Title: Dataset Augmentation in Feature Space - Graham Taylor

Date: Sep 15, 2017 11:00 AM

URL: http://pirsa.org/17090063

Abstract: <p>Dataset augmentation, the practice of applying a wide array of domain-specific transformations to synthetically expand a training set, is a standard tool in supervised learning. While effective in tasks such as visual recognition, the set of transformations must be carefully designed, implemented, and tested for every new domain, limiting its re-use and generality. In this talk, I will describe recent methods that transform data not in input space, but in a feature space found by unsupervised learning. We start with data points mapped to a learned feature space and apply simple transformations such as adding noise, interpolating, or extrapolating between them. Working in the space of context vectors generated by sequence-to-sequence recurrent neural networks, this simple and domain-agnostic technique is demonstrated to be effective for both static and sequential data. </p>

<p> </p>

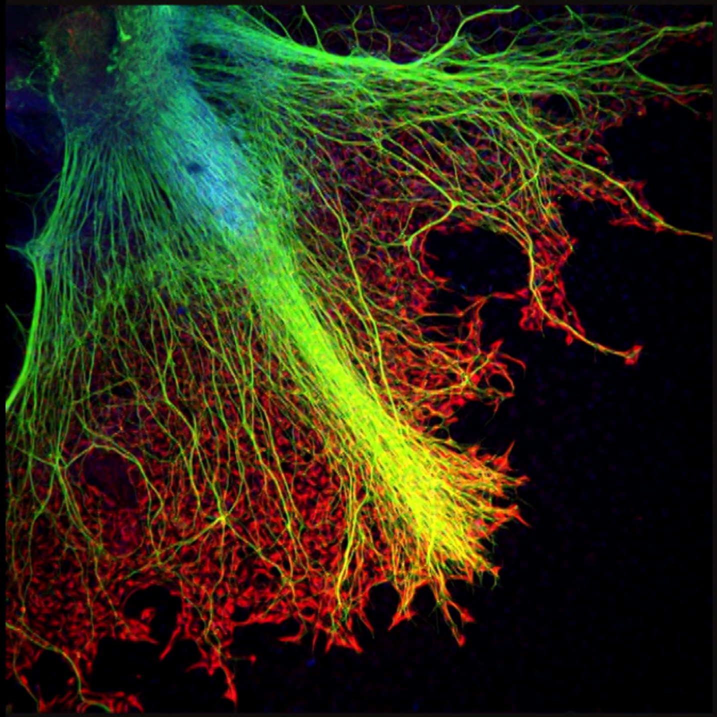
<p> </p>

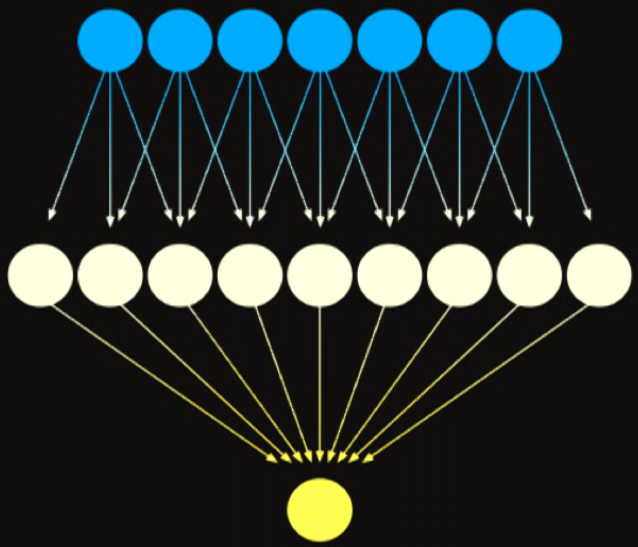
<p>Bio: Graham Taylor is an Associate Professor at the University of Guelph where he leads the Machine Learning Research Group. He is a member of the Vector Insitute for Artificial Intelligence and is an Azrieli Global Scholar with the Canadian Institute for Advanced Research. He received his PhD in Computer Science from the University of Toronto in 2009, where he was advised by Geoffrey Hinton and Sam Roweis. He spent two years as a postdoc at the Courant Institute of Mathematical Sciences, New York University working with Chris Bregler, Rob Fergus, and Yann LeCun. </p>

<p> </p>

<p>Dr. Taylor's research focuses on statistical machine learning, with an emphasis on deep learning and sequential data. Much of his work has focused on "seeing people" in images and video, for example, activity and gesture recognition, pose estimation, emotion recognition, and biometrics.</p>

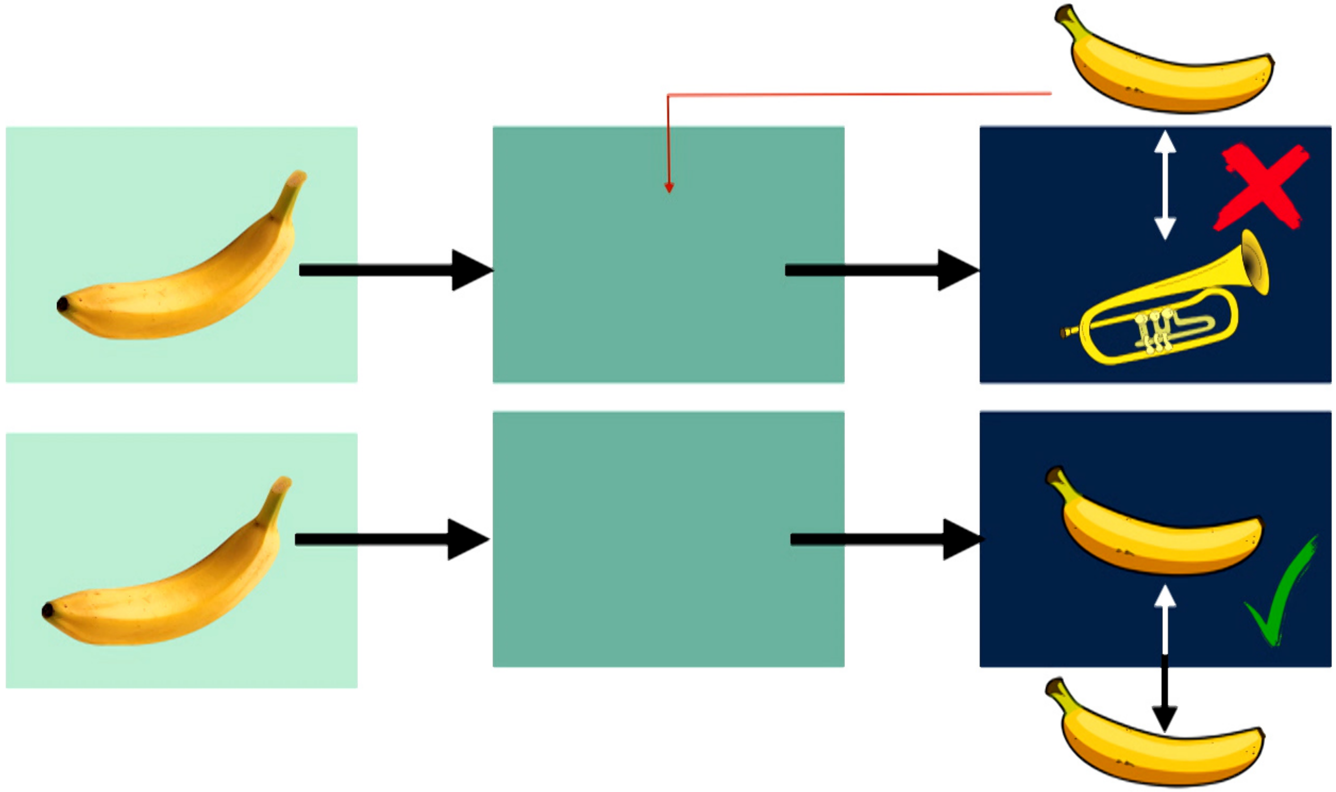

EXIT
72
RIGHT
55
WEST
20
EAST
EXIT ONLY
via bloomberg.com





via nikonsmallworld.com
cifar.ca

via imgur.com



15 Sep 2017 / 9
Perimeter Institute · Dataset Augmentation in Feature Space / G Taylor

via imgur.com

Learning architectures

- Romanticized notion of DL - end of feature engineering
- Feature engineering has decreased
- Architectures have become more complex

« Smerity.com

In deep learning, architecture engineering is the new feature engineering

June 11, 2016

Two of the most important aspects of machine learning models are **feature extraction** and **feature engineering**. Those features are what supply relevant information to the machine learning models.

Representing the word **overfitting** using various feature representations:

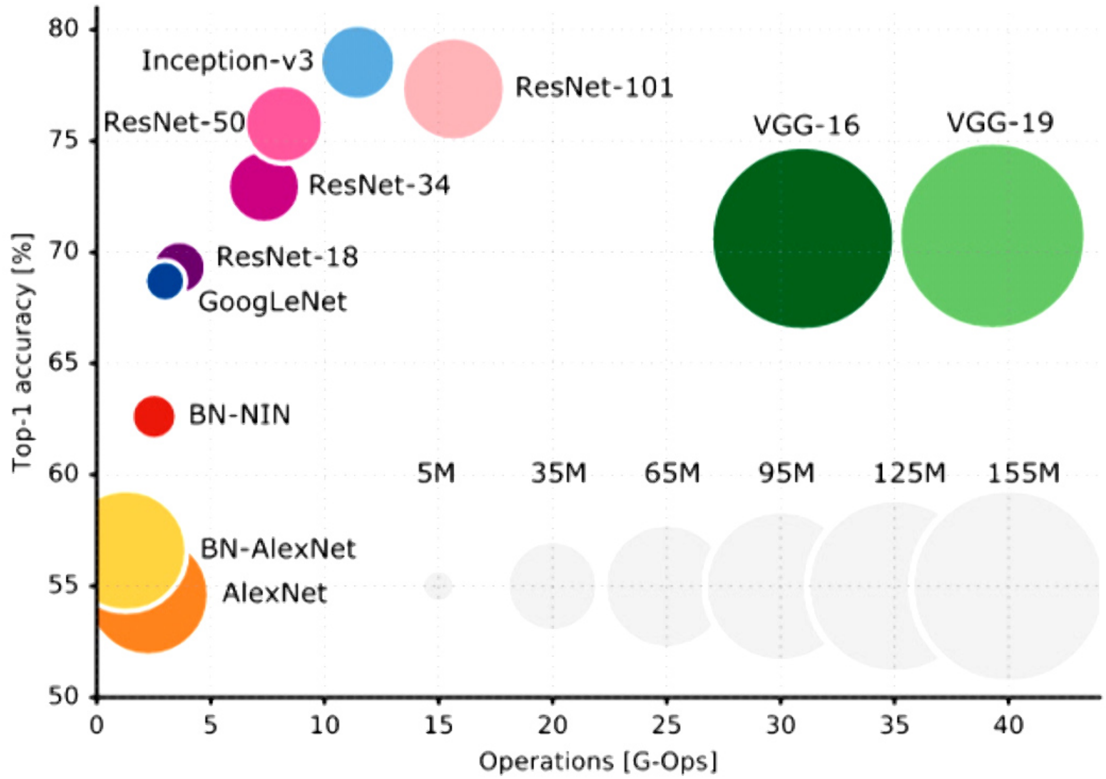
- Morphological = [(prefix, **over-**), (root, **fit**), (suffix=imperfect tense, **-ing**)]
- Unigrams = ['o', 'v', 'e', 'r', 'f', 'i', 't', 't', 'i', 'n', 'g']
- Bigrams = ['ov', 've', 'er', 'rf', 'fi', 'it', 'tt', 'ti', 'in', 'ng']
- Trigrams = ['ove', 'ver', 'erf', 'rfi', 'fit', 'itt', 'tti', 'tin', 'ing']
- One-hot = [0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]
- **Word vector** = [-0.26, 0.34, 0.48, -0.06, 0.16, 0.11, 0.13, -0.15, 0.47, -0.49, 0.07, -0.39, -0.13, -0.15, 0.06, 0.09]
- ...

If the features are few or irrelevant, your model may have a hard time making any useful predictions. If there are too many features, your model will be slow and likely overfit.

Humans don't necessarily know what feature representation are best for a given task. Even if they do, relying on feature engineering means that a human is always in the loop. This is a far cry from the future we might want, where you can throw any dataset at a

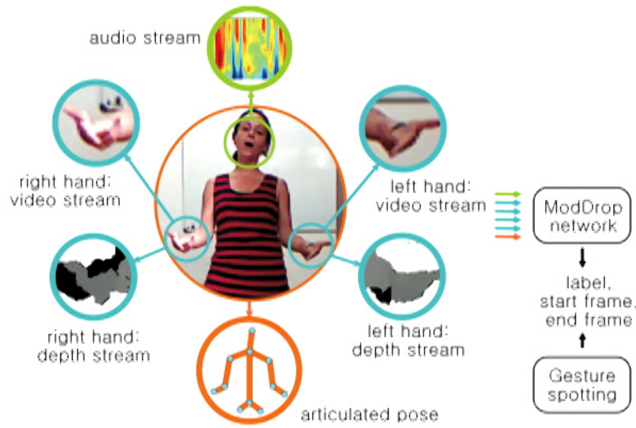
http://smerity.com/articles/2016/architectures_are_the_new_feature_engineering.html

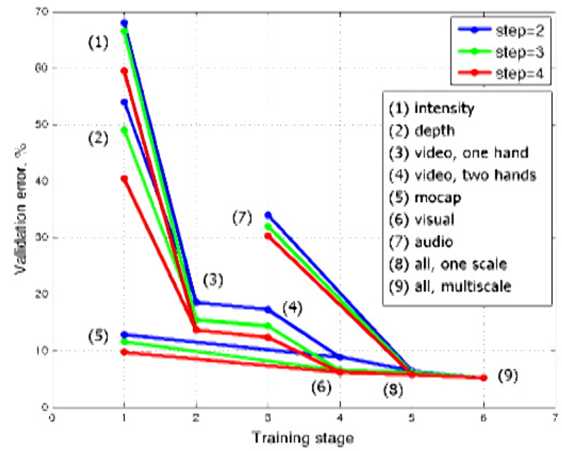
Saturation?



Eugenio Culurciello's blog:
https://culurciello.github.io/tech/2016/06/04/nets.html

Engineering multi-modal fusion

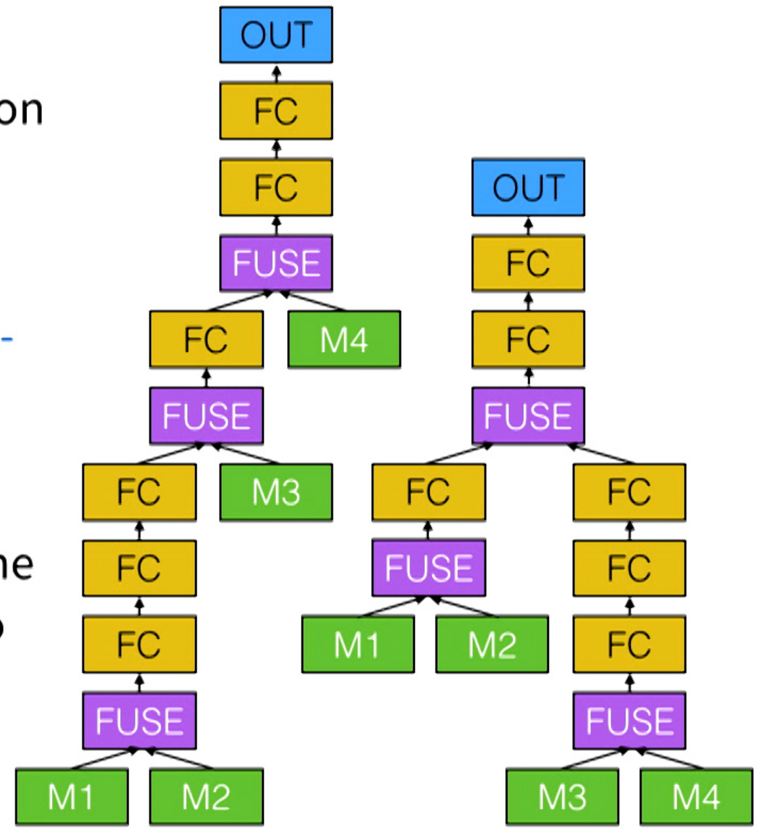




Neverova, Wolf, Taylor, Nebout (2016)
"ModDrop: adaptive multi-modal gesture recognition "

BayesOpt over Architectures

- Treat each aspect of fusion as a hyperparameter
- Pro: simple implementation— graph-induced kernel enables use of off-the-shelf tools
- Con: slow— must train the model to convergence to make a single update

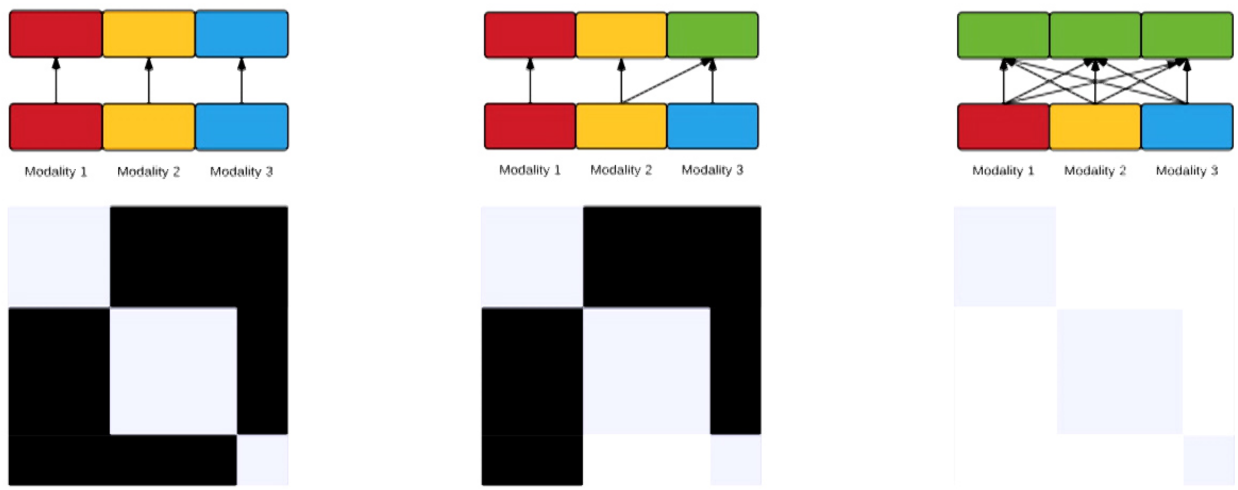


Ramachandram, Lisicki, Shields, Amer, Taylor (2017)
"Structure optimization for deep multimodal fusion networks"

Modout: Learning Architectures by Stochastic Regularization

Fan Li
1986-2016

Three typical fusion architectures achievable by Modout, with corresponding weight masks:



Independent | Fused | Fully-connected

Li, Neverova, Wolf, Taylor (2017)
“Modout: learning multi-modal architectures by stochastic regularization”

via topleftpixel.com

ML for Agri-Food

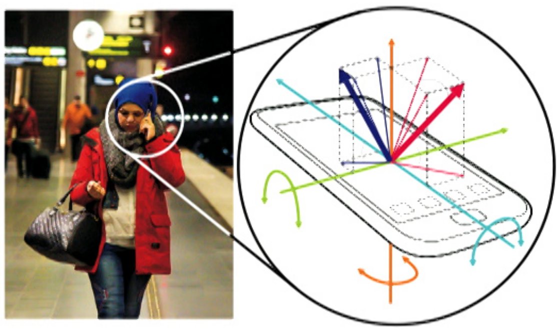
Detect pests → Report risk

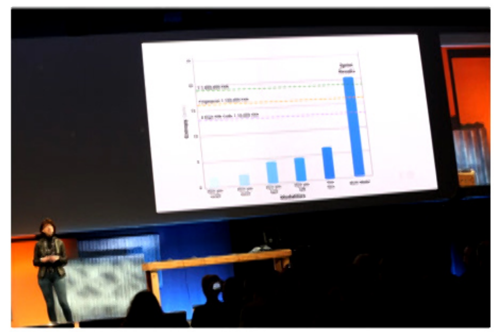
Ding and Taylor (2016)
"Automatic moth detection from trap images for pest management"

Google

Natalia Neverova

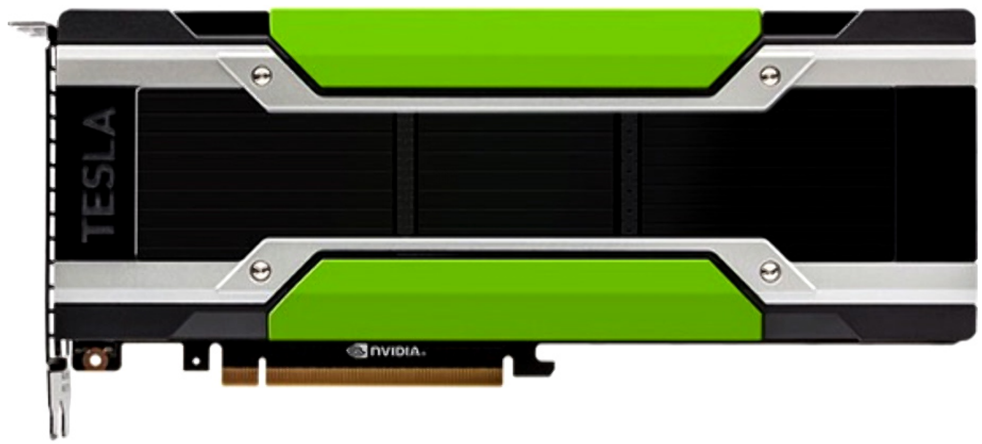
Continuous Authentication on Mobile





Neverova et al. (2016)
“Learning human identity from motion patterns”

3. Acceleration

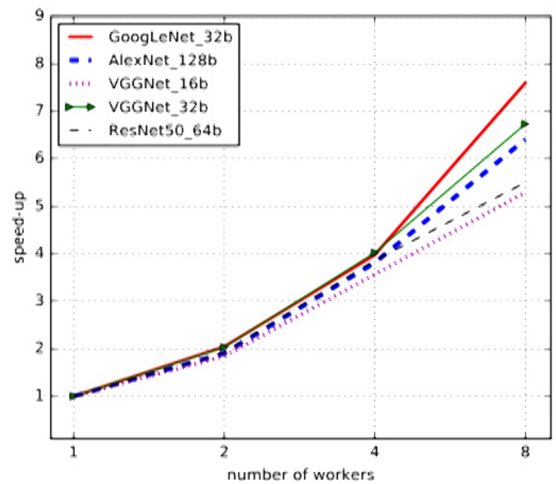


Multi-node GPU Parallelism

- Complete and **scalable** training framework for accelerating training of deep learning models
  - Python/Theano-based (distribute your Theano model!)
  - Key Technologies: NVIDIA RDMA, CUDA-aware MPI for IPC, LMDB
  - Multiple parallelism strategies investigated: BSP, ASP/Downpour, SSP, Elastic Averaging

He Ma

Fei Mao



Ma, Mao, Taylor (2016)
"Theano-MPI: a Theano-based Distributed Training Framework"

FPGA_Caffe Project

- Aim: make FPGA-based acceleration accessible to ML Researchers and Data Scientists
- Extension of popular Caffe deep learning framework
- Support for CPU, GPU, and FPGA
- Support for both CUDA and OpenCL

Griffin Lacey

HUAWEI

DiCecco, Lacey, Vasiljevic, Chow, Taylor, Areibi (2016)
"Caffeinated FPGAs: FPGA framework for CNNs"

Image credit: https://github.com/jcjohnson/neural-style

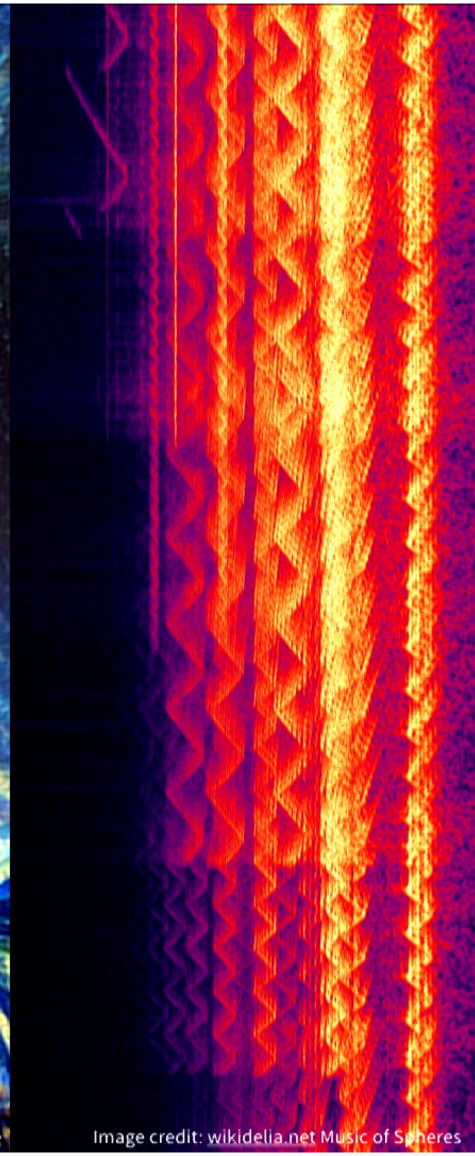
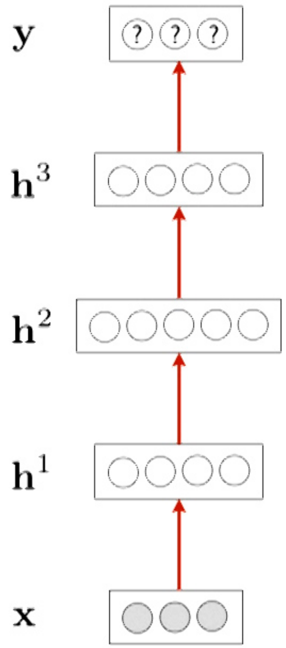


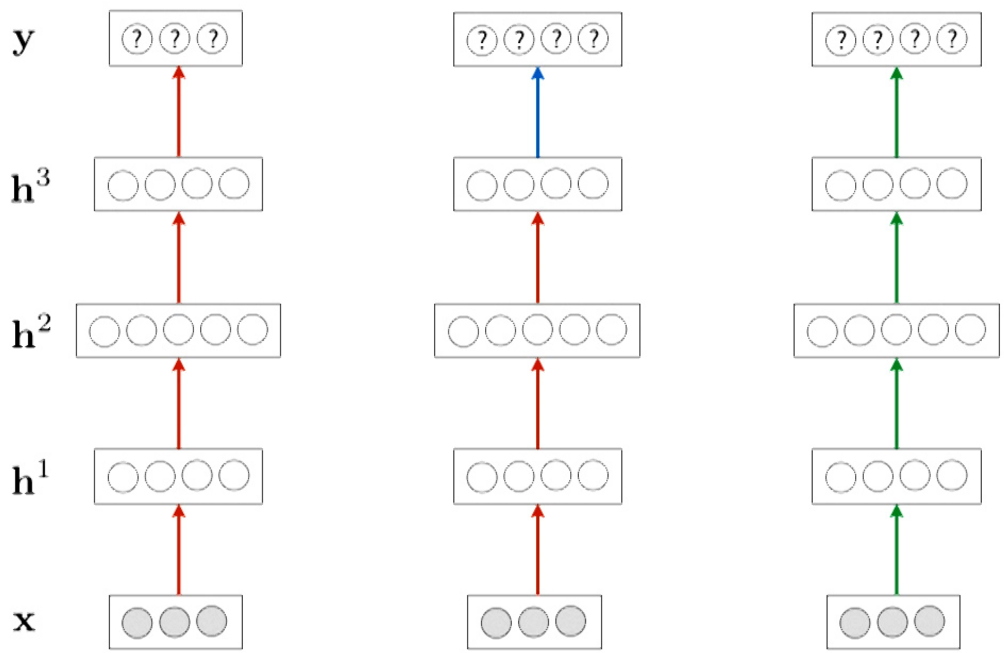
Image credit: wikidelia.net Music of Spheres

The pre-training trick



"Pre-train" net
on large dataset

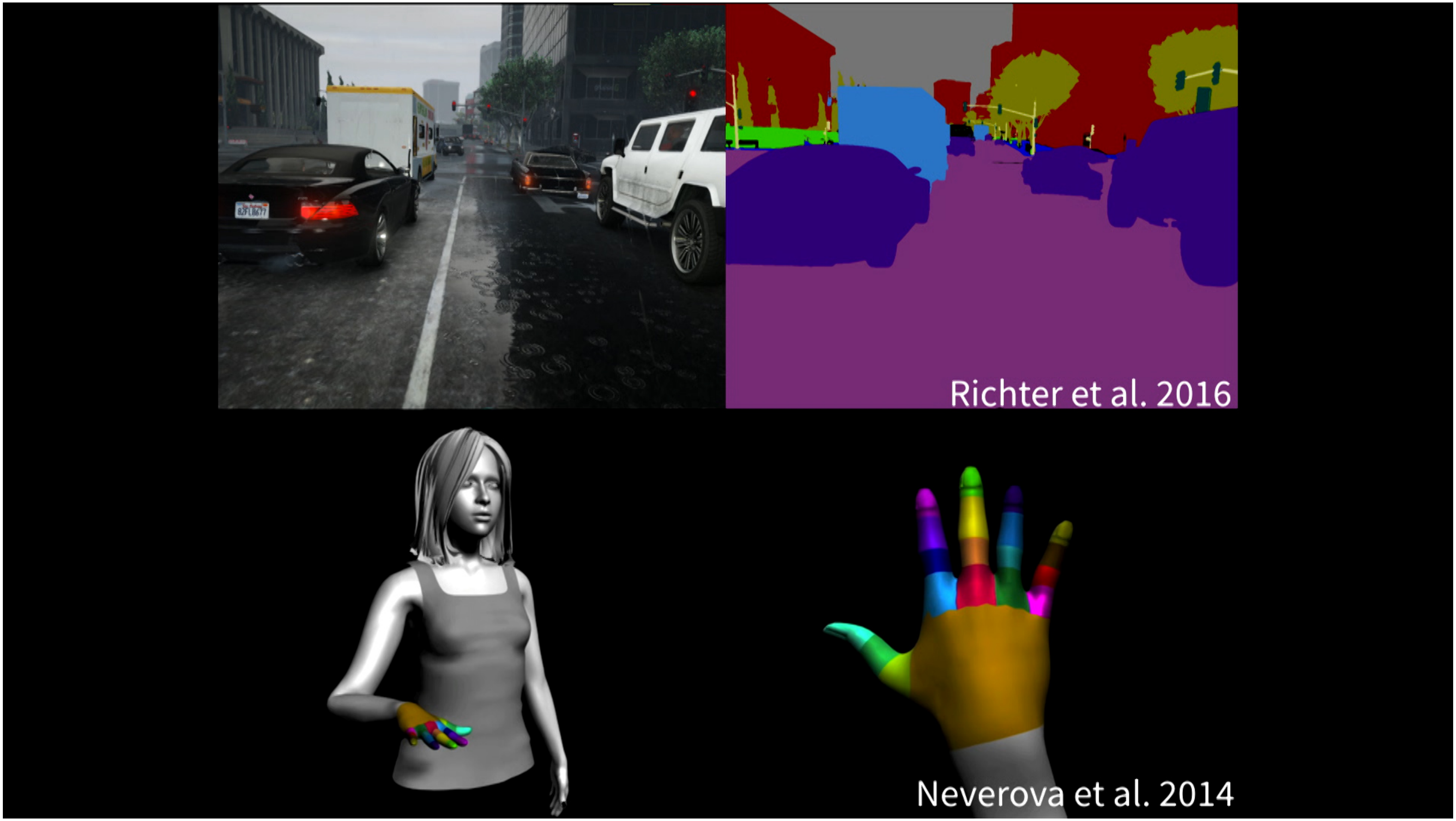
The pre-training trick



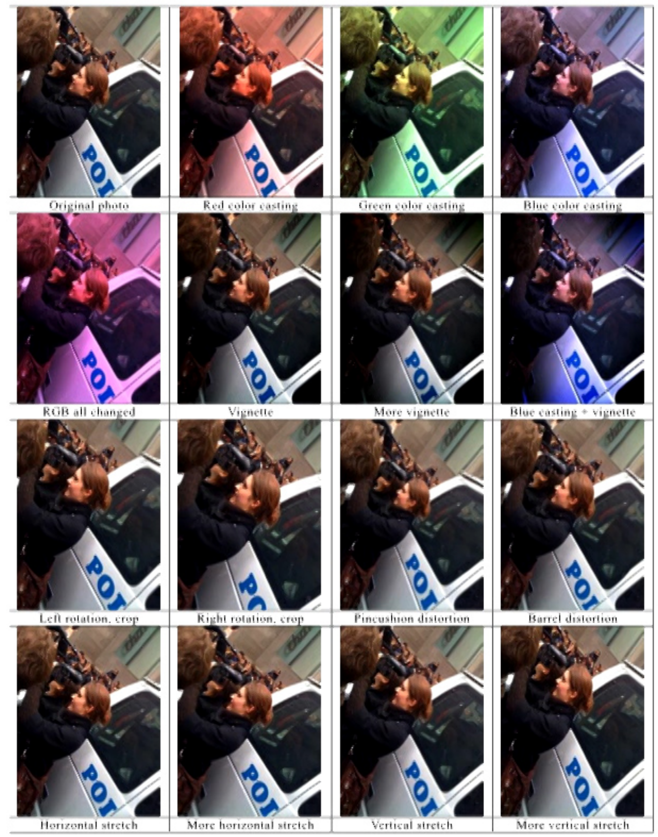
"Pre-train" net on large dataset

Replace top layers & initialize

"Fine-tune" net on limited dataset

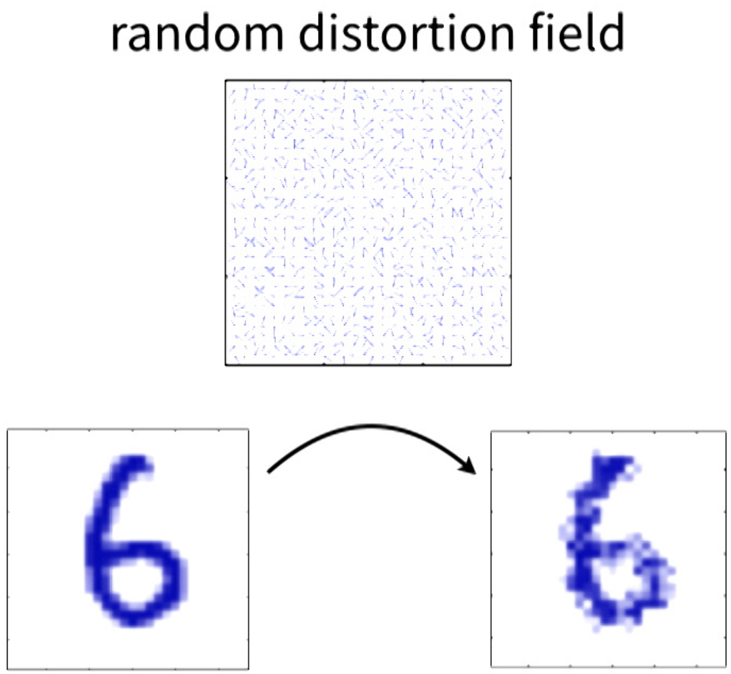

Richter et al. 2016
Neverova et al. 2014

Dataset augmentation



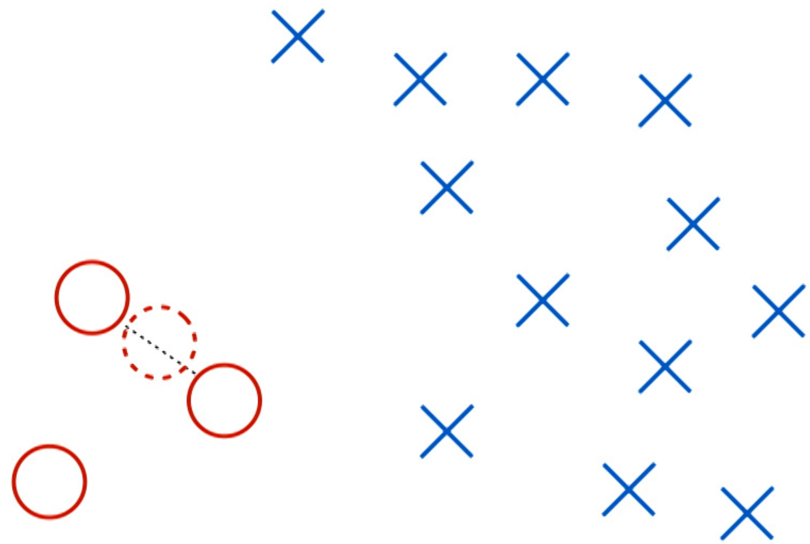
via baidu.com

More general transformations…
Still domain-specific



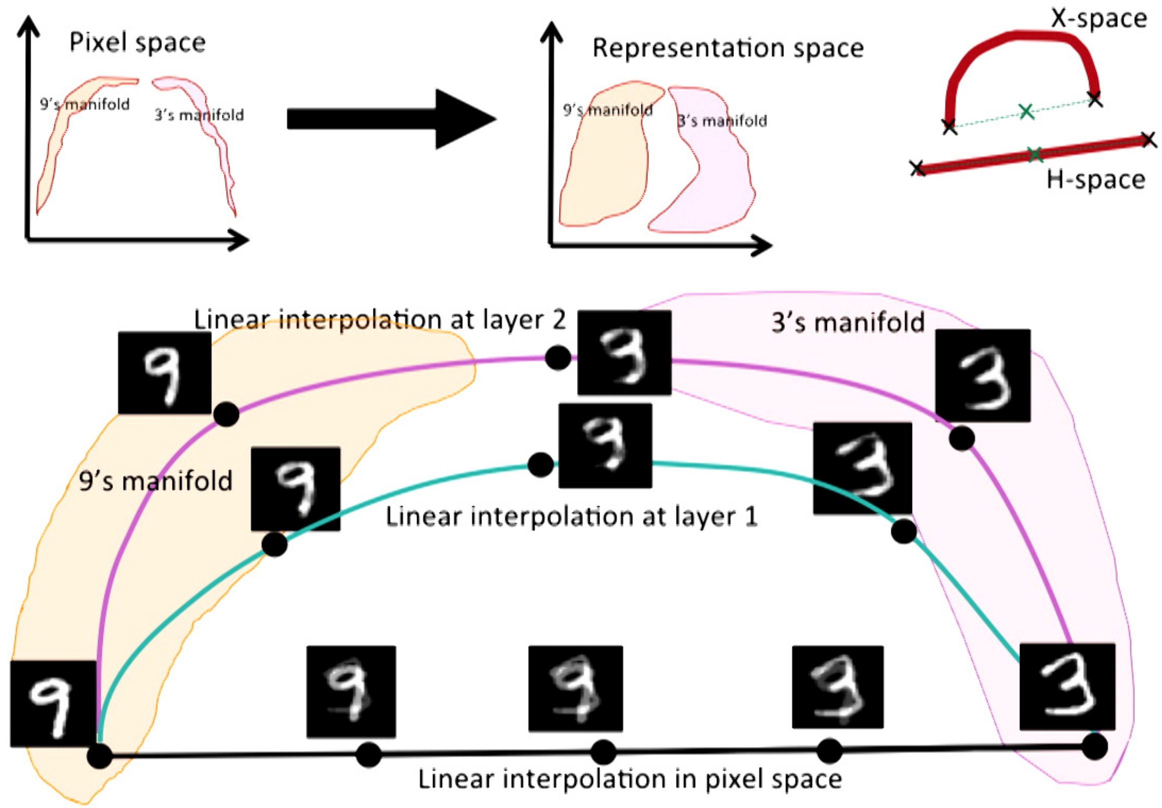
Bishop text via Hugo Larochelle

What about interpolation?



See: Chawla et al. Synthetic Minority Oversampling Technique (2002)

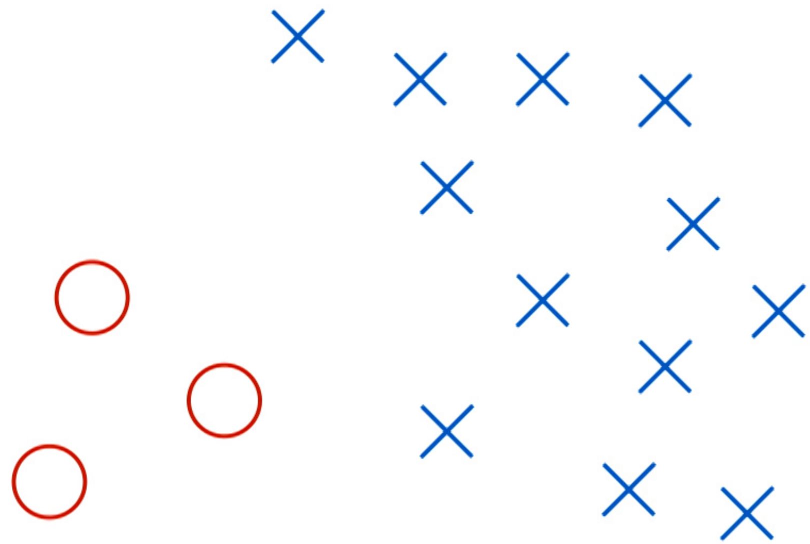
The problem (and proposal)



15 Sep 2017 / 36
Perimeter Institute - Dataset Augmentation in Feature Space / G Taylor

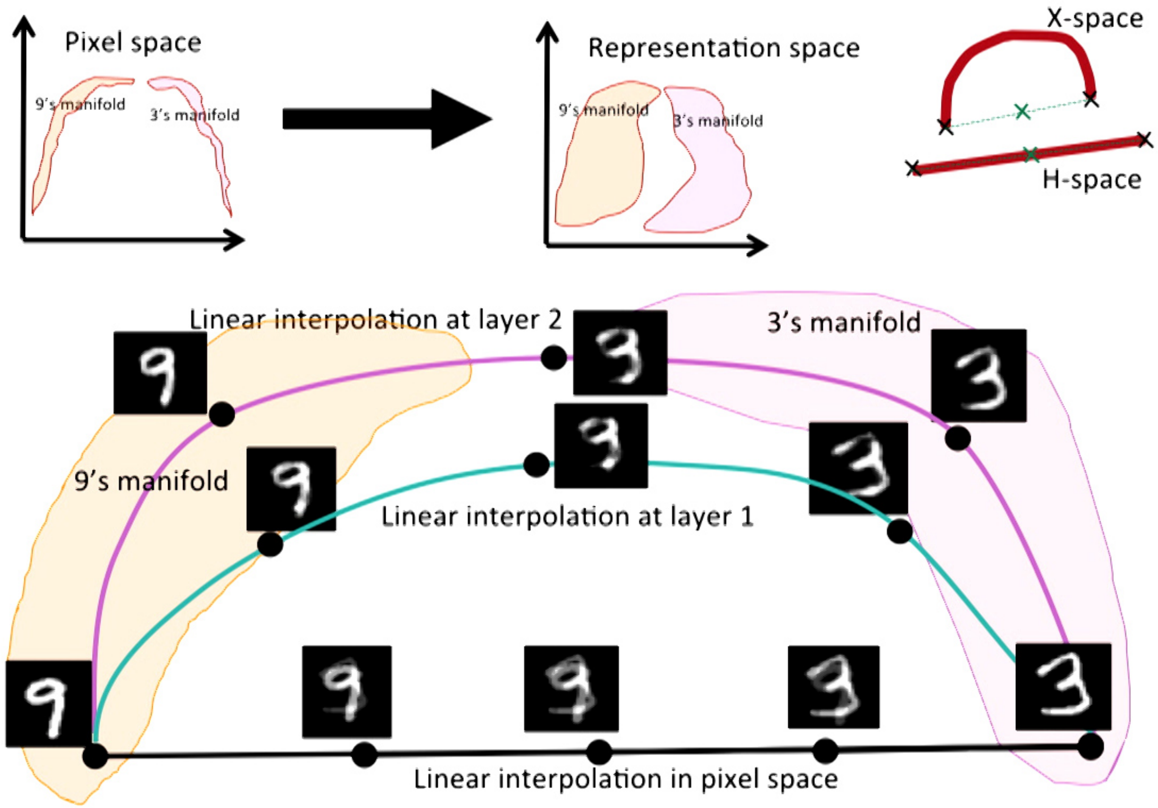
Ozair and Bengio (2014) based on Bengio et al. (2013)

What about interpolation?



See: Chawla et al. Synthetic Minority Oversampling Technique (2002)

The problem (and proposal)



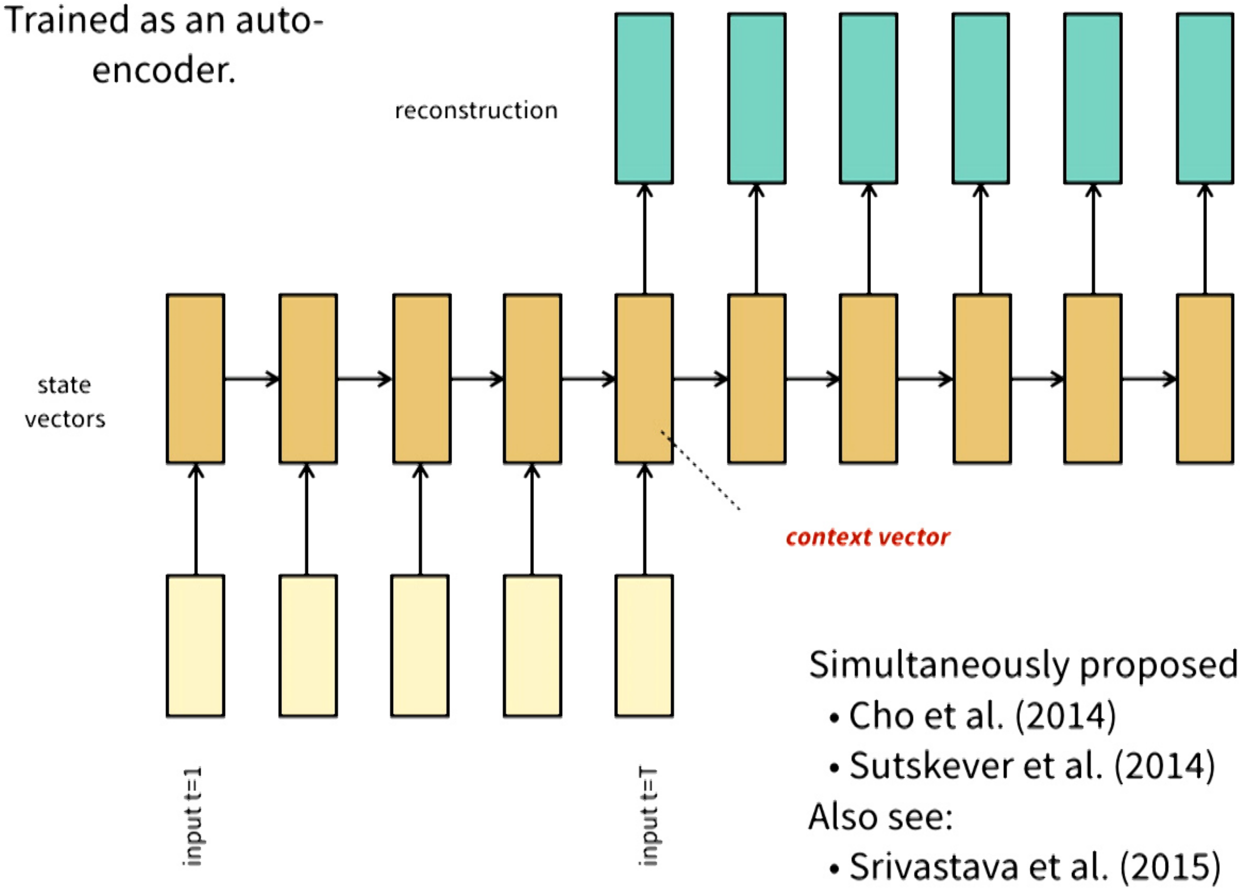
Ozair and Bengio (2014) based on Bengio et al. (2013)

Key Question

- Can we achieve a domain-agnostic type of dataset augmentation by working in feature space and applying simple transformations? e.g.
  - interpolation
  - extrapolation
  - adding noise

Proposal: seq2seq AE

Trained as an auto-encoder.

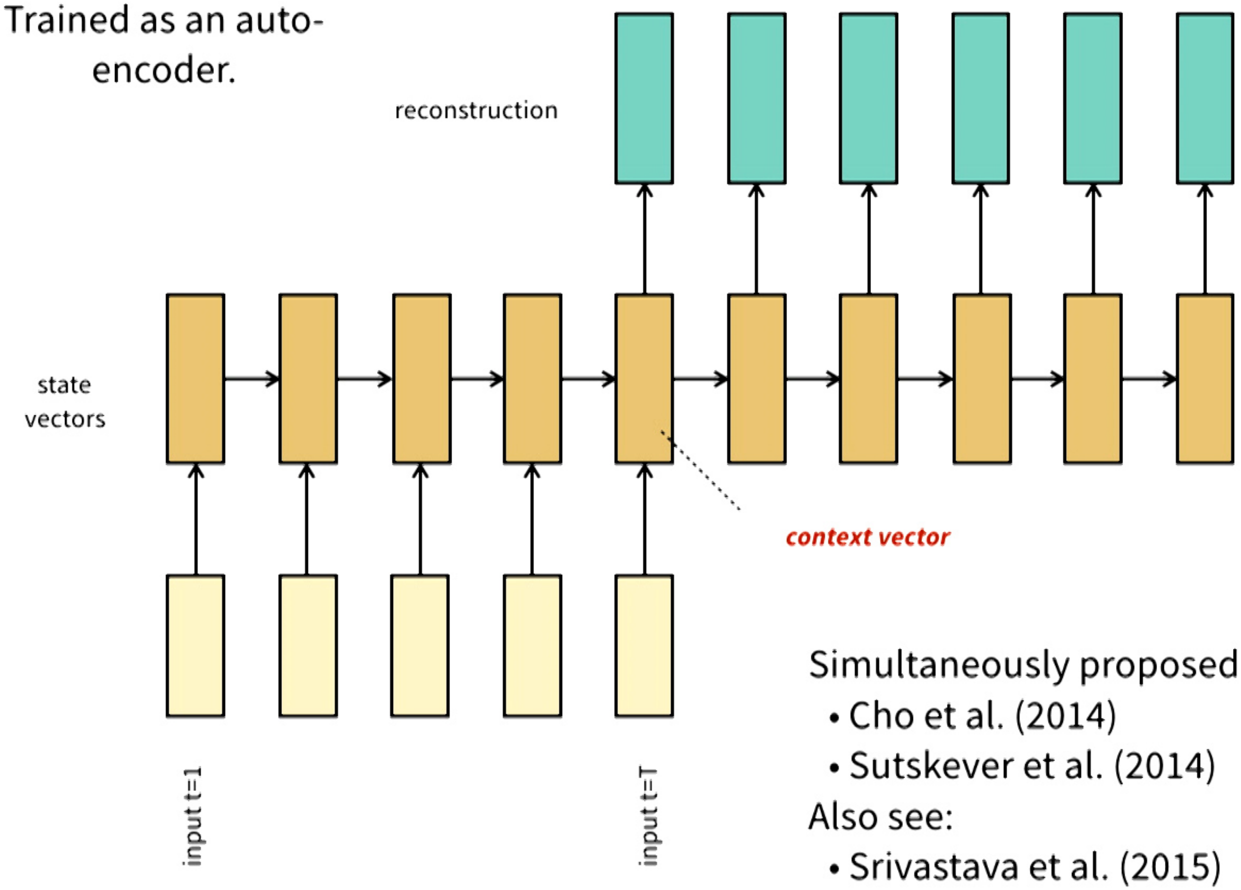


Simultaneously proposed by:
- Cho et al. (2014)
- Sutskever et al. (2014)

Also see:
- Srivastava et al. (2015)

Proposal: seq2seq AE

Trained as an auto-encoder.



Simultaneously proposed by:

- Cho et al. (2014)
- Sutskever et al. (2014)

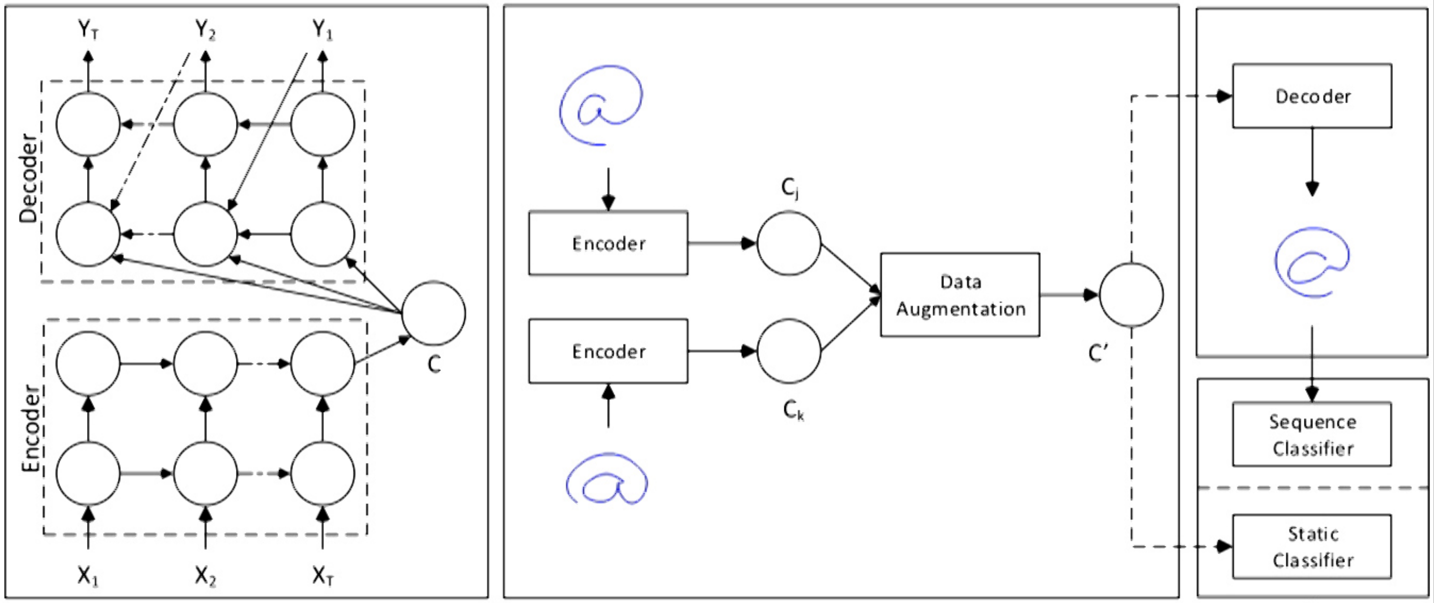
Also see:

- Srivastava et al. (2015)

Why seq2seq autoencoder?

- Main reason to adopt seq2seq autoencoder (SA): we favour a generic method that can be used for temporal or static signal data
  - Simply convert static data (e.g. image) to a stream that can be processed by a SA (e.g. spatial RNNs)

Overview of method



(1) Train seq2seq AE (unsupervised)

(2) Data is encoded to context vectors (C) and then augmented

(3) The resulting C's are either classified directly using a static classifier, or decoded to sequences for training a sequence classifier

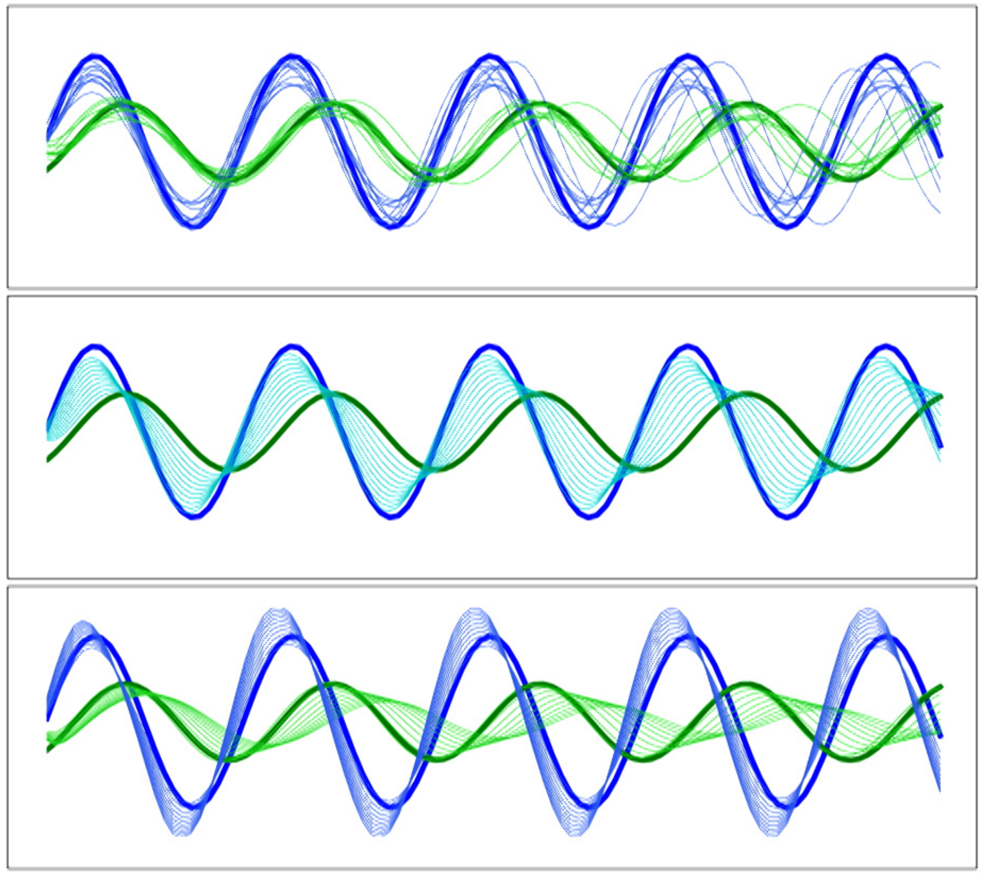
Visualization - sinusoids

Feature Augmenter

Additive Gaussian noise

Interpolation

Extrapolation



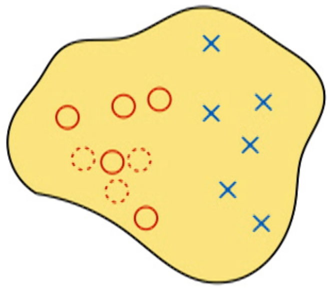
Synthesis Procedure

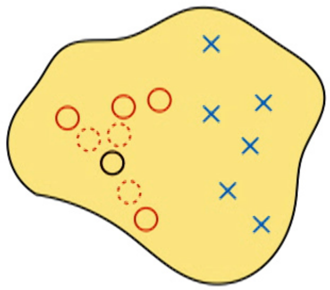
Noise

$$c_i' = c_i + \gamma n_i, \quad n_i \sim \mathcal{N}\left(0, \sigma_i^2\right)$$

Interpolation/extrapolation

- For each sample j, find K nearest neighbours in feature space which share the same label
- Then, for each neighbour k
  - Interpolation:
    $$\mathbf{c}' = \lambda\left(\mathbf{c}^{(k)} - \mathbf{c}^{(j)}\right) + \mathbf{c}^{(j)}, \quad \lambda \in (0,1)$$
  - Extrapolation:
    $$\mathbf{c}' = \lambda\left(\mathbf{c}^{(j)} - \mathbf{c}^{(k)}\right) + \mathbf{c}^{(j)}, \quad \lambda \in (0,\infty)$$





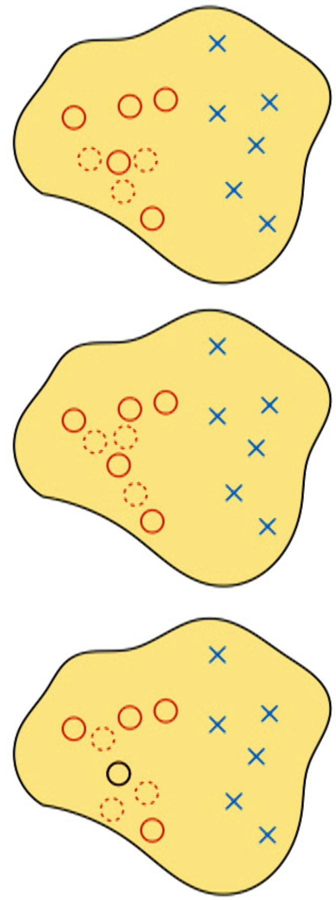
Synthesis Procedure

Noise

$$c_i' = c_i + \gamma n_i, \quad n_i \sim \mathcal{N}\left(0, \sigma_i^2\right)$$

Interpolation/extrapolation

- For each sample j, find K nearest neighbours in feature space which share the same label
- Then, for each neighbour k
  - Interpolation:
    $$\mathbf{c}' = \lambda\left(\mathbf{c}^{(k)} - \mathbf{c}^{(j)}\right) + \mathbf{c}^{(j)}, \quad \lambda \in (0,1)$$
  - Extrapolation:
    $$\mathbf{c}' = \lambda\left(\mathbf{c}^{(j)} - \mathbf{c}^{(k)}\right) + \mathbf{c}^{(j)}, \quad \lambda \in (0,\infty)$$



Quantitative results across modalities

Arabic Spoken Digits (Speech) N_c=10

Method	Test Error (%)
Baseline: MLP 256-256 + Dropout	1.36 ± 0.15
Baseline + random noise (ours)	1.10 ± 0.15
Baseline + NN interpolation (ours)	1.57 ± 0.19
Baseline + NN extrapolation (ours)	**0.74 ± 0.11**
GMM (Hammami et al., 2012)	**0.69**

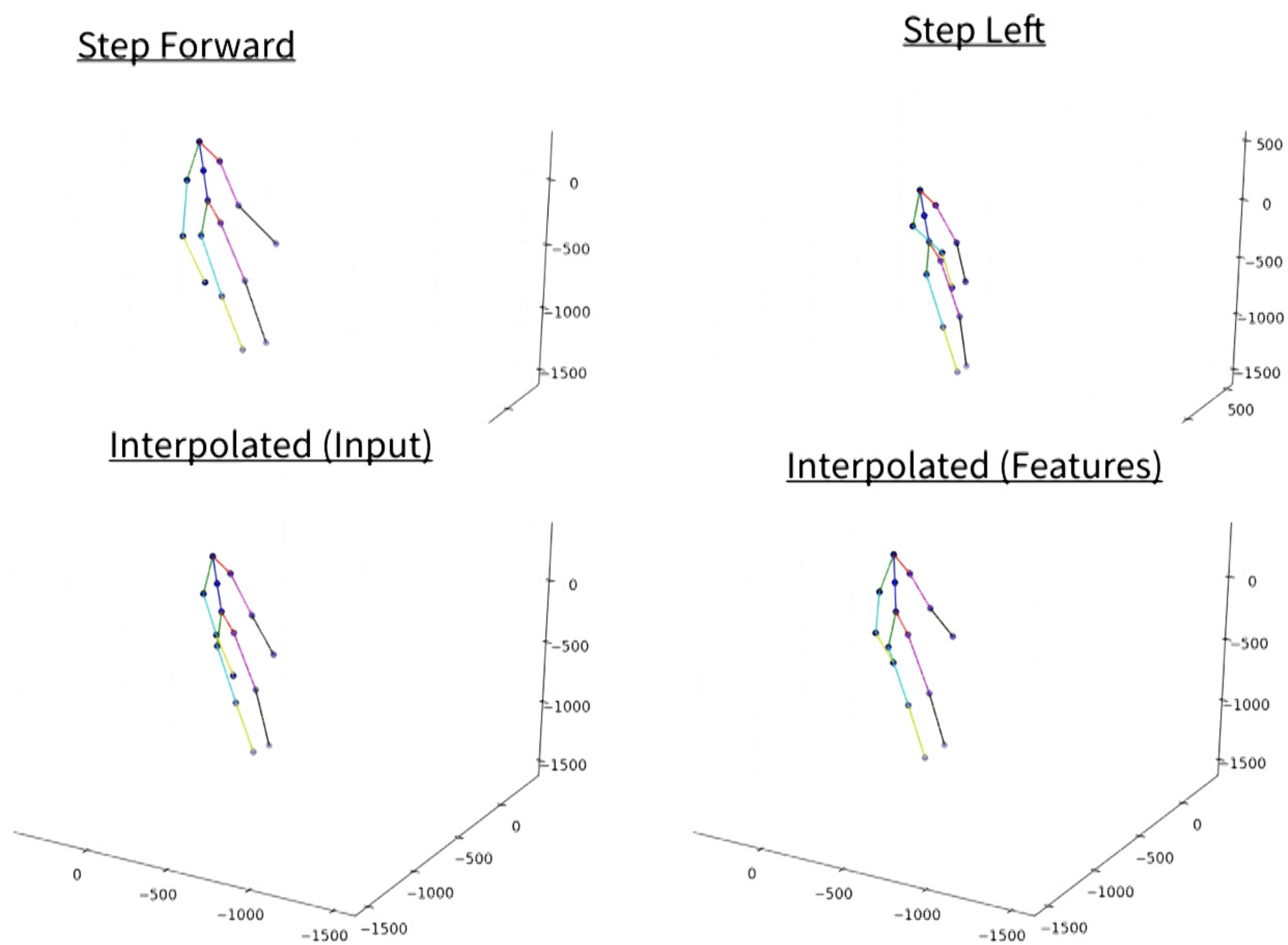
AUSLAN (Hand Mocap) N_c=95

Method	Test Error (%)
Baseline: MLP 512-512 + Dropout	1.53 ± 0.26
Baseline + random noise (ours)	1.67 ± 0.12
Baseline + NN interpolation (ours)	1.87 ± 0.44
Baseline + NN extrapolation (ours)	**1.21 ± 0.26**
SVM* (Rodriguez et al., 2005)	**1.28**

UCFKinect (Full Body Mocap) N_c=16

Method	Test Error (%)
Baseline: MLP 512-512 + Dropout	4.92 ± 0.11
Baseline + NN extrapolation (ours)	3.59 ± 0.08
HMM (Beh et al., 2014)	**1.10**

Qualitative results - mocap



Experiments: static image data

- Always use the same SA architecture to learn representation
- CIFAR-10 we flip and crop to 24 x 24
- Images are fed to the network one row of pixels at a time (Dai & Le 2016)

Method	MNIST	CIFAR
Baseline MLP 256-256/1024-1024	1.093 ± 0.057	30.65 ± 0.27
Baseline + input affine transformations	1.477 ± 0.068	
Baseline + input NN extrapolation	1.010 ± 0.065	
Baseline + feature NN extrapolation	**0.950 ± 0.036**	**29.24 ± 0.27**

Experiments: Convnets

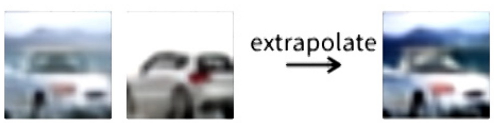
Architecture: Wide ResNet (Zagoruyko & Komodakis 2016)

Test #	Image size	Method	Test Error (%)
1	32×32	Original dataset	8.79
2	24×24	Center crop	11.21
3	24×24	Center crop + extrapolation	14.11
4	24×24	Center crop + shifts and mirrors	**7.33**
5	24×24	CC + shifts and mirrors + extrapolation	8.55

Experiments: Convnets

Architecture: Wide ResNet (Zagoruyko & Komodakis 2016)

Test #	Image size	Method	Test Error (%)	Test Error (%) Using reconstructions of original data
1	32×32	Original dataset	8.79	
2	24×24	Center crop	11.21	18.75
3	24×24	Center crop + extrapolation	14.11	17.72
4	24×24	Center crop + shifts and mirrors	**7.33**	13.55
5	24×24	CC + shifts and mirrors + extrapolation	8.55	**11.99**



Summary

- **Domain-independent** and **inexpensive** data augmentation
  - For supervised learning applications where large, labeled datasets are unavailable and supervised pre-training fails
- Sequence autoencoders + basic transformations
  - Suited to temporal and static data classification
  - Works in four different domains *using the same simple architecture*
- Compliments domain-specific augmentation (CIFAR-10)
- Another way of evaluating unsupervised learning algorithms?

Future Work

- Experiments with new domains (such as NLP) and larger datasets
- End-to-end learning
- Synthesizing from > 2 examples
- Learning an optimal transformation operator

Acknowledgements

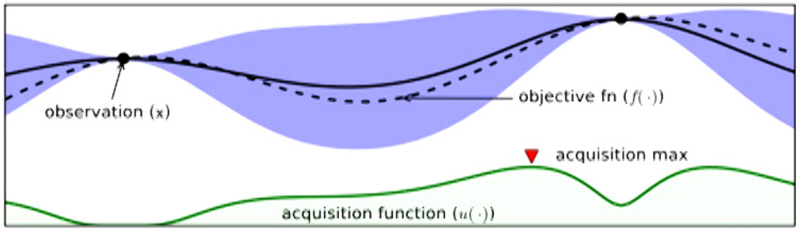
Terrance DeVries
University of Guelph

NSERC
CRSNG

INNOVATION.CA
CANADA FOUNDATION FOR INNOVATION | FONDATION CANADIENNE POUR L'INNOVATION

Bayesian Optimization

t=2



10 Feb 2016 / 8
DARPA QPR - Model Search/ G Taylor

Image Credit: Brochu et al. (2010)

Bayesian Optimization

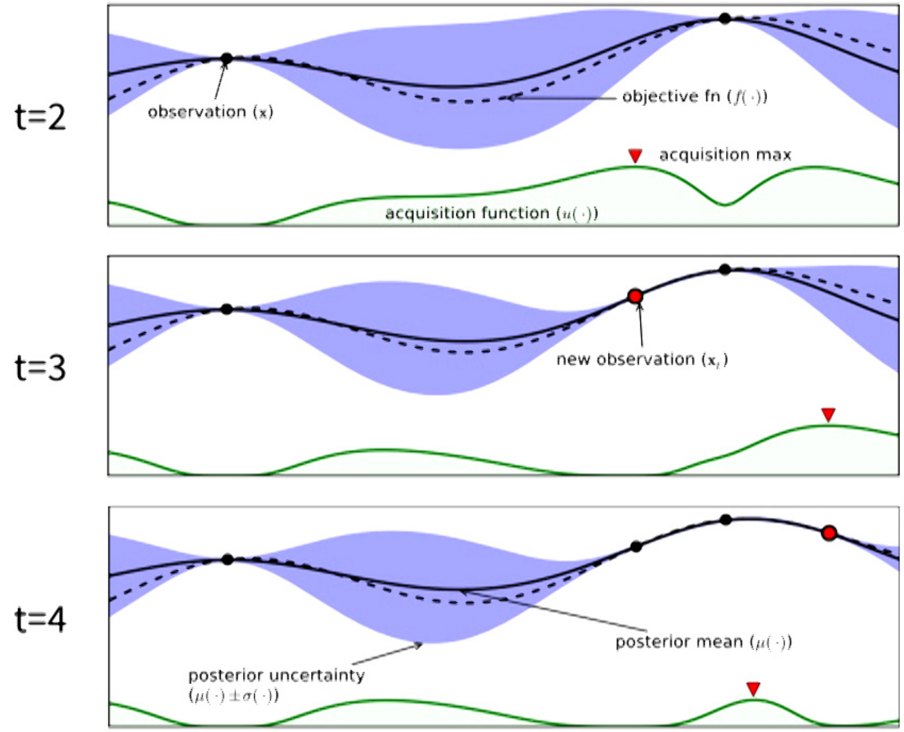


Image Credit: Brochu et al. (2010)