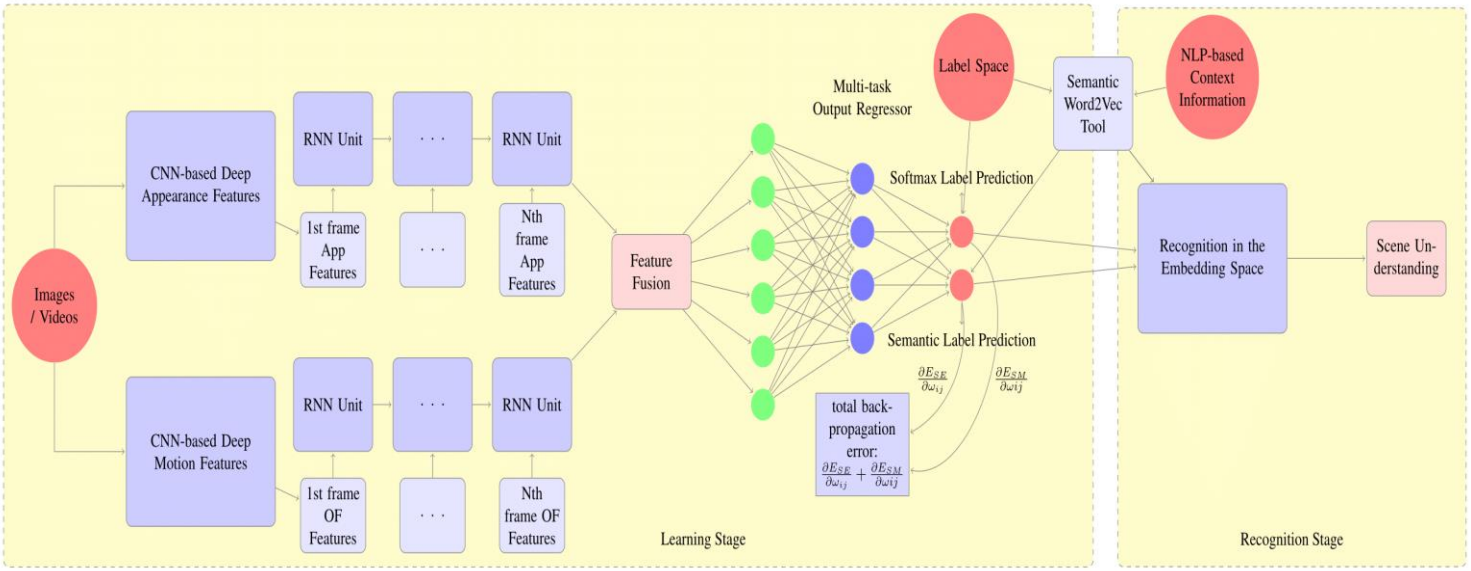
Mostly based on the following papers:

- Yikang Li, Sheng-hung Hu, Baoxin Li, “Recognizing Unseen Actions in a Domain-Adapted Embedding Space”, ICIP, Sep 2016.
- Yikang Li, Sheng-hung Hu, Baoxin Li, “*Video2Vec*: Learning Semantic Spatio-Temporal Embeddings for Video Representations”, ICPR, Dec 2016.

Video2Vec Deep Learning Model: Key Idea



Video2Vec Deep Learning Model: Implementation



- A two-stream CNN for extracting appearance and optical flow features
- RNNs for further global spatial-temporal encoding
- A MLP for final semantic embedding space

Applications of the Model

Visual tasks:

- Video Action Recognition
- Zero-Shot Learning
- Semantic Video Retrieval

Dataset: UCF101 dataset (13320 video clips from 101 categories; training/testing ratio is 7:3; the split list is provided by its own web)

Additional Implementation Details 1/4

Pretraining for the component models:

- **Pre-trained Spatial CNN Model:** VGG-f trained on ImageNet
- **Pre-trained OF CNN Model:** Flow-net trained on UCF Sports
- **Pre-trained Word2Vec Model:** Wikipedia corpus contained 1 billion words

Additional Implementation Details 2/4

Deep model parameter settings:

- **CNNs:** Pretrained model + the last layer (fc7) features (dimension: 4096x1)
- **RNNs:** Hidden layer size is 1024x1
- **MLP:** Input layer size (2048x1), hidden layer size (1200x1), output layer size (500x1)

Loss function:

- Hinge loss function for semantic embedding
- Softmax loss function for fine-tuning and classification

Additional Implementation Details 3/4

Video processing settings:

- Dense Optical Flow and RGB frames are extracted at 10fps.
- Building Video Sequence Mask for each training batch to make each sequence the same length.

Additional Implementation Details 4/4

Training parameter settings:

- Learning rate: initialized as 0.0001 and reduced by half each 15 epochs
- Total epoch: 60 epochs
- Batch size: 30 video clips
- Margin value for Hinge Loss function:
  - a. For zero-shot learning, 0.4
  - b. For video retrieval and action recognition, 0.55

Summaries of Key Results

Dataset: UCF101 dataset

Zero-shot learning results

- The model achieved state-of-the-art performance on ZSL even without any domain-adapted strategy.

Video action recognition

- The performance was on par with those with sophisticated fusion strategies or deeper networks.

Additional Results

- The task is to retrieve videos from training dataset by using query words that never appear in the training stage but share some information with training labels.

- The results show the top 10 retrieval video clips among video dataset.

Query Labels	Top10 Retrieve Results	Query Labels	Top10 Retrieve Results
NBA	Basketball Dunk (10)	Extreme	Rock Climbing Indoor (5), Uneven Bars (2), Soccer Juggling (2), Pole Vault (1)
Orchestra	Playing Cello (9), Playing Piano (1)	Tide	Cliff Diving (4), Surfing (2), Throw Discus (2), Sky Diving (1), Rafting (1)
Army	Military Parade (10)	India	Paying Tabla (4), Playing Sitar (2), Head Massage (1), Cricket Shot (1), Mixing (1)
Music	Playing Sitar (9), Playing Piano (1)	Celebrate	Military Parade (6), Long Jump (1), Band Marching (1), Ice Dancing (1), Blowing Candles (1)
Computer	Typing (10)	Home-run	Baseball Pitch (5), Basketball Dunk (3), Field Hockey Penalty (1), Frisbee Catch (1)
Park	Biking (9), Golf Swing (1)	Boat	Kayaking (4), Rafting (2), Rowing (2), Cliff Diving (1), Push Ups (1)
Summit	Cliff Diving (7), Skiing (2), Rope Climbing (1)	Toy	Yo-yo (4), Nun chucks (4), Pull Ups (1), Juggling Ball (1)
School	Skate Boarding (10)	Snow	Skiing (2), Ice Dancing (2), Cricket Bowling (1), Pole Vault (1), Blowing Candles (1), Blow Dry Hair (1), Rafting (1), Sky Diving (1)
Park	Biking (9), Golf Swing (1)	Acrobatics	Juggling Balls (5), Soccer Juggling (5)
Water	kayaking (10)	Ocean	Cliff Diving (4), Sky Diving (3), Kayaking (2), Rafting (1)
FIFA	Soccer Penalty (8), Soccer Juggling (2)	Hurl	Throw Discus (2), Mopping Floor (2), Baby Crawling (1), Javelin Throw (1), Cricket Shot (1), Blowing Candles (1), Pull Ups (1)
Club	Golf Swing (8), Soccer Juggling (2)	Hiking	Biking (5), Kayaking (4), Rafting (1)
Nature	Tai Chi (7), Hammering (2), Walking with Dog (1)	Swim	Diving (5), kayaking (3), Cricket Bowling (1), Sky Diving (1)
Beethoven	Playing Cello (8), Playing Voilin (2)	Jogging	Biking (5), Skate Boarding (2), Soccer Juggling (1), Skiing (1), Ice Dancing (1)
Classical	Playing Cello (7), Playing Voilin (3)	Foam	Blowing Candles (7), Pull Ups (1), Rope Climbing (1), Juggling Balls (1)
Yankees	Baseball Pitch (10)	Hip-hop	Trampoline Jumping (6), Swing (4)
Duel	Boxing Punching Bag (8), Punch(2)	Scramble	Pull Ups (6), Trampoline Jumping (2), Rope Climbing (1), Cricket Shot (1)
Lifting	Body Weight Squats (4), Rope Climbing (4), Pull Ups (2)	Mat	Rope Climbing (4), Pommel Horse (3), Trampoline Jumping (2), Javelin Throw (1)
Martial	Fencing (3), Archery (3), Boxing Punching Bag (3), Balance Beam (1)	Parachuting	Diving (6), Cricket Bowling (2), Hand Stand Walking (1), Sky Diving (1)
Tumbling	Trampoline Jumping (8), Throw Discus (1), Frisbee Catch (1)	Hunting	Horse Riding (3), Kayaking (3), Nun chucks (3), Frisbee Catch (1)

Exemplar Deep Learning Applications

Generative Adversarial Networks (GANs)

Ira A. Fulton Schools of Engineering
Arizona State University

Objective

Objective

Describe basic concepts and architecture for GANs

Objective

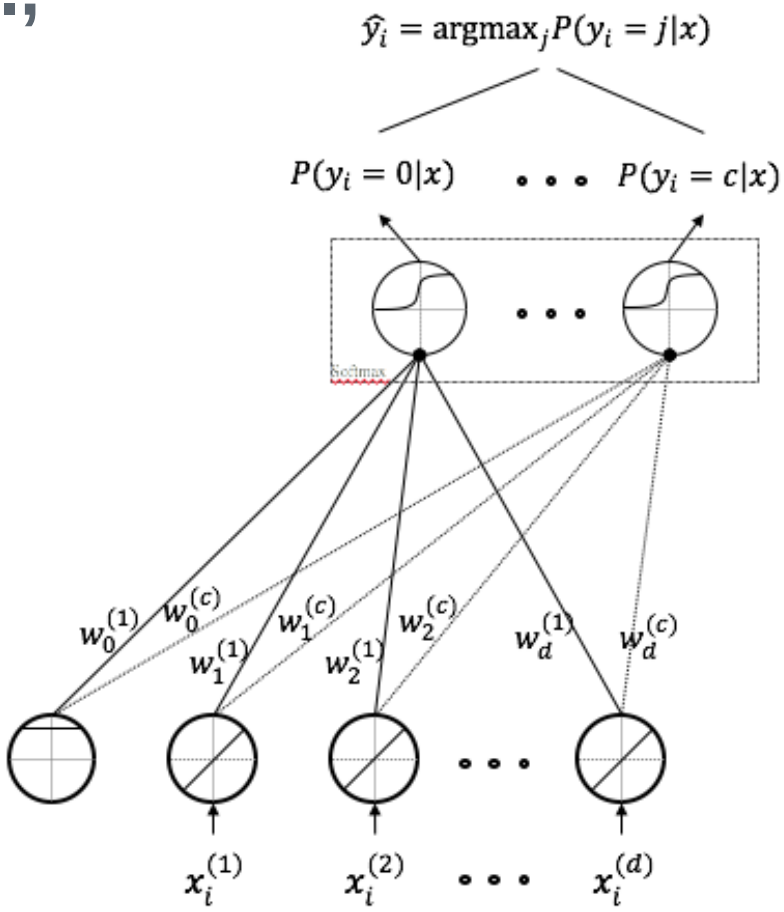
Illustrate variants of GANs and their applications

Generative Adversarial Networks (GANs)

- **Proposed in 2014 by Goodfellow *et al.***
- **An architecture with two neural networks gaming against each other.**
  - One attempting to learn a *generative model*
- **Many variants have been proposed since the initial model.**

Discriminative vs Generative Models 1/4

- **Discriminative models: E.g., the familiar MLP**
  - Given $\{(x_i, y_i)\}$, to learn $P(y_i|x)$
- **More generally, we try to learn a *posterior distribution* of y given x, *p*(y|x)**
  - Usually reduced to posterior probabilities for classification problems



➔ See also earlier discussion on Naïve Bayes vs Logistic Regression.

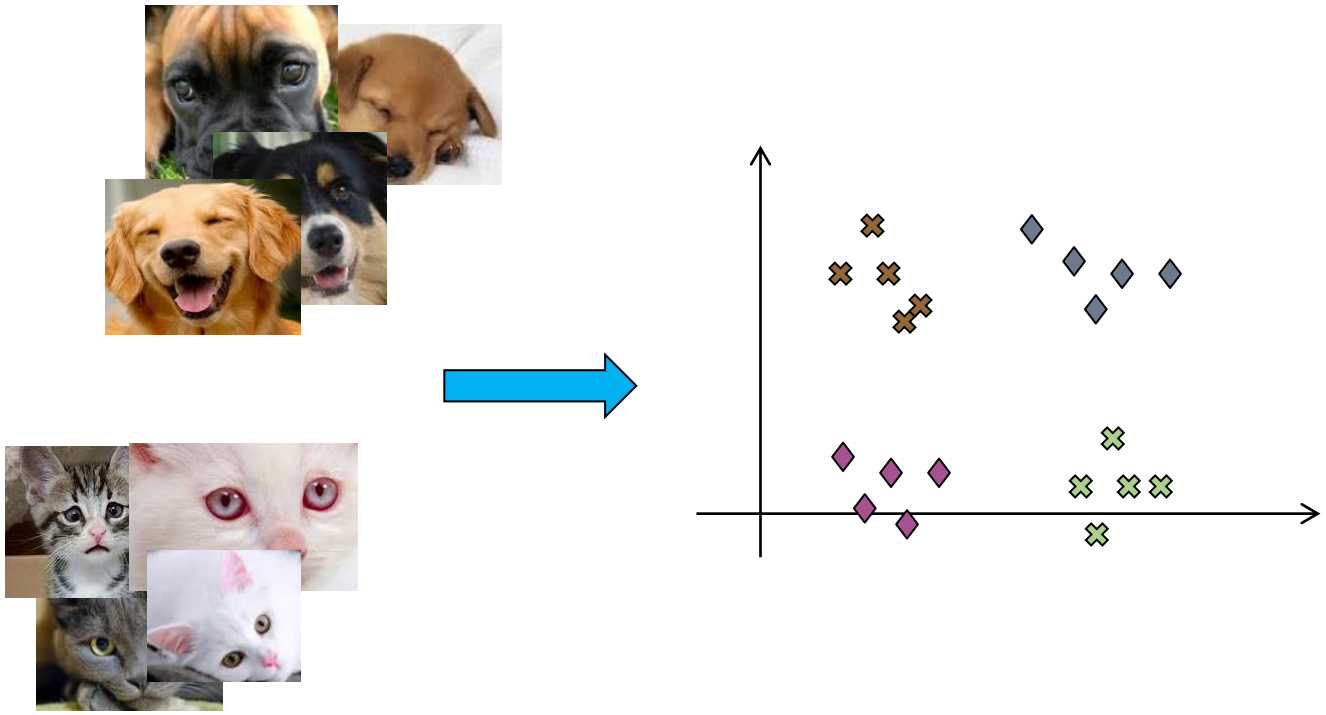
Discriminative vs Generative Models 2/4

- **Generative models think the other direction: how to generate x given y**
  - E.g., x_i=? if y_i = 2?

- **More generally, we try to learn a *conditional distribution* of x given y, *p*(x|y)**

Discriminative vs Generative Models 3/4

Illustrating the ideas



Discriminative vs Generative Models 4/4

Estimating *p*(x|y) (or, in general any *p*(x), if we drop y by assuming it is given)

- Explicit density estimation: assuming some parametric or non-parametric models.
- Implicit density estimation: learn (essentially equivalent) models that may create good samples (as if from the "true" model), without explicitly defining the true model.

→ **GAN is such an approach**

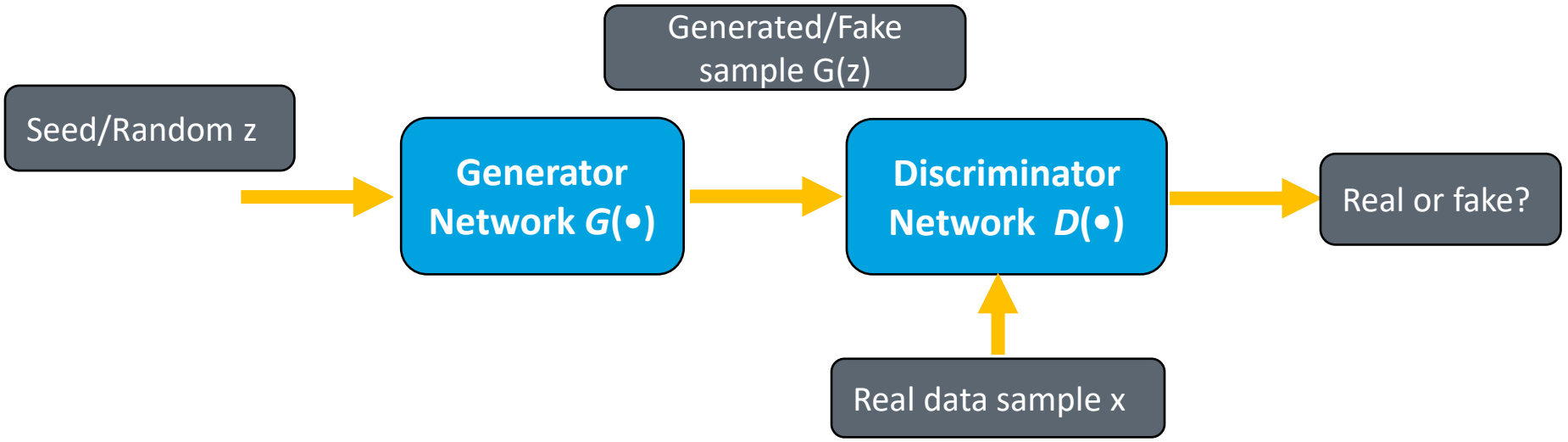
Discriminative vs Generative Models 4/4

Estimating *p*(x|y) (or, in general any *p*(x), if we drop y by assuming it is given)

- Explicit density estimation: assuming some parametric or non-parametric models.
- Implicit density estimation: learn (essentially equivalent) models that may create good samples (as if from the "true" model), without explicitly defining the true model.

→ **GAN is such an approach**

Basic GAN Architecture



- **Objective of the Discriminator Network:**

  making $D(x) \rightarrow 1$, $D(G(z)) \rightarrow 0$

- **Objective of the Generator Network:**

  making $D(G(z)) \rightarrow 1$

Basic GAN Training Algorithm

for number of training iterations **do**

  for k steps **do**

    Sample minibatch of m noise samples $\{\boldsymbol{z}^{(1)}, \ldots, \boldsymbol{z}^{(m)}\}$ from noise distribution $p_g(\boldsymbol{z})$

    Sample minibatch of m examples $\{\boldsymbol{x}^{(1)}, \ldots, \boldsymbol{x}^{(m)}\}$ from data distribution $p_{data}(\boldsymbol{x})$

    Update the discriminator by ascending its stochastic gradient:

$$\nabla_{\theta_d} \frac{1}{m} \sum_{i=1}^{m} [\log D\left(\boldsymbol{x}^{(i)}\right) + \log\left(1 - D\left(G\left(\boldsymbol{z}^{(i)}\right)\right)\right)]$$

  end for

    Sample minibatch of m noise samples $\{\boldsymbol{z}^{(1)}, \ldots, \boldsymbol{z}^{(m)}\}$ from noise distribution $p_g(\boldsymbol{z})$

    Update the generator by descending its stochastic gradient

$$\nabla_{\theta_g} \frac{1}{m} \sum_{i=1}^{m} \log\left(1 - D\left(G\left(\boldsymbol{z}^{(i)}\right)\right)\right)$$

end for

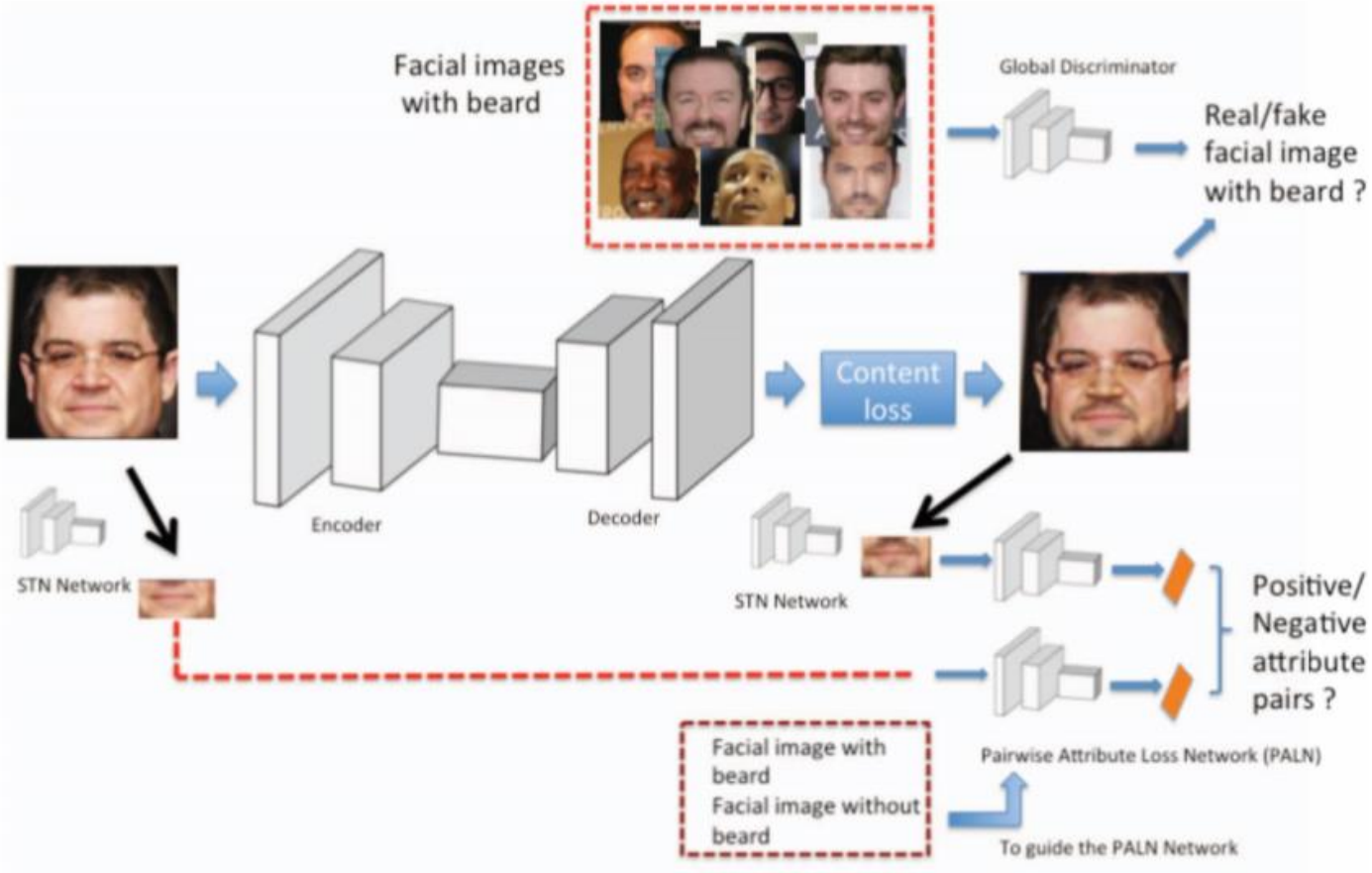
θ_d and θ_g are the parameters of the discriminator and generator respectively.

SOURCE: Goodfellow *et al.* https://arxiv.org/pdf/1406.2661.pdf

Applications of GAN

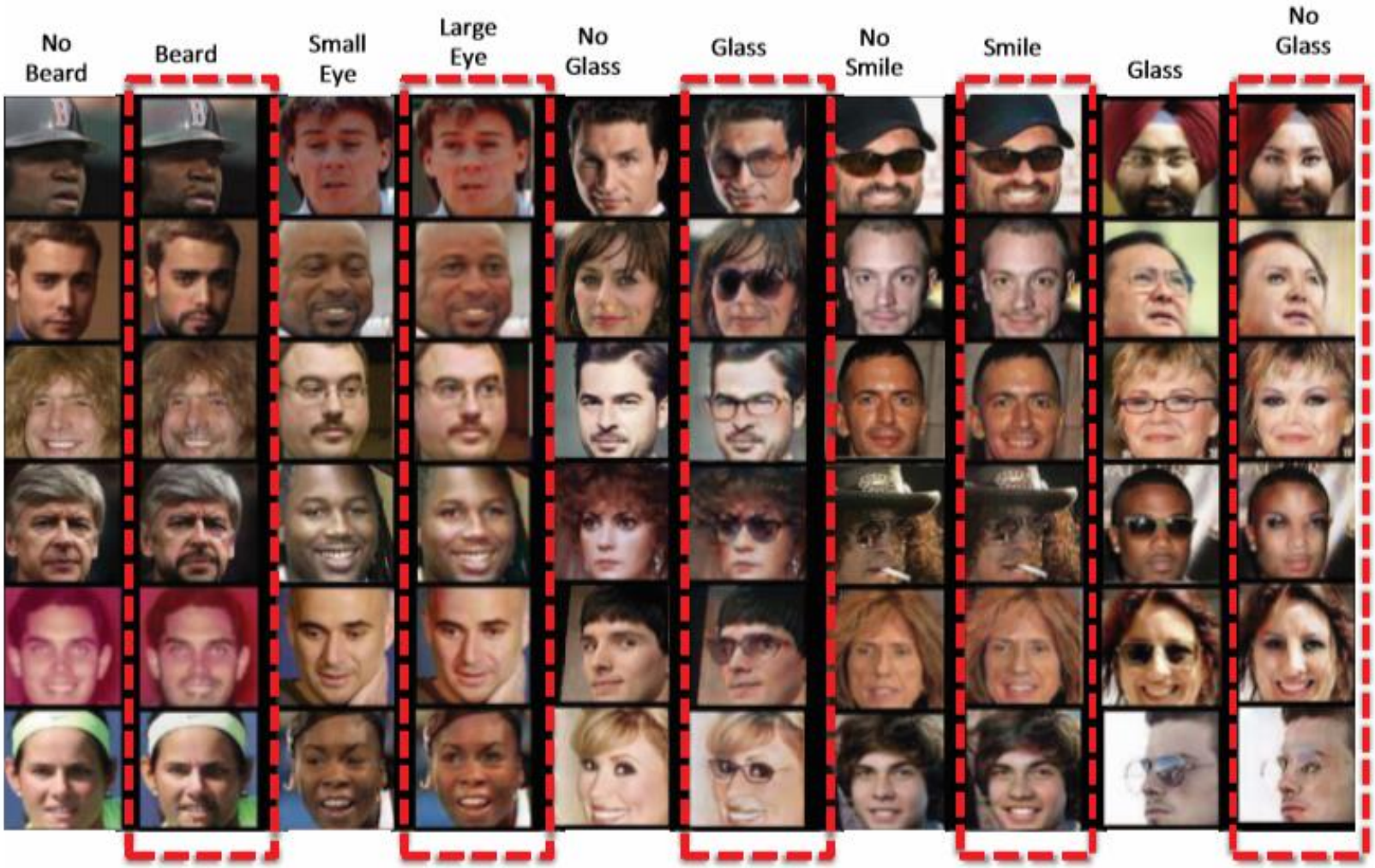
- **GAN enabled many novel/interesting/fun applications.**
- **Many GAN-based models have been proposed, following the initial paper.**
- **Consider one example: Facial attribute manipulation**
  - Y. Wang *et al. "Weakly Supervised Facial Attribute Manipulation via Deep Adversarial Network"*, WACV 2018.

Facial Attribute Manipulation 1/2



SOURCE: Y. Wang *et al. "Weakly Supervised Facial Attribute Manipulation via Deep Adversarial Network"*, WACV 2018.

Facial Attribute Manipulation 2/2



SOURCE: Y. Wang *et al.* *"Weakly Supervised Facial Attribute Manipulation via Deep Adversarial Network"*, WACV 2018.