LEARNING TO COMPARE

GRAHAM TAYLOR

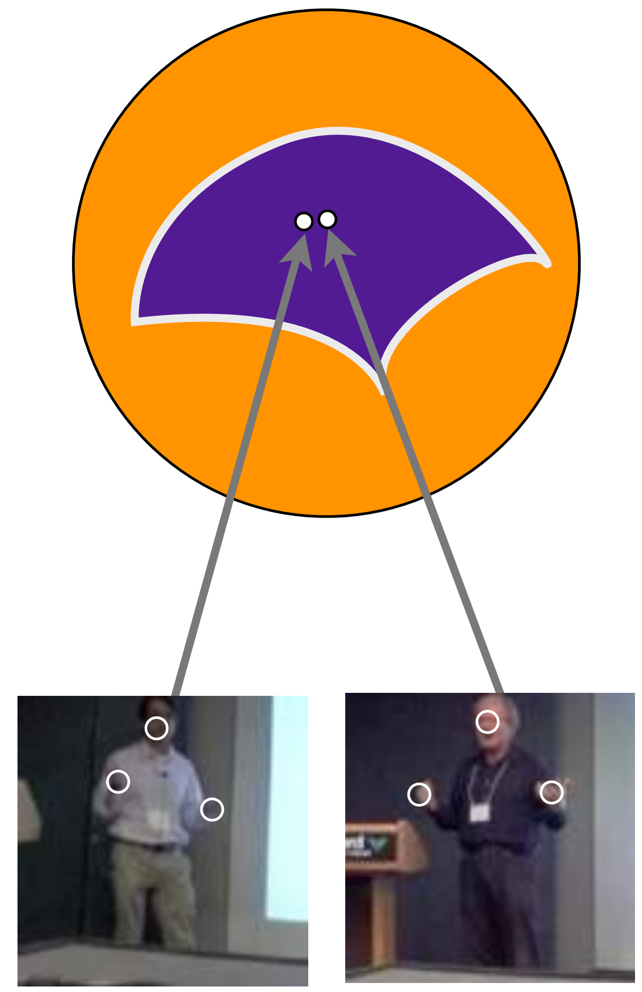
SCHOOL OF ENGINEERING
UNIVERSITY OF GUELPH

Deep Learning Summer School 2015
Montreal, Quebec

Overview: this talk

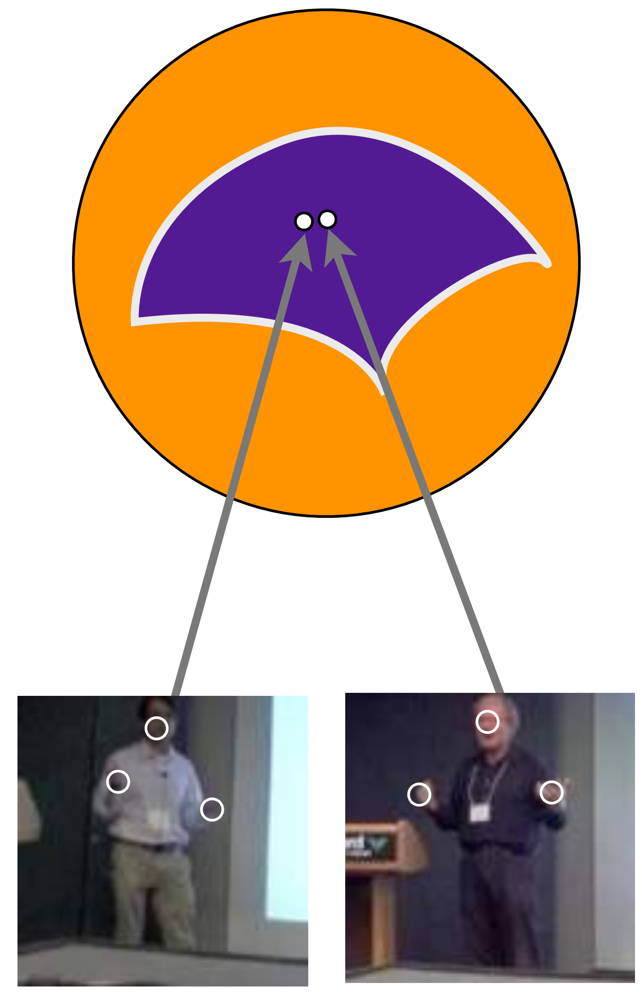
Overview: this talk

- Learning to compare examples
  - it's a big field!
  - we will focus on methods inspired by deep learning and representation learning



Overview: this talk

- Learning to compare examples
  - it's a big field!
  - we will focus on methods inspired by
    deep learning
    and representation learning



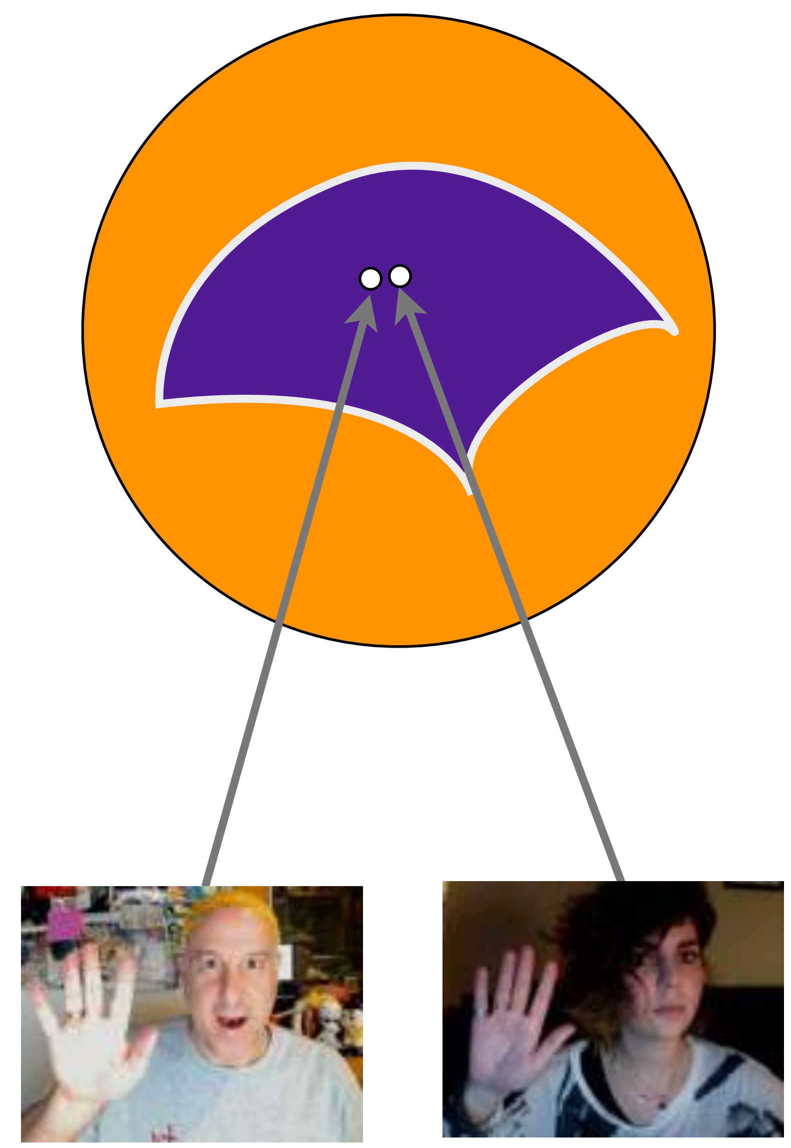
- Applications: finding similar documents, pose-sensitive retrieval, zero-shot learning

Learning similarity

- Pixel distance ≠ perceptual similarity
- Computing distances in pixel space is also computationally expensive
- Learning parametric embeddings that are *invariant* to certain input variability

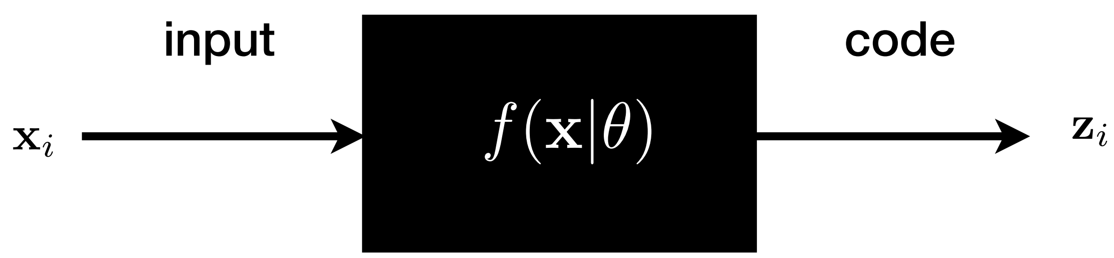
The setup

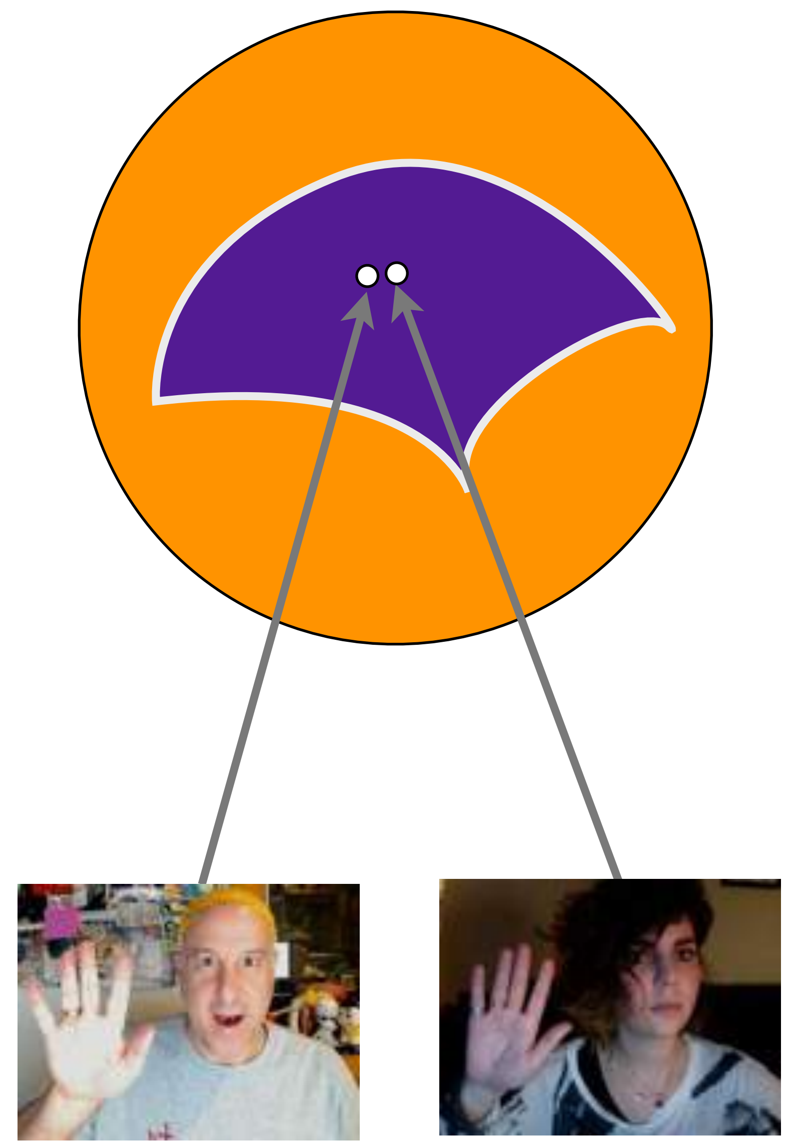
- Perceptually similar observations are mapped to nearby points on a manifold
- Key question: where does similarity come from?



The setup

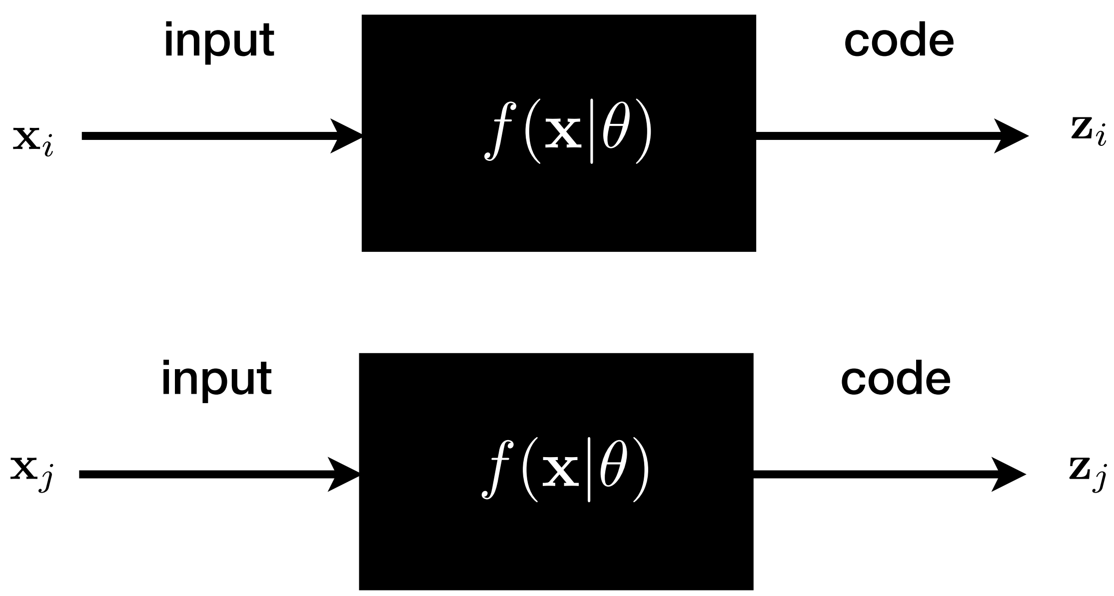
- Perceptually similar observations are mapped to nearby points on a manifold
- Key question: where does similarity come from?

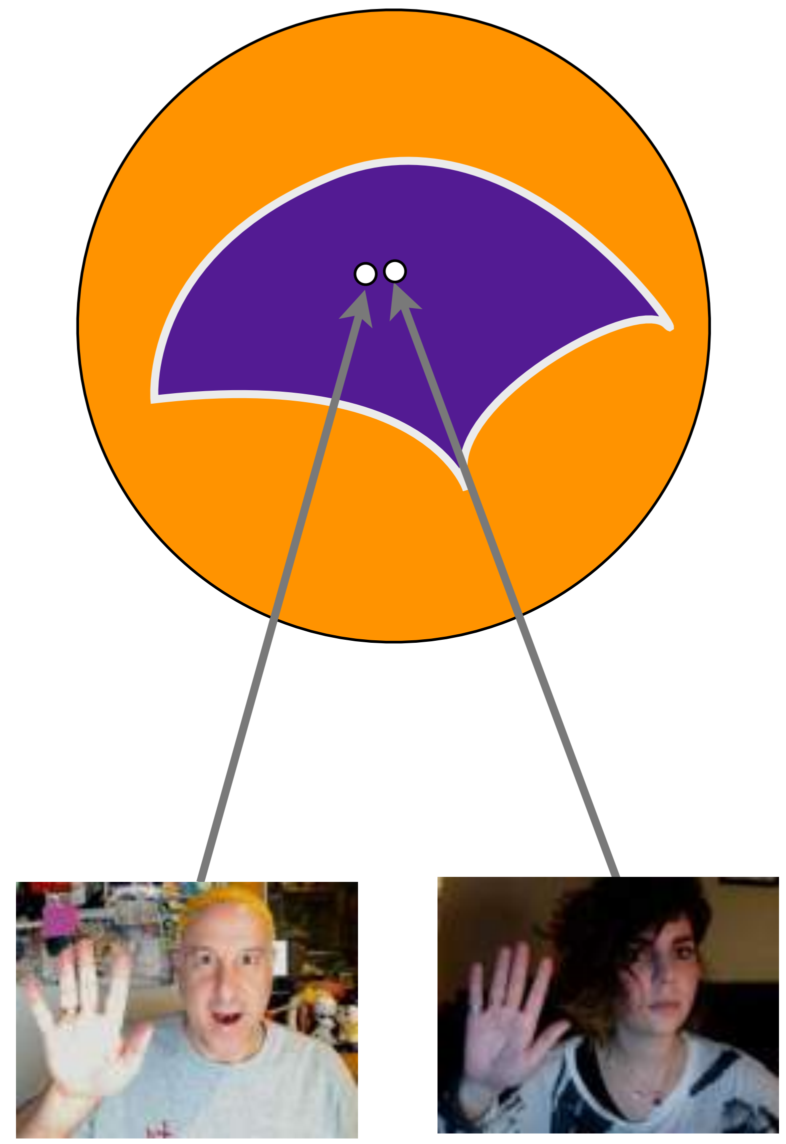




The setup

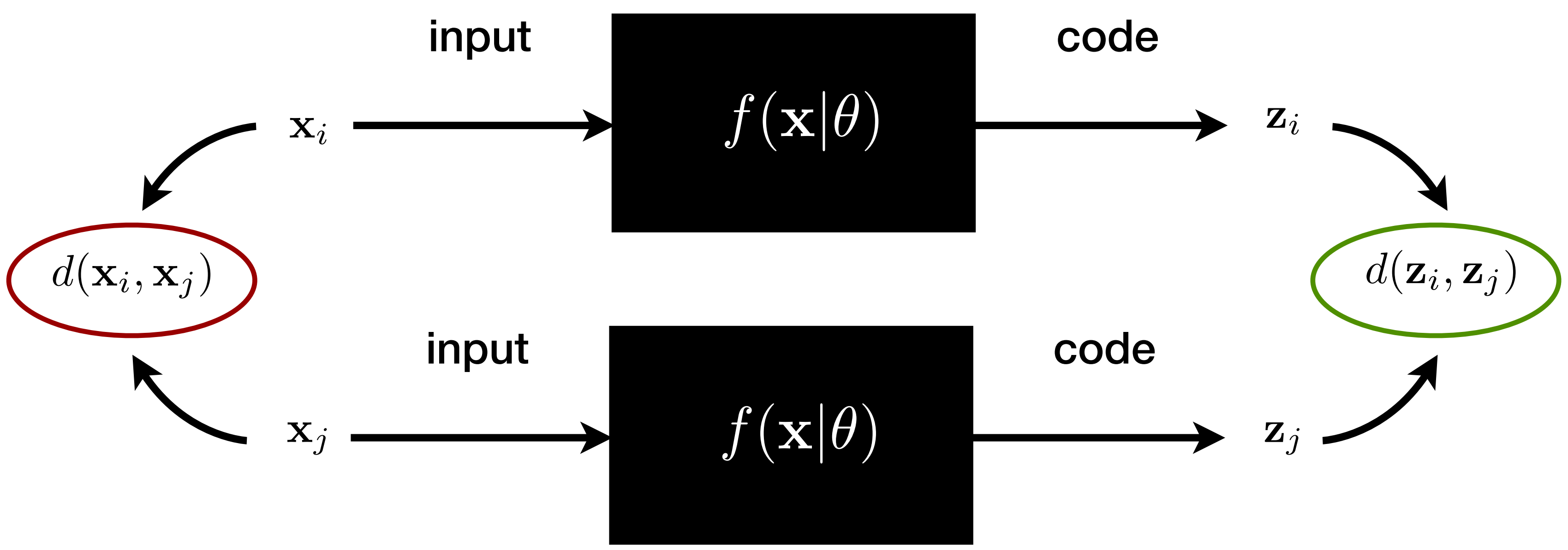
- Perceptually similar observations are mapped to nearby points on a manifold
- Key question: where does similarity come from?

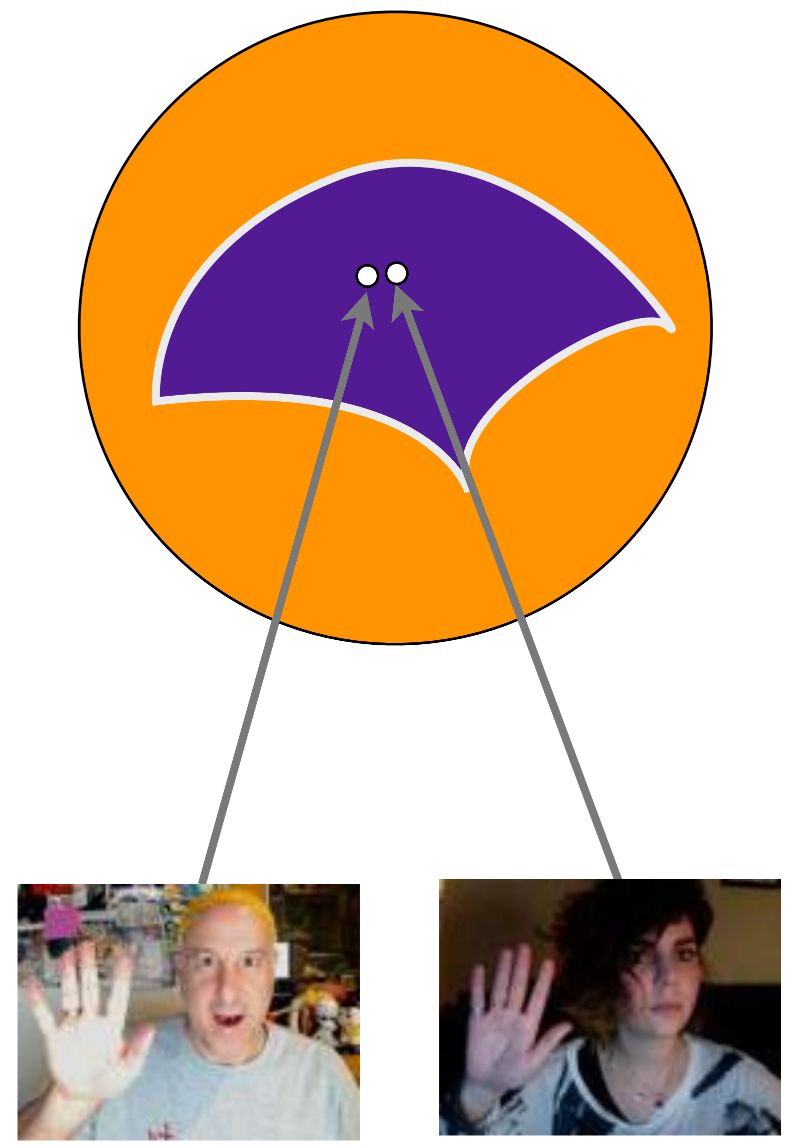




The setup

- Perceptually similar observations are mapped to nearby points on a manifold
- Key question: where does similarity come from?





One motivation: nearest neighbour methods

- Surprisingly effective (Boiman et al. 2008, McCann and Lowe, 2012)
- Fast, especially when combined with Approximate Nearest Neighbour or Hashing
- Generalize to new classes at near-zero cost (Mensink et al. 2013)

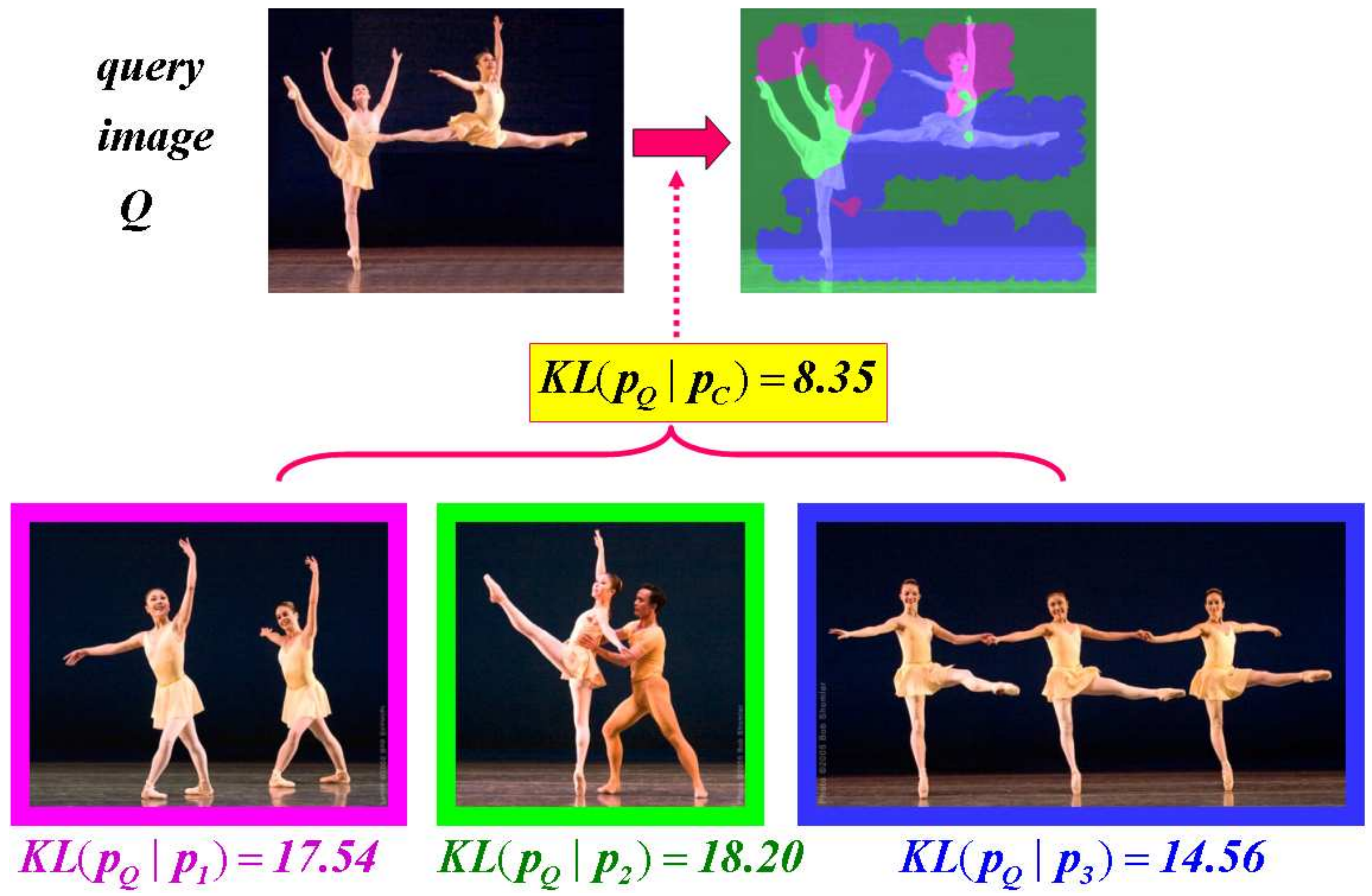


Image: Boiman et al. (2008)

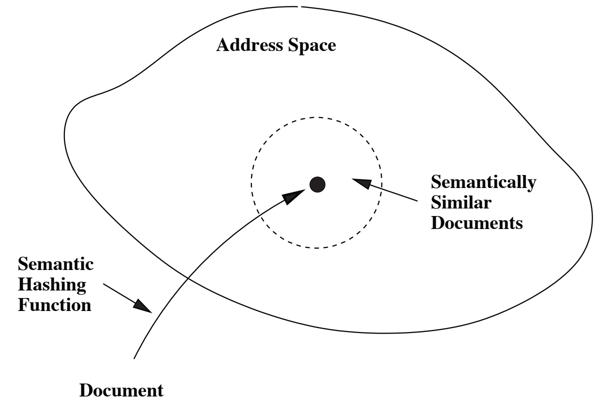
Credit: Google Server Farm in Council Bluffs, Iowa (Wired)

Outline

Outline

Unsupervised

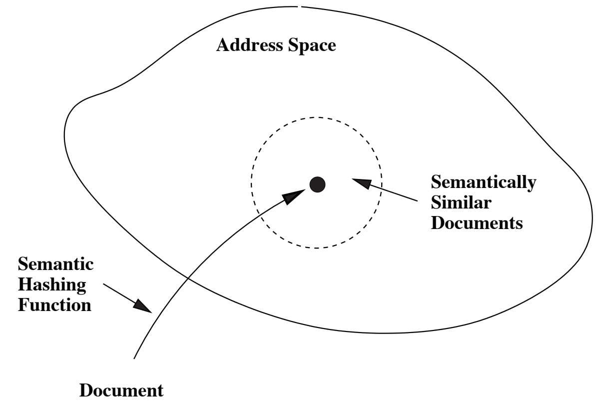
LSA, Semantic Hashing, Multi-index Hashing



Outline

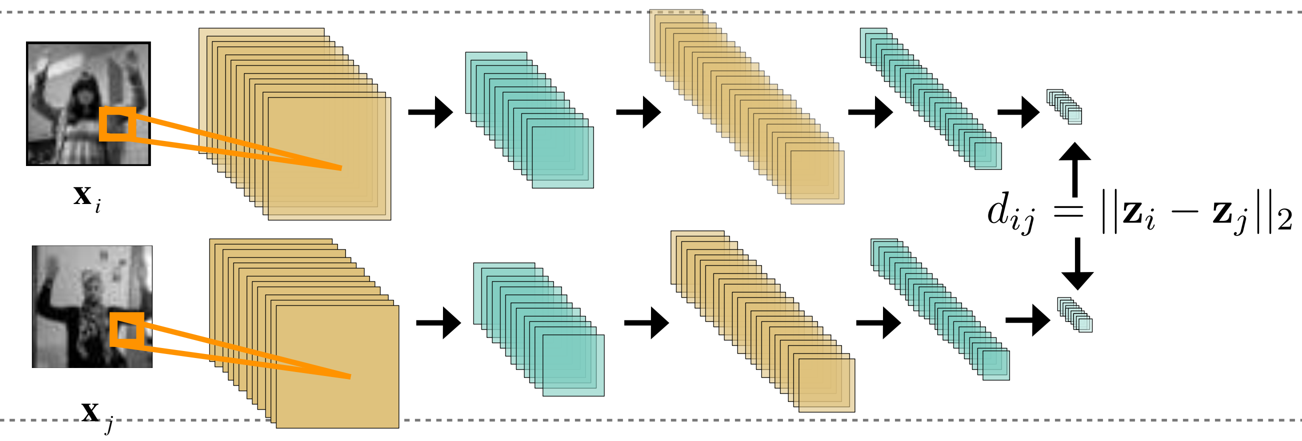
Unsupervised

LSA, Semantic Hashing, Multi-index Hashing



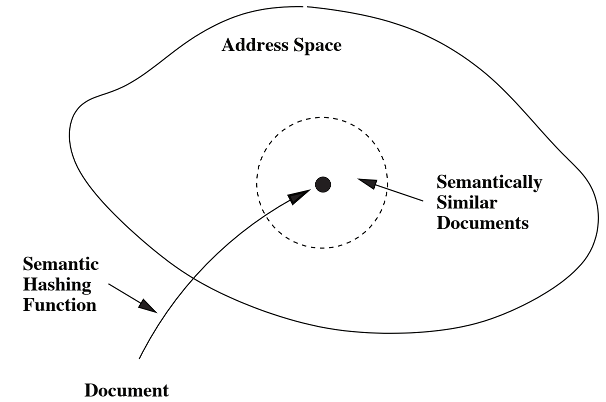
Supervised

NCA, Nonlinear NCA, DrLIM, Triplet Embedding

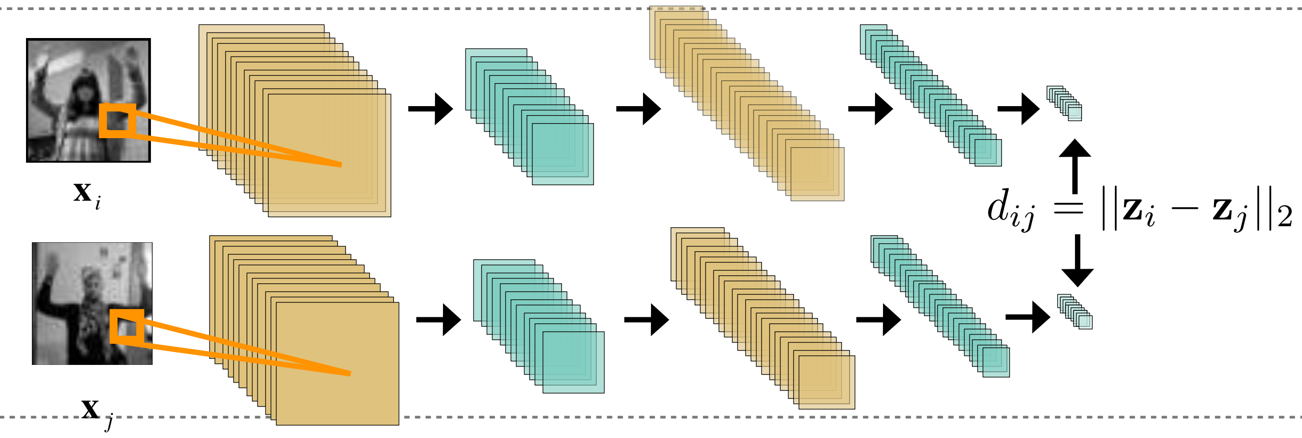


Outline

Unsupervised
LSA, Semantic Hashing, Multi-index Hashing



Supervised
NCA, Nonlinear NCA, DrLIM, Triplet Embedding



Weakly supervised
Applications to pose-sensitive retrieval, zero-shot learning

Unsupervised approach

Unsupervised approach

- Learn (possibly deep) representations completely unsupervised
  - compute distances between top-level representations
  - representations are usually low-dimensional

Unsupervised approach

- Learn (possibly deep) representations completely unsupervised
  - compute distances between top-level representations
  - representations are usually low-dimensional
- Classical methods: Latent Semantic Analysis (based on SVD), pLSA, LDA
  - But directed models don't seem like a natural fit
  - fast inference is important for information retrieval

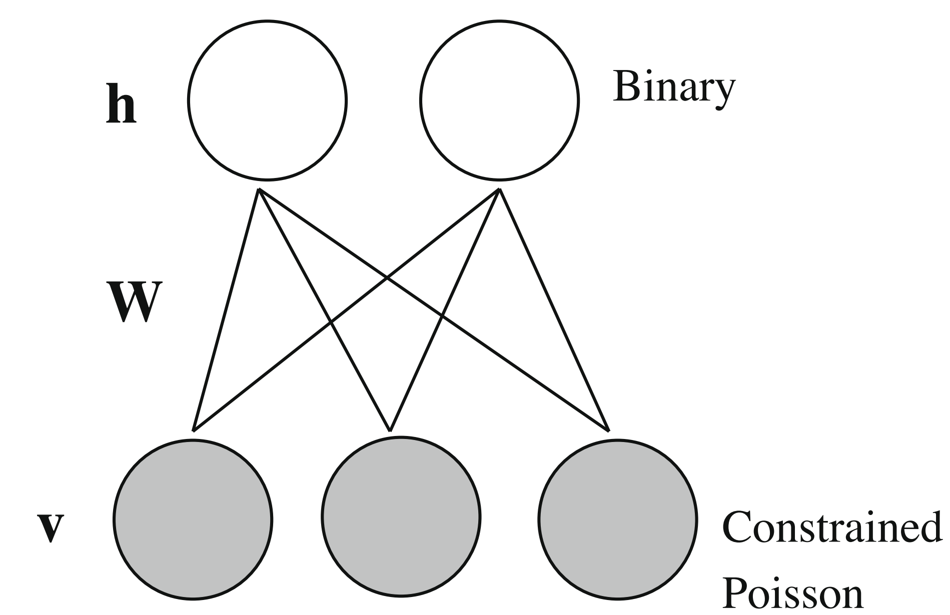
Unsupervised approach

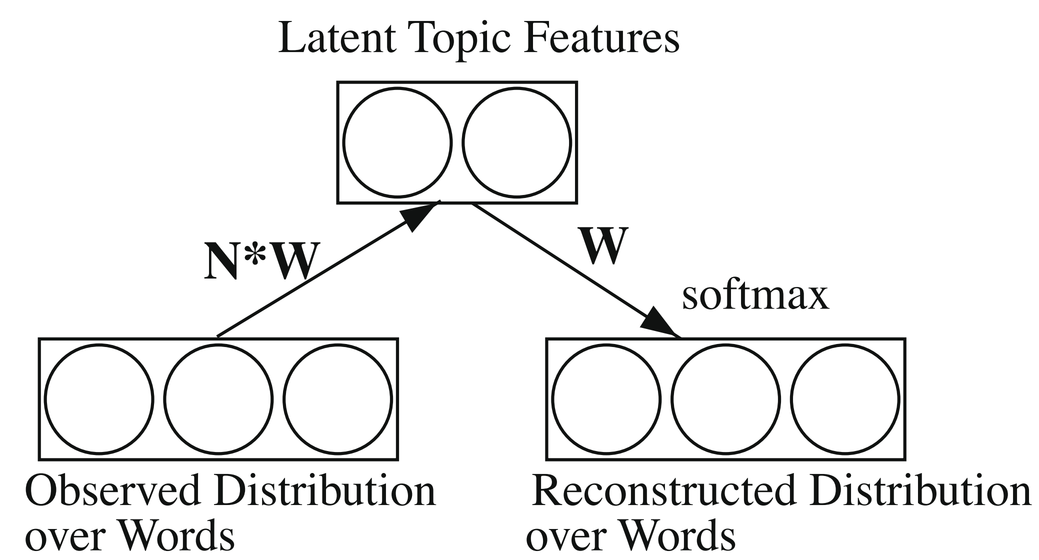
- Learn (possibly deep) representations completely unsupervised
  - compute distances between top-level representations
  - representations are usually low-dimensional
- Classical methods: Latent Semantic Analysis (based on SVD), pLSA, LDA
  - But directed models don’t seem like a natural fit
  - fast inference is important for information retrieval
- Use undirected models in which exact inference is fast
  - Single layer approach by generalizing RBMs: Welling et al. 2005
  - Multi-layer approach: Salakhutdinov and Hinton 2007 “Semantic Hashing”

Constrained Poisson model

- Visible layer represents word-count vector of a document
  - special RBM: "Constrained Poisson Model"
- Learn Constrained Poisson ➛ Binary first layer
- This allows you to represent each document with a binary representation
- Forms the first layer of a deep model

Restricted Boltzmann Machine (RBM)

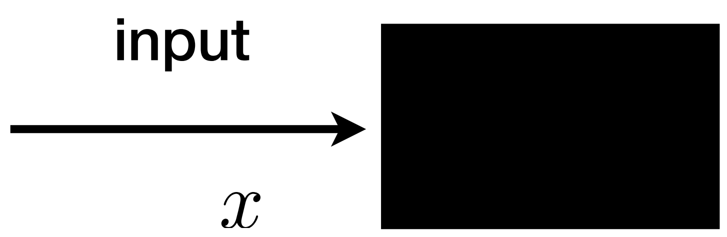




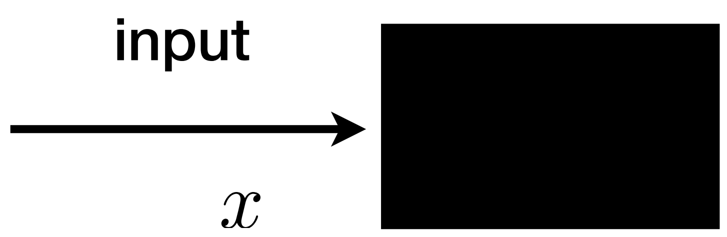
(Figures from R. Salakhutdinov and G. Hinton)

Deep auto-encoders

Deep auto-encoders

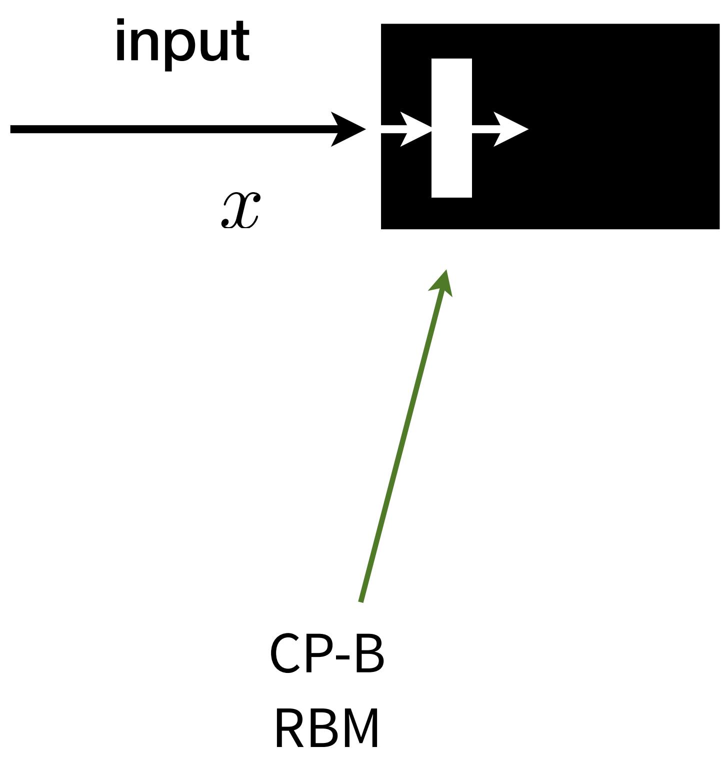


Deep auto-encoders



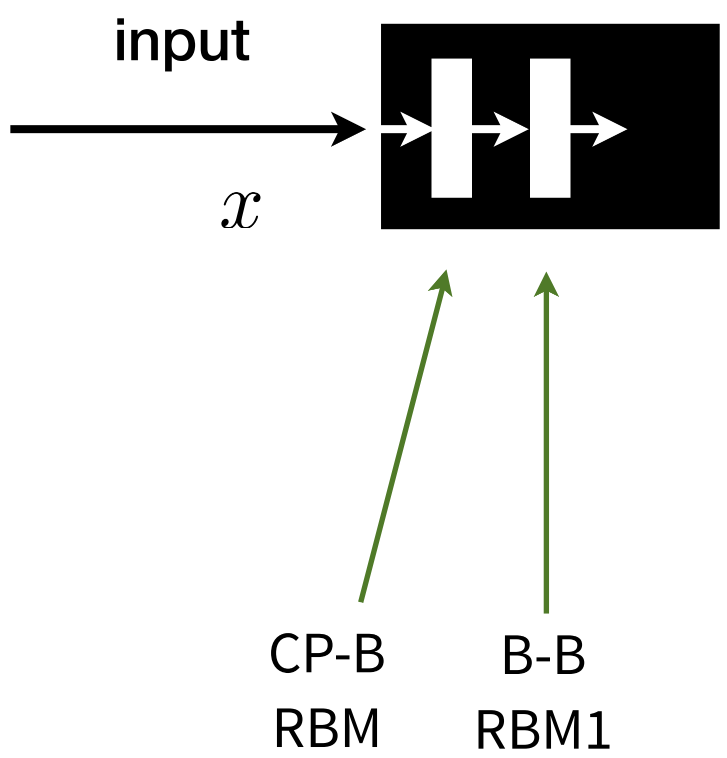
- Learn one or more binary RBMs in a “greedy” fashion

Deep auto-encoders



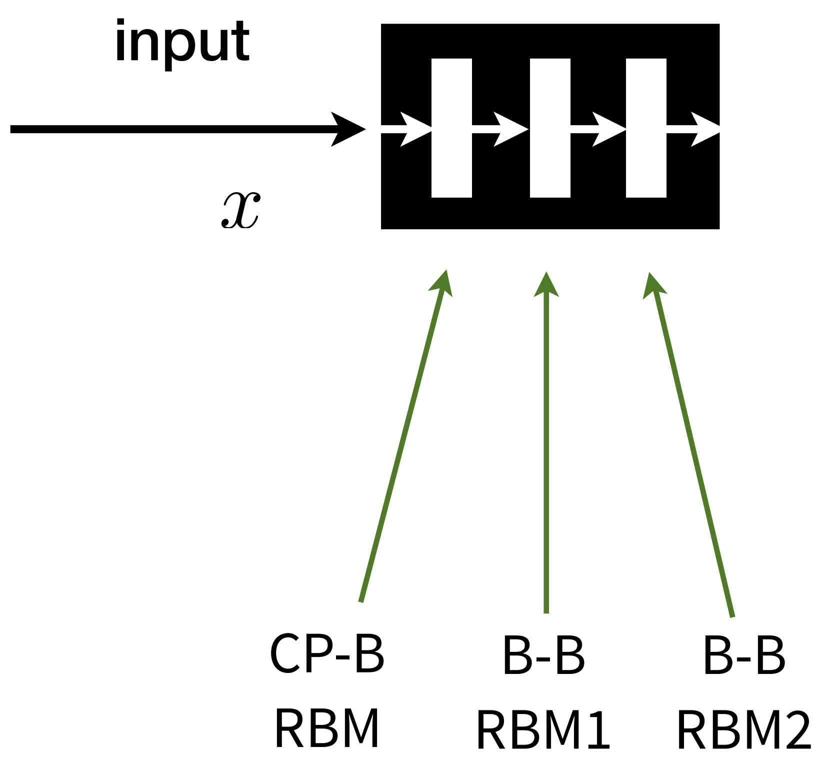
- Learn one or more binary RBMs in a "greedy" fashion

Deep auto-encoders



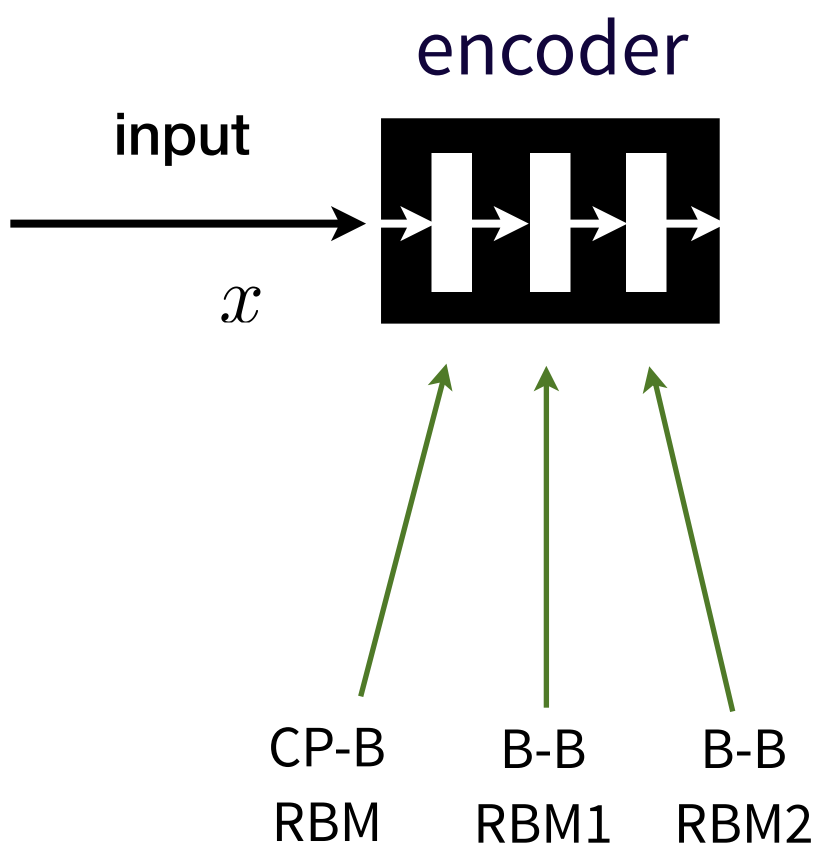
- Learn one or more binary RBMs in a “greedy” fashion

Deep auto-encoders



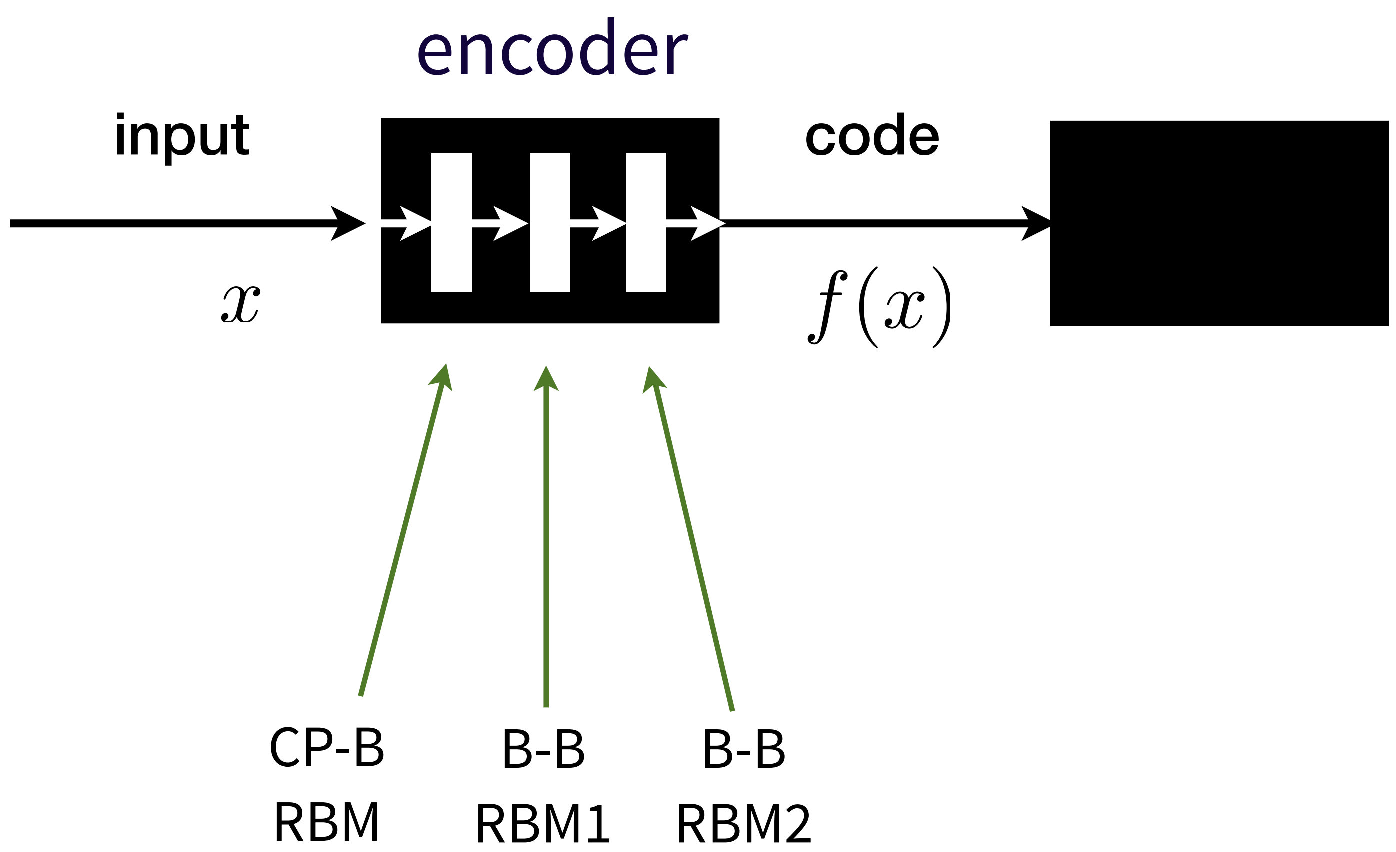
- Learn one or more binary RBMs in a “greedy” fashion

Deep auto-encoders



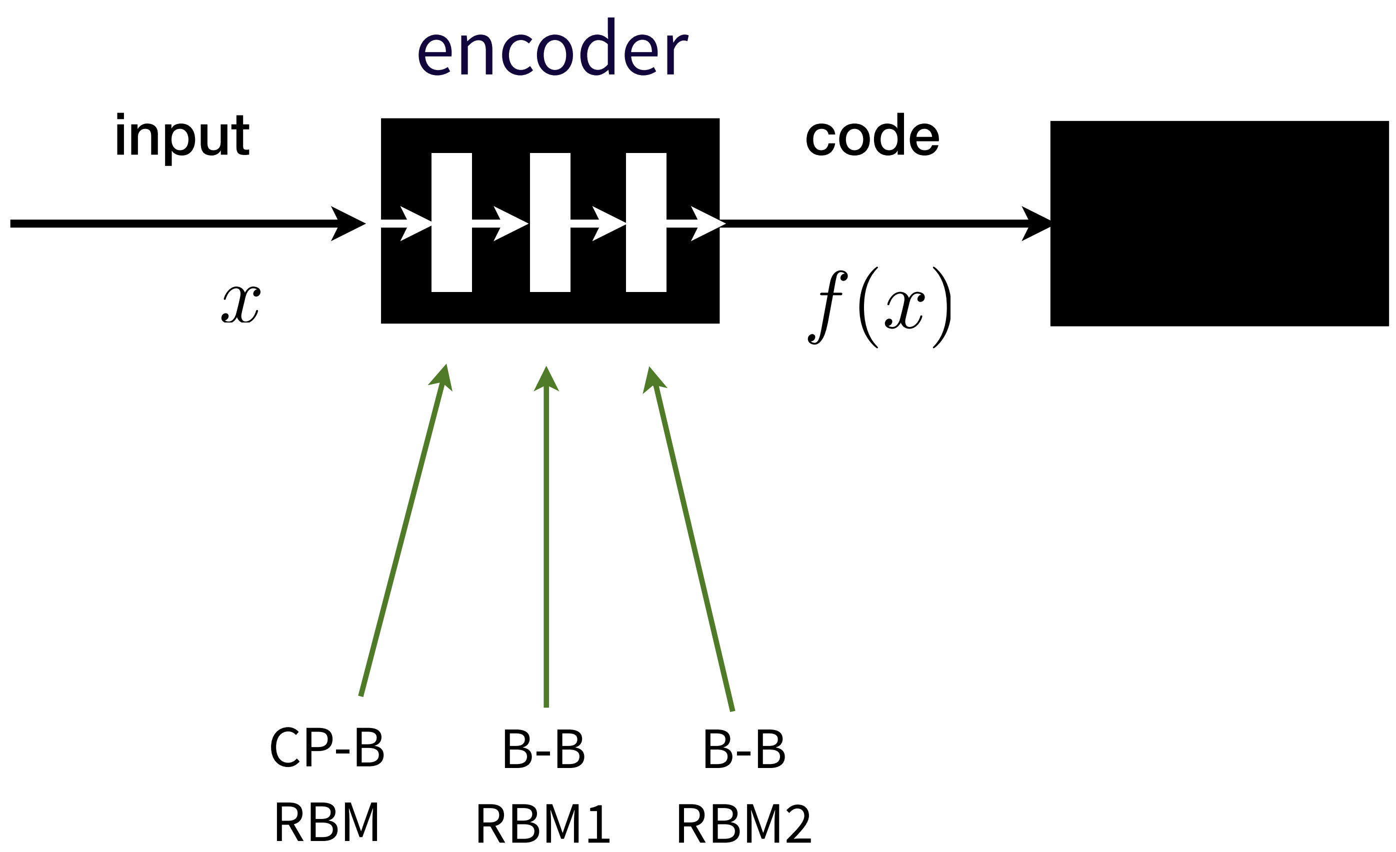
- Learn one or more binary RBMs in a “greedy” fashion

Deep auto-encoders



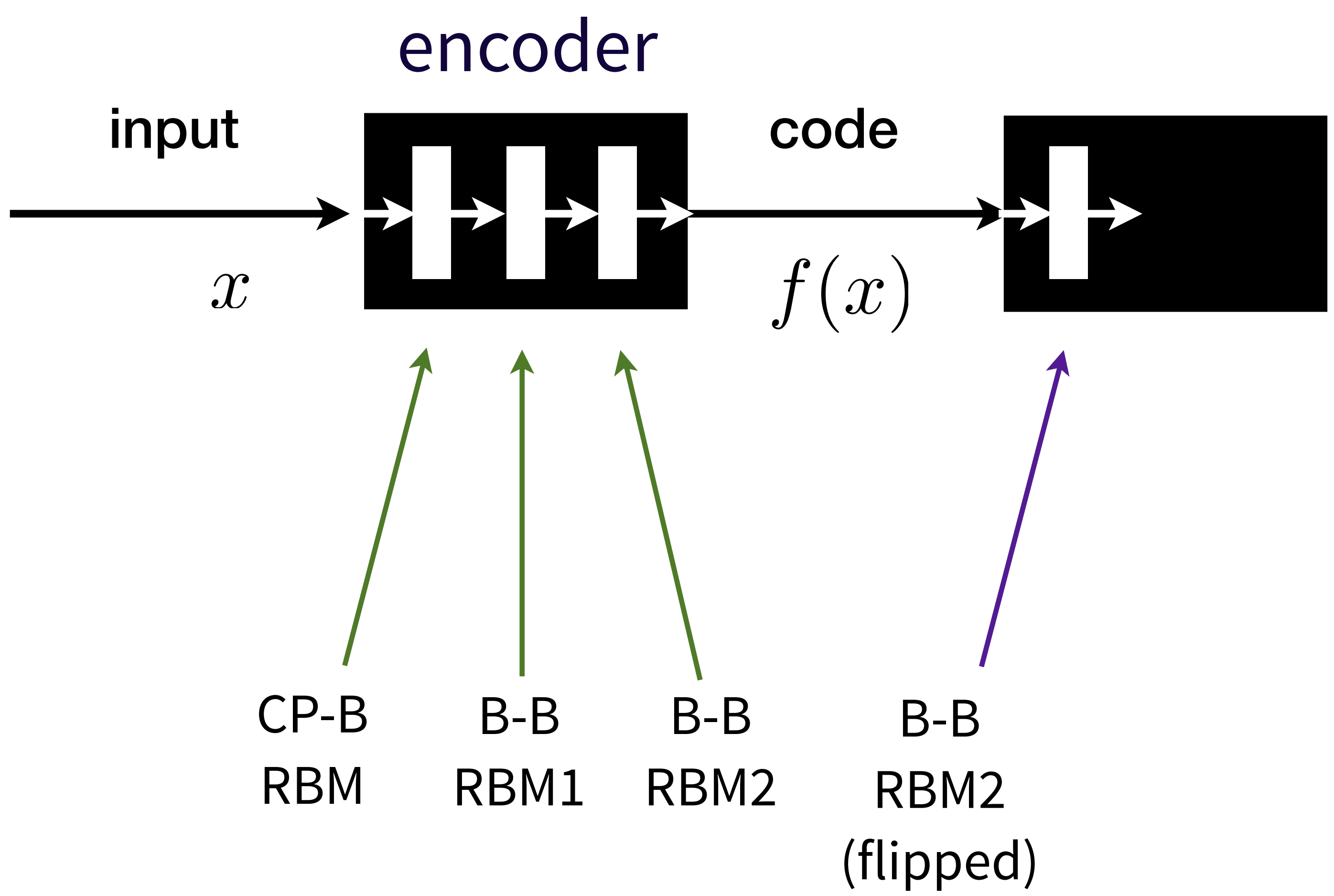
- Learn one or more binary RBMs in a “greedy” fashion

Deep auto-encoders



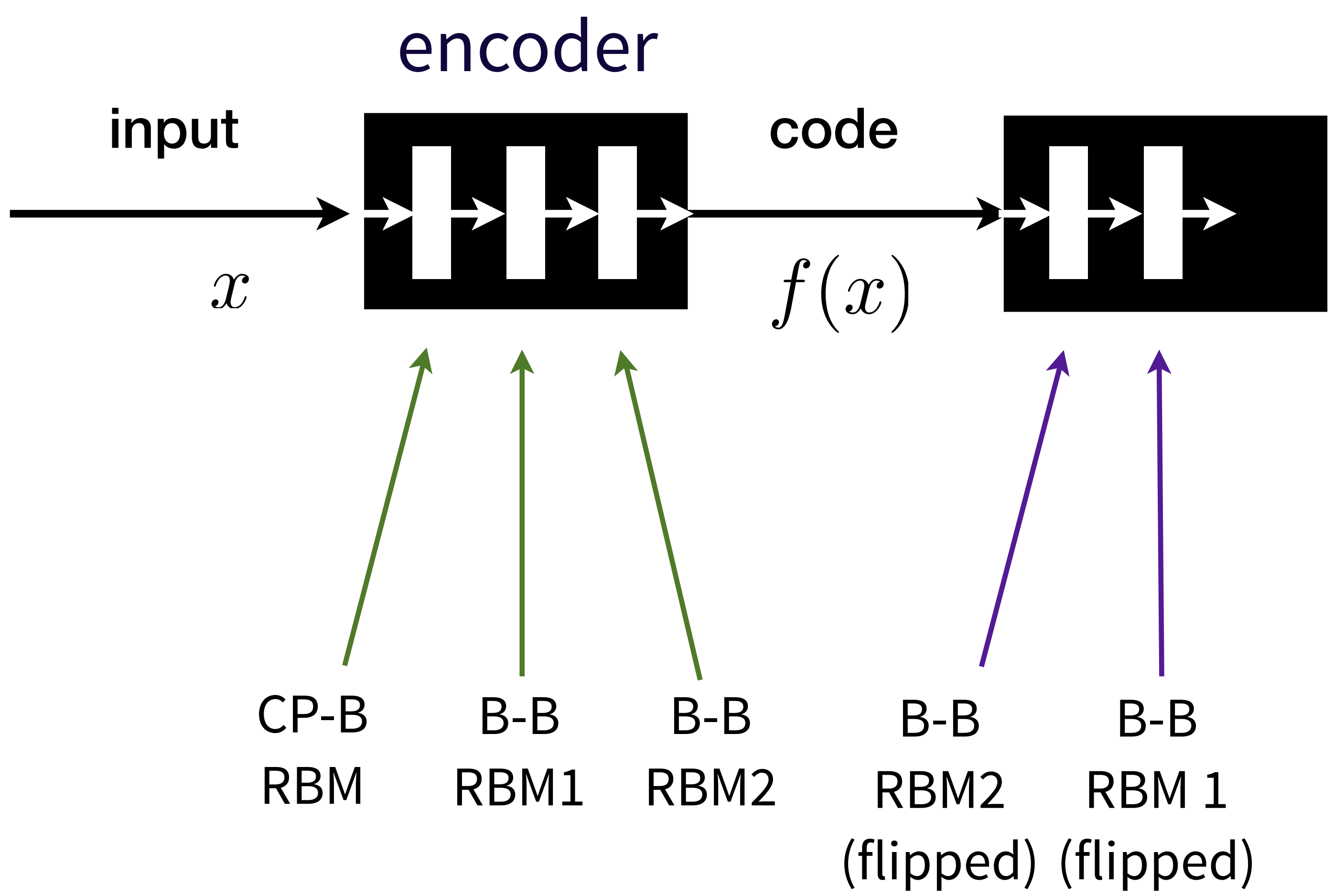
- Learn one or more binary RBMs in a “greedy” fashion
- Unroll to a deep autoencoder and “fine-tune” w/ backprop

Deep auto-encoders



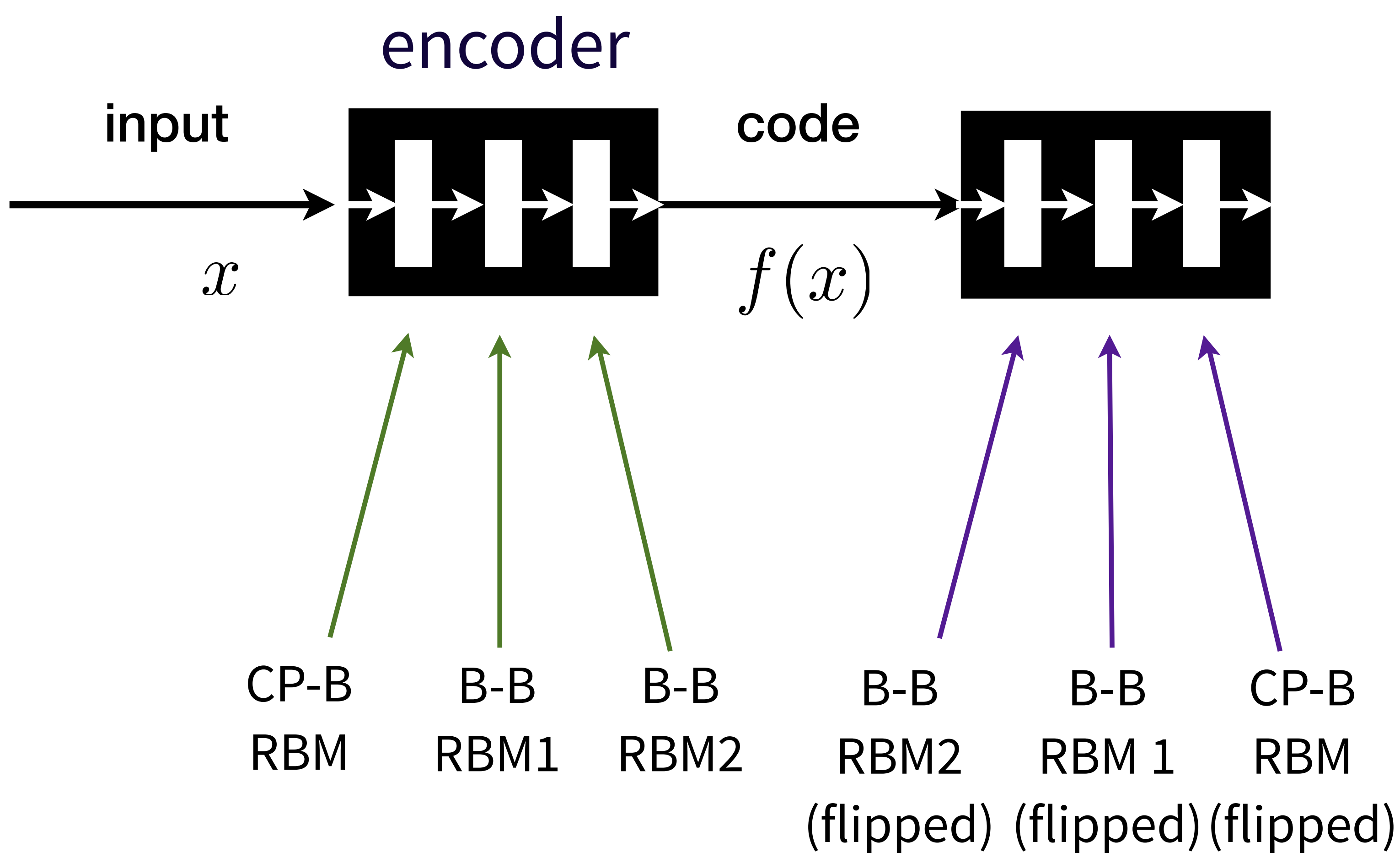
- Learn one or more binary RBMs in a “greedy” fashion
- Unroll to a deep autoencoder and “fine-tune” w/ backprop

Deep auto-encoders



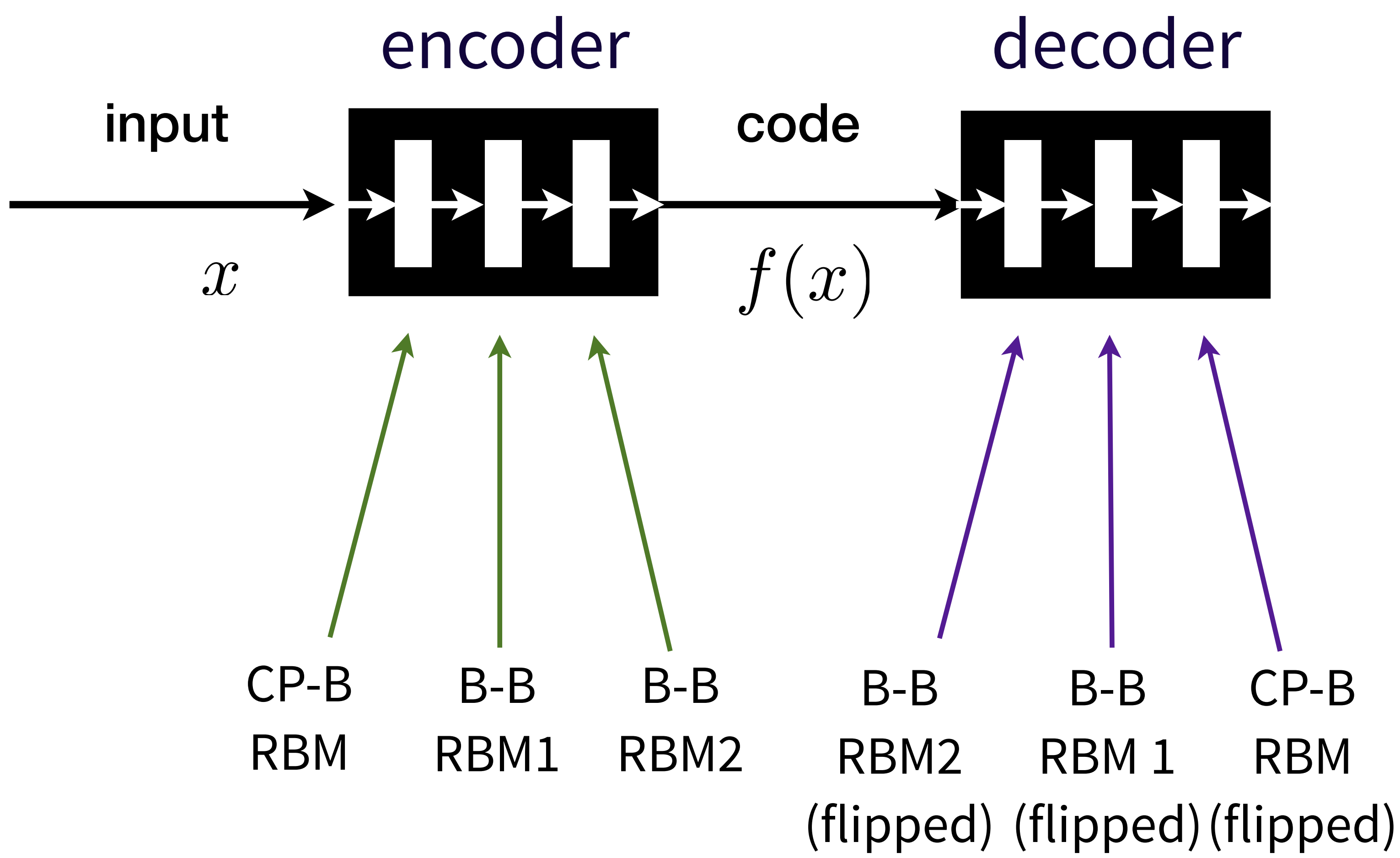
- Learn one or more binary RBMs in a “greedy” fashion
- Unroll to a deep autoencoder and “fine-tune” w/ backprop

Deep auto-encoders



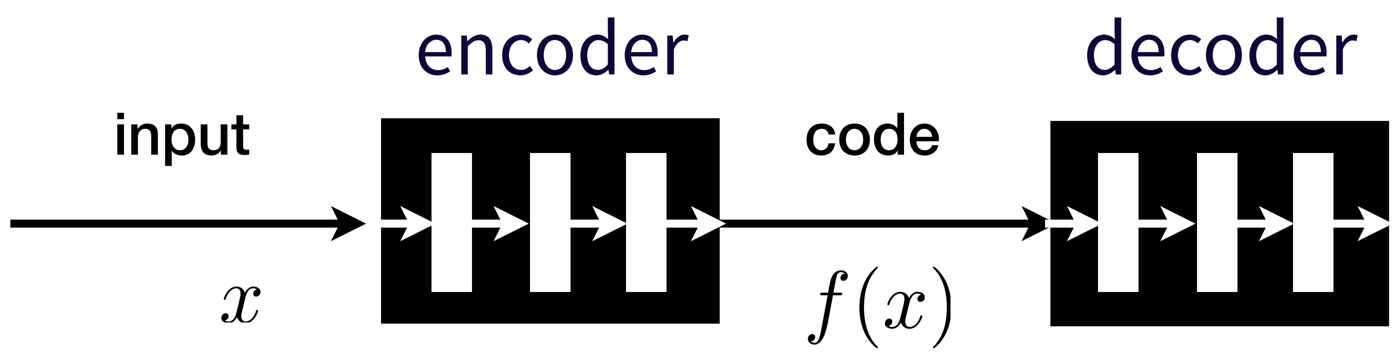
- Learn one or more binary RBMs in a “greedy” fashion
- Unroll to a deep autoencoder and “fine-tune” w/ backprop

Deep auto-encoders



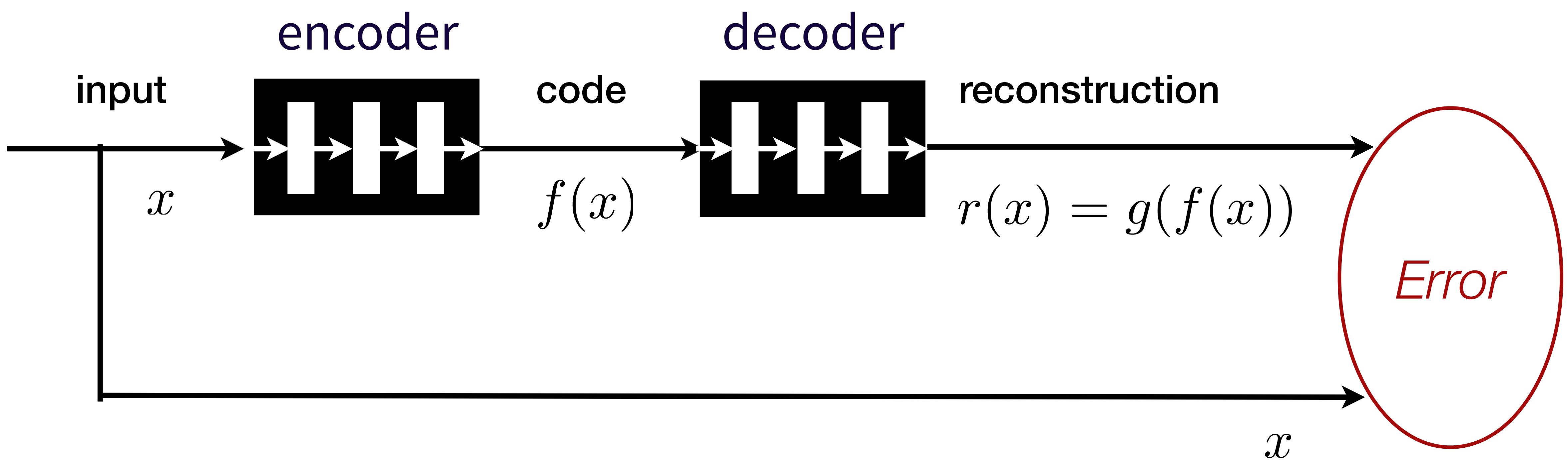
- Learn one or more binary RBMs in a “greedy” fashion
- Unroll to a deep autoencoder and “fine-tune” w/ backprop

Deep auto-encoders



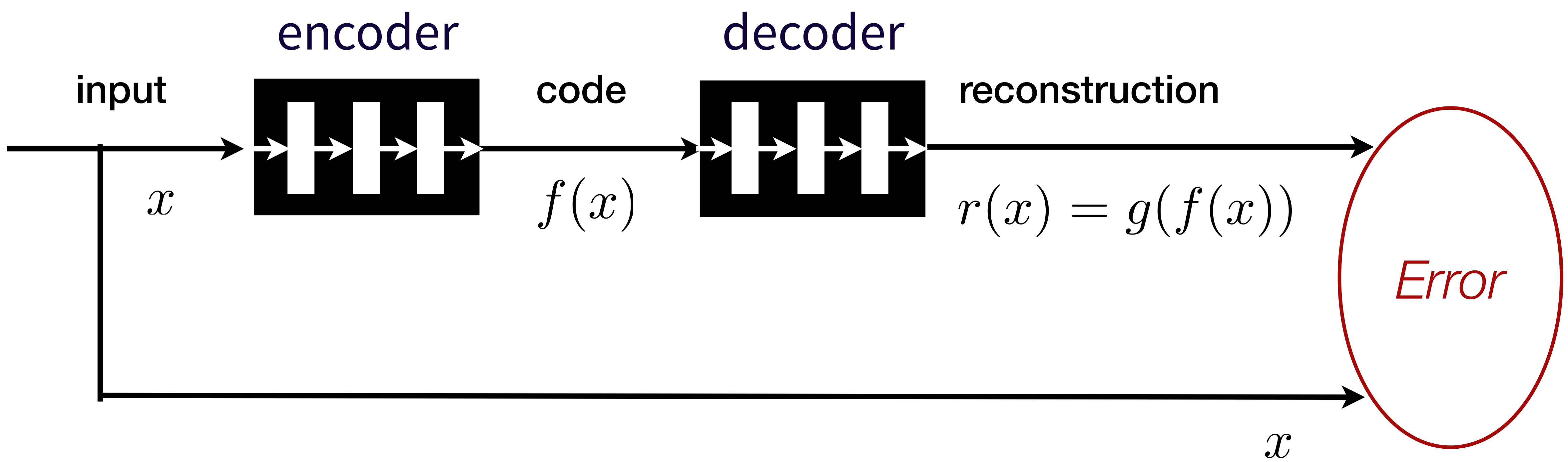
- Learn one or more binary RBMs in a “greedy” fashion
- Unroll to a deep autoencoder and “fine-tune” w/ backprop

Deep auto-encoders



- Learn one or more binary RBMs in a “greedy” fashion
- Unroll to a deep autoencoder and “fine-tune” w/ backprop

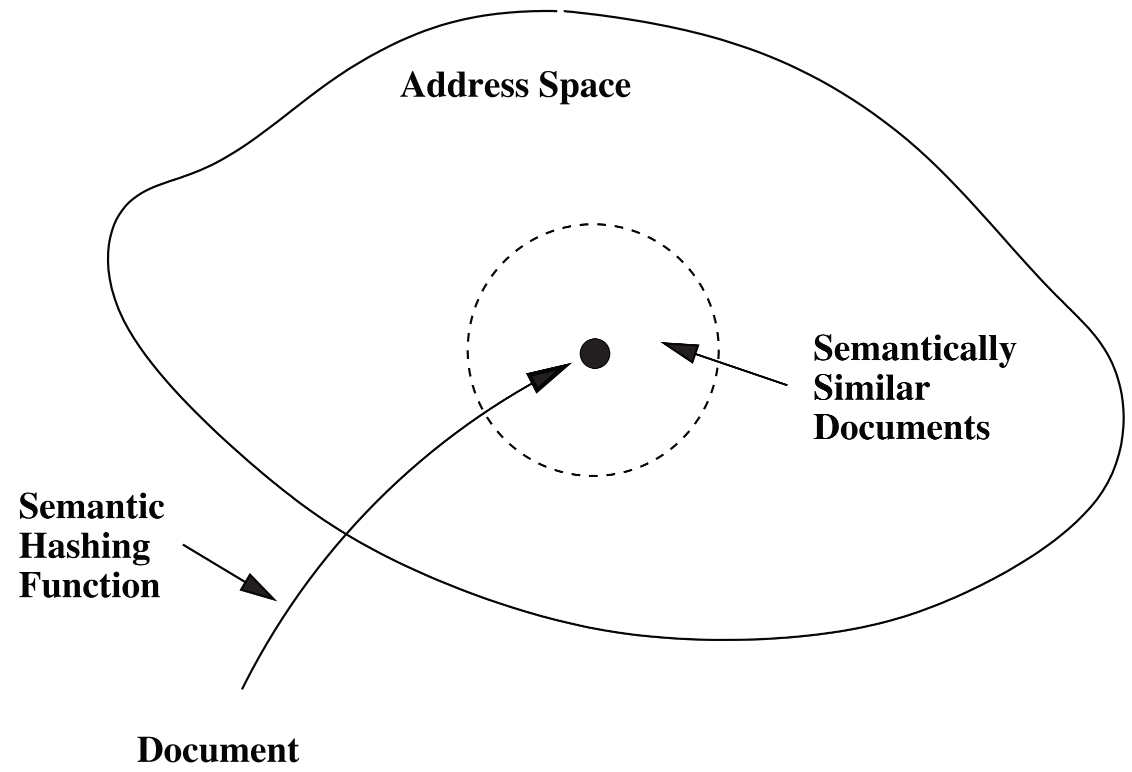
Deep auto-encoders

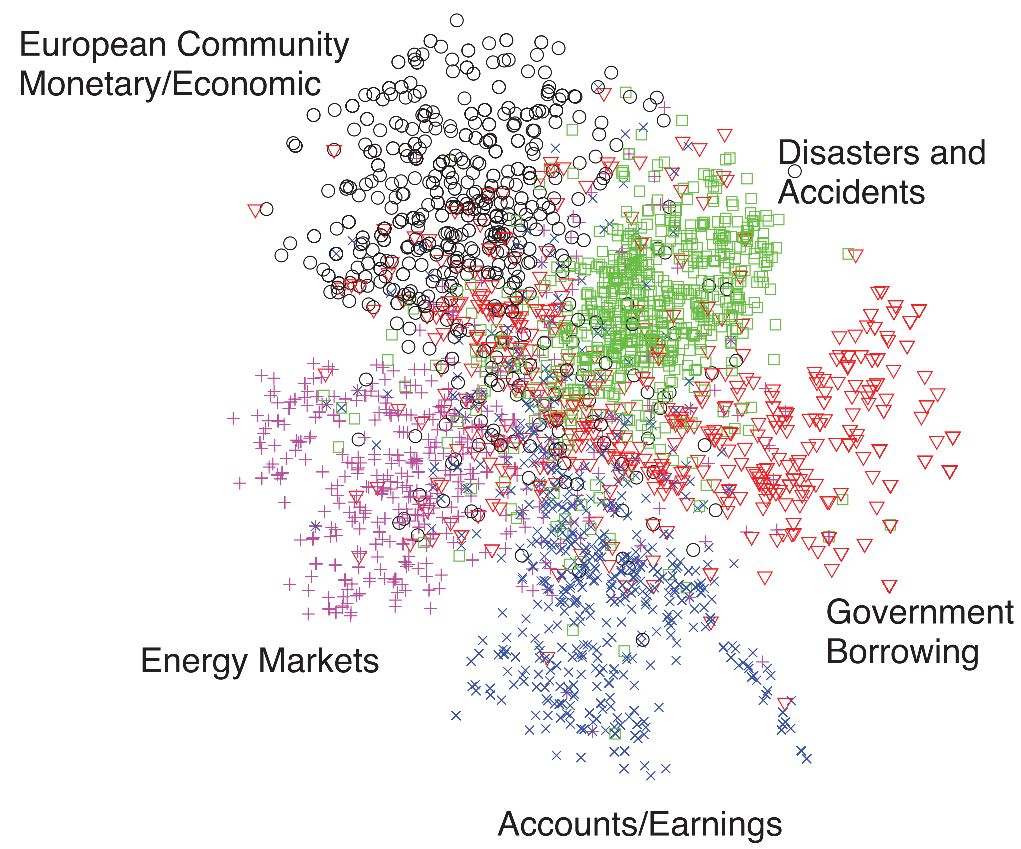


- Learn one or more binary RBMs in a “greedy” fashion
- Unroll to a deep autoencoder and “fine-tune” w/ backprop
  - During fine-tuning add Gaussian noise to code layer
  - This forces the codes to be close to binary

Extremely fast retrieval

- Documents are mapped to 20-D binary codes
- Can retrieve similar documents stored at nearby addresses with no search
- Binary LSA significantly reduces performance
    - Not surprising: it has not been optimized to make binary codes perform well
- One weakness: documents with similar addresses have similar content but the converse is not necessarily true
    - Can we use external information (e.g. labels) to pull together codes of similar documents?

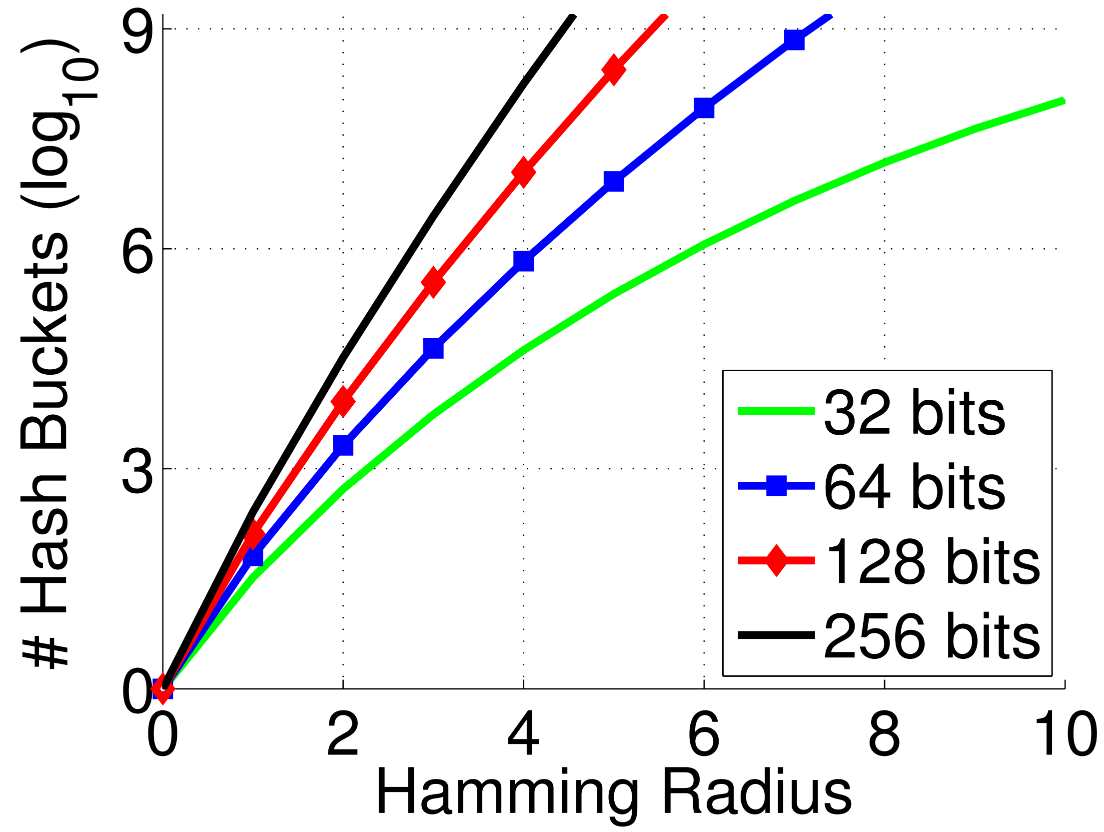


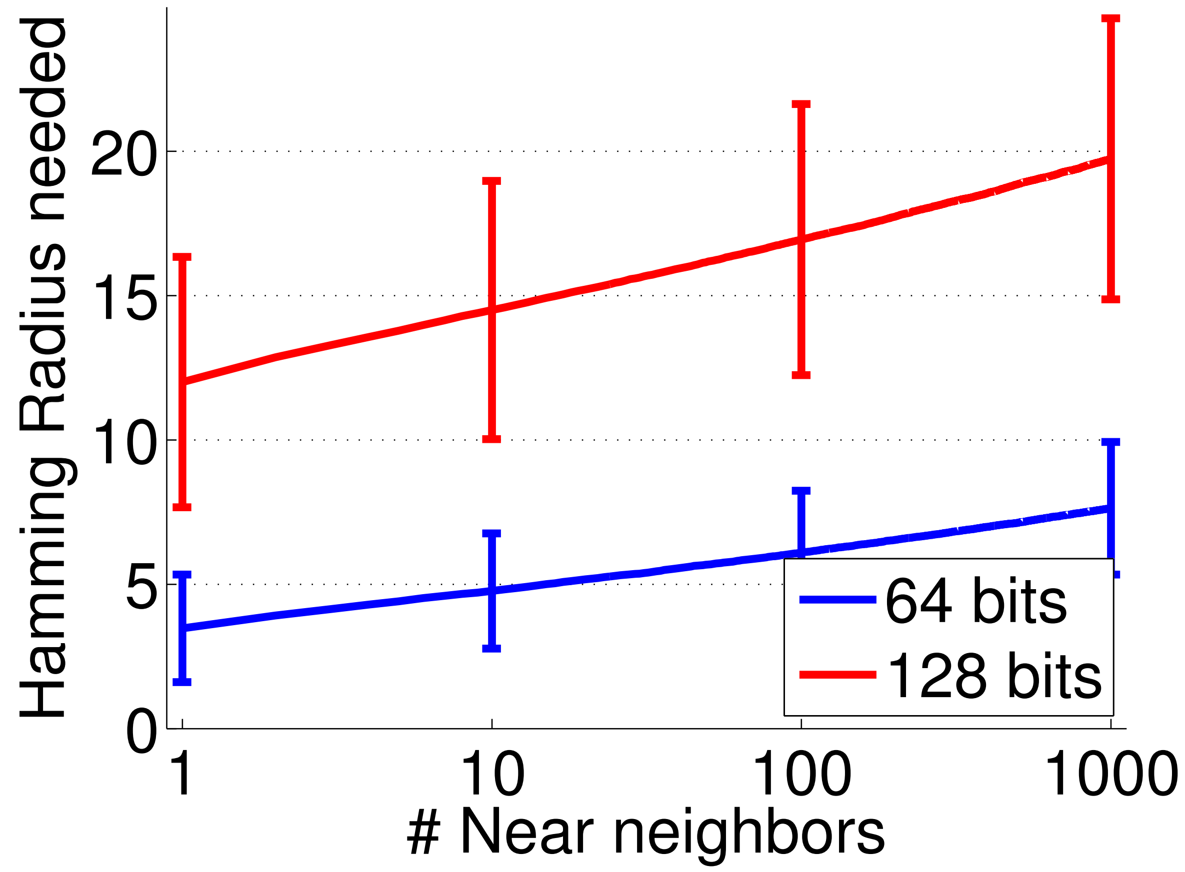


Figures from R. Salakhutdinov and G. Hinton

Hashing longer codes

- If code lengths are > 32 bits, use codes as direct indices (addresses) into a hash table
  - dramatic increase in search speed compared to exhaustive linear scan
- Code lengths are often much longer in order to achieve good performance
  - but number of hash buckets to examine grows near-exponentially with search radius





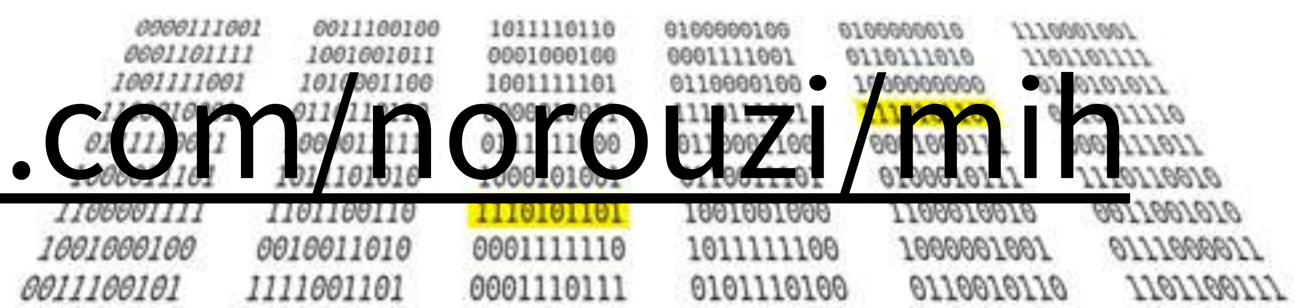
Figures: Norouzi et al. (2014)

Multi-index hashing

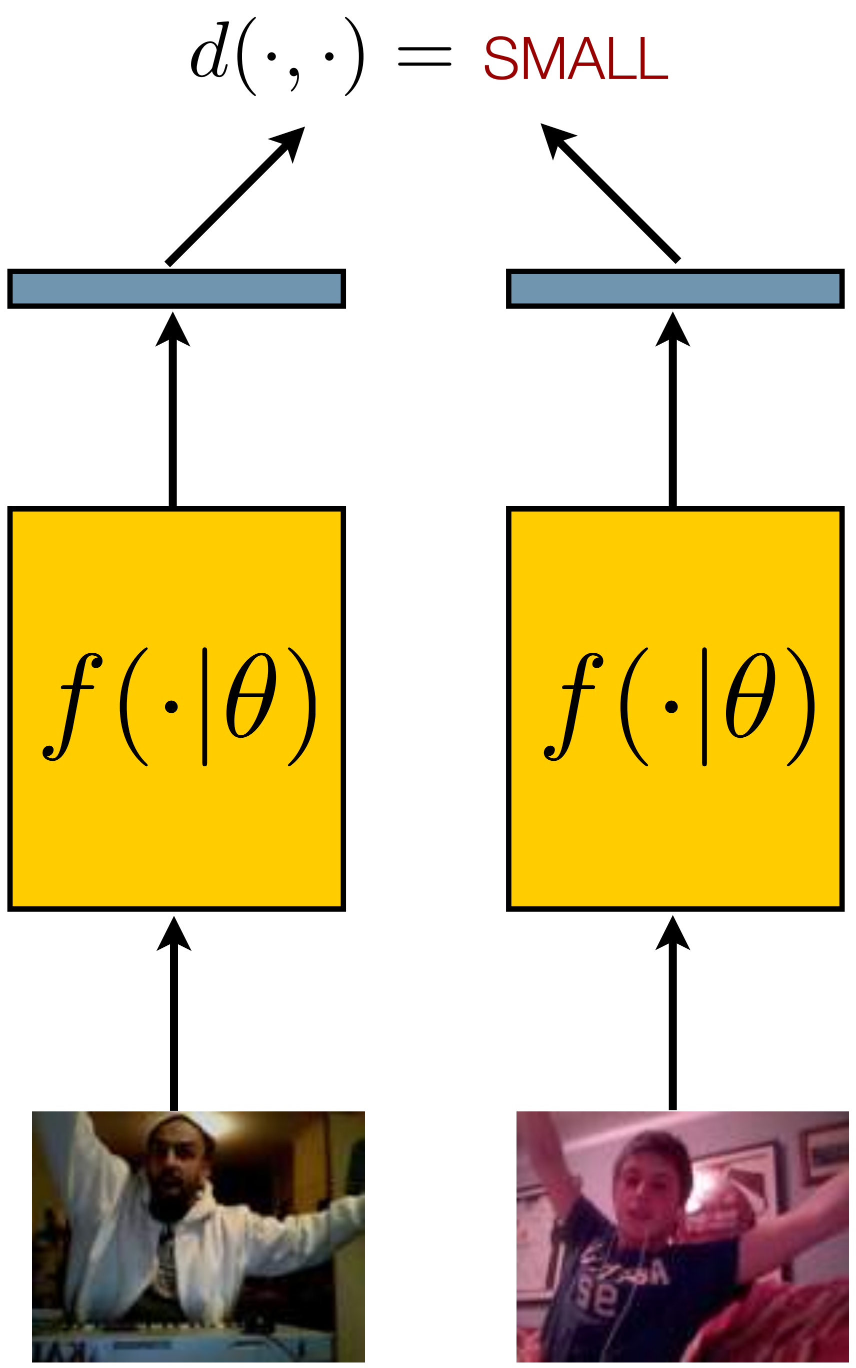
(Norouzi et al. 2012, 2014)

- When hash codes are > 32 bits, use Multi-index hashing
- Provably sub-linear search complexity for uniformly distributed codes
- Binary codes are indexed *m* times into *m* different hash tables, based on *m* disjoint substrings
- Given a query code, entries that fall close to the query in at least one such substring are considered neighbour candidates
- Candidates then checked for validity using entire binary code
- Guaranteed that all true neighbours will be found

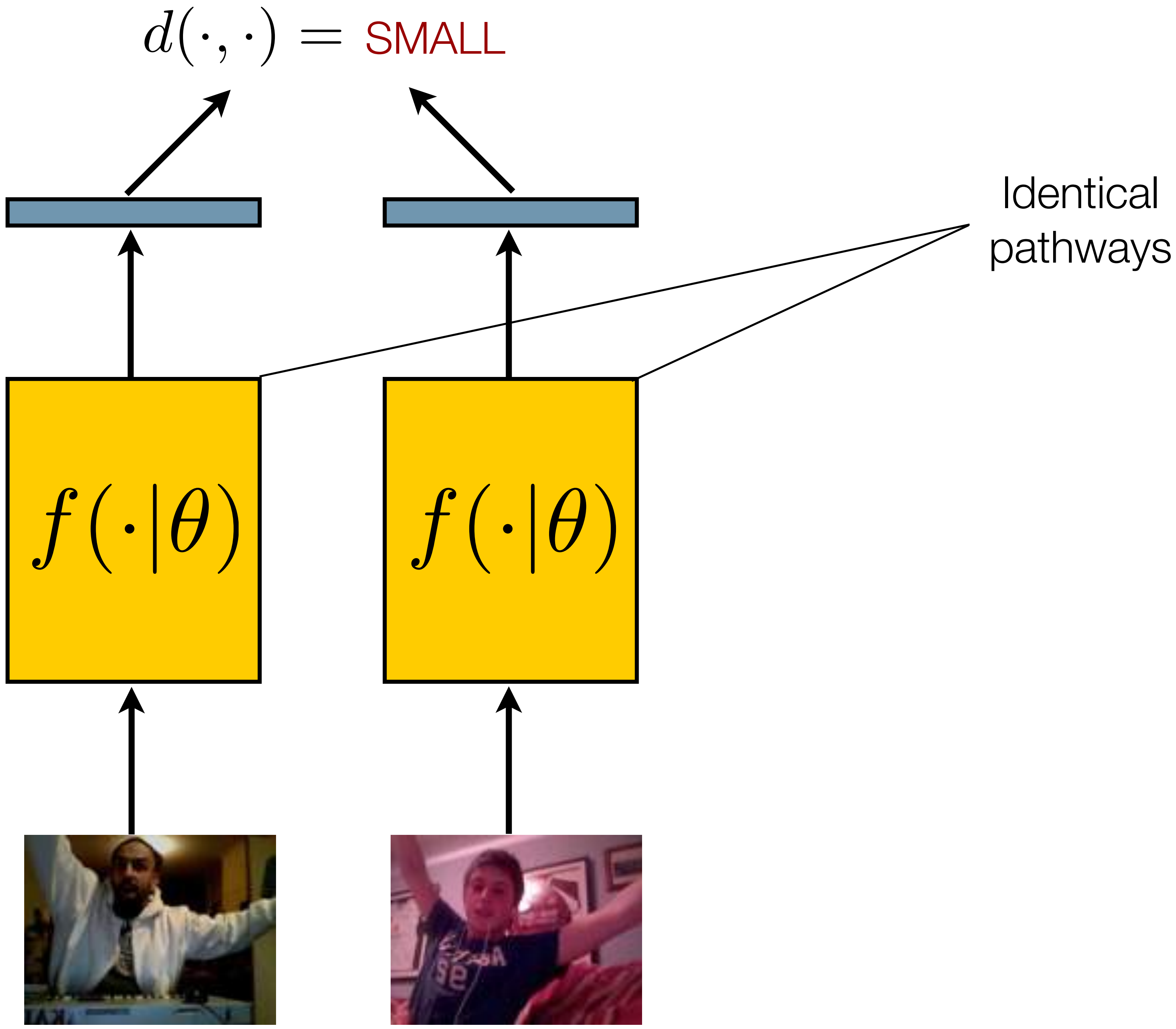
https://github.com/norouzi/mih



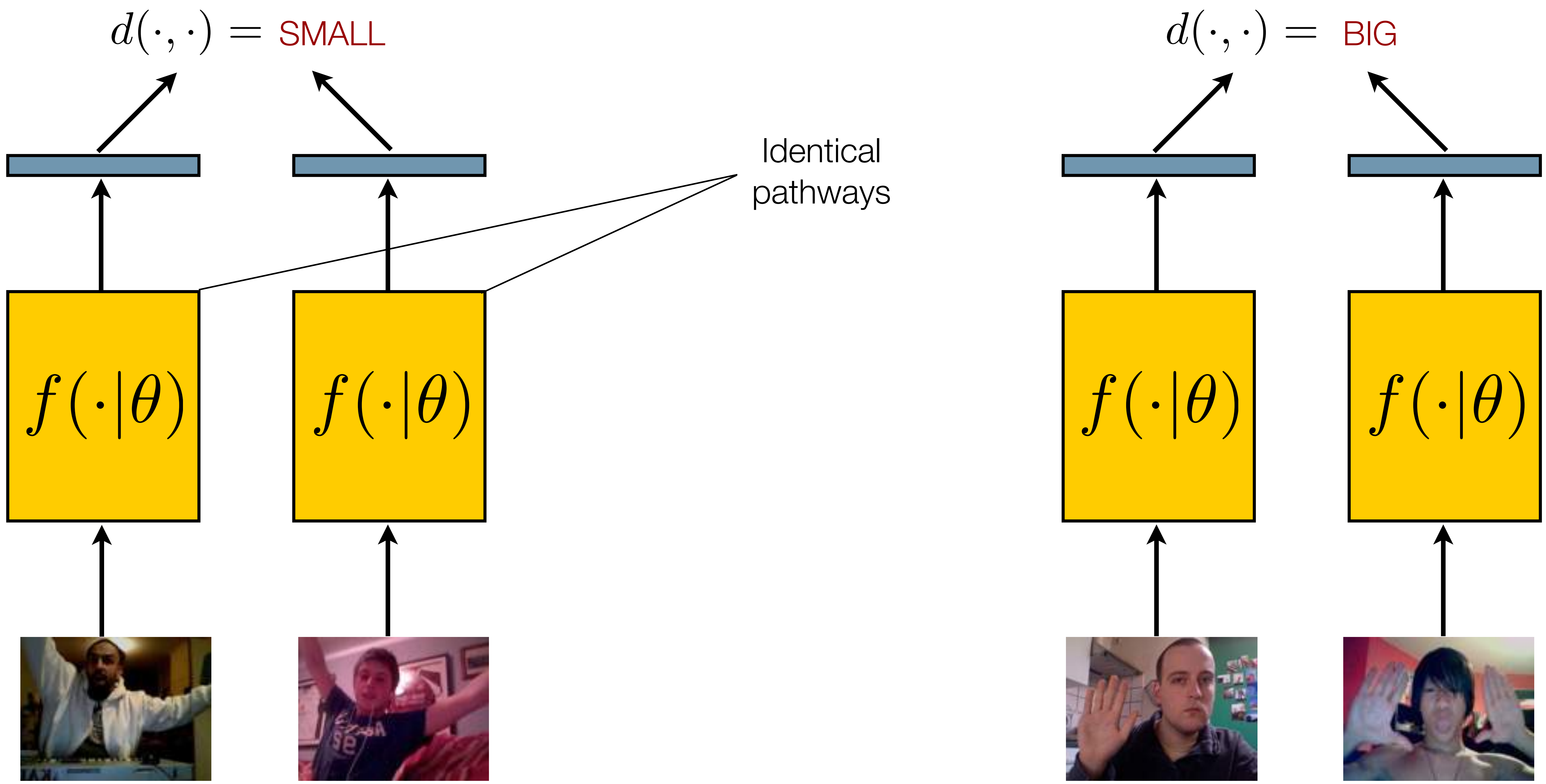
Learning embeddings with a Siamese network



Learning embeddings with a Siamese network



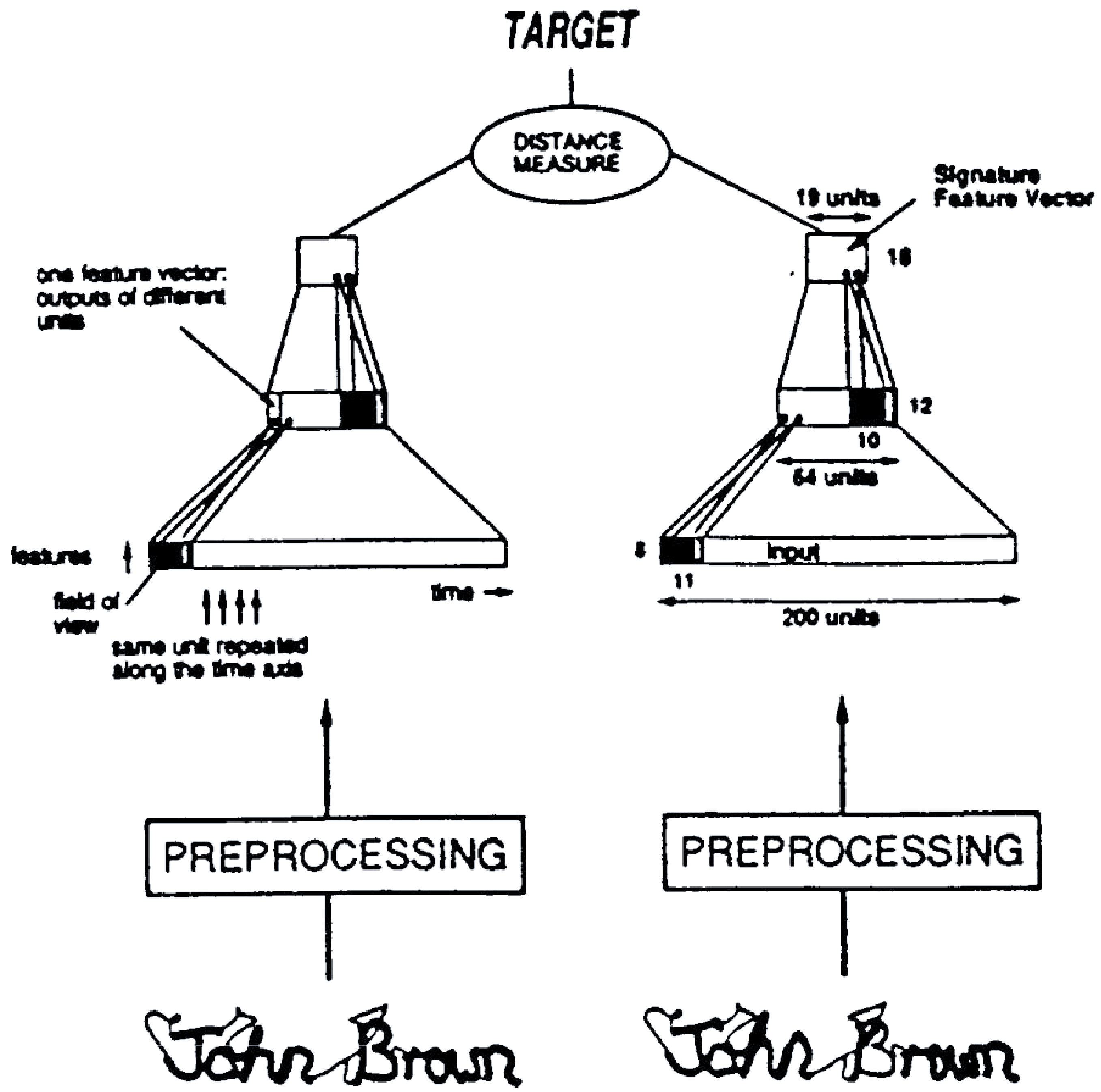
Learning embeddings with a Siamese network



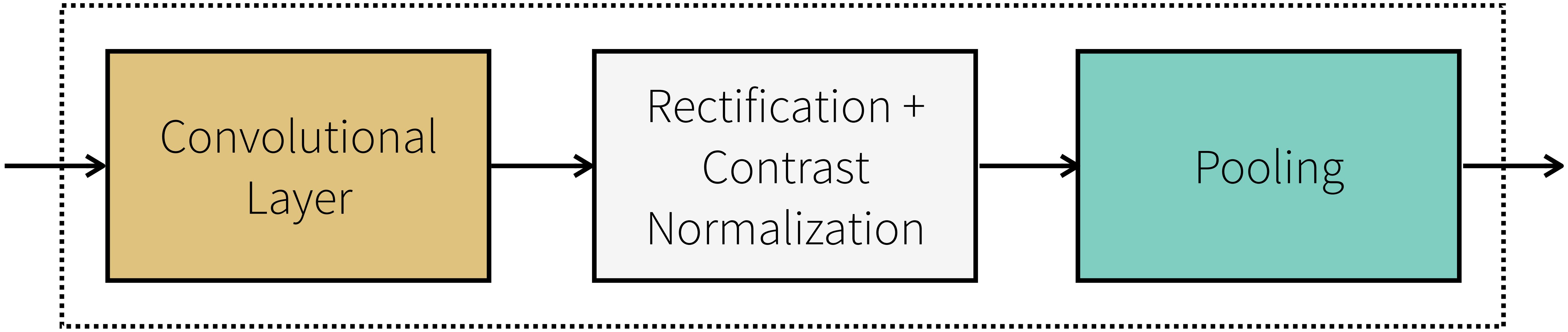
Not a new idea!

(Bromley, Guyon, LeCun, Sackinger, and Shah 1994)

- Architecture proposed for signature verification
  - didn't really get the distance function right
  - learning unstable
  - small (by today's standards) training set
- 1D convolution (TDNN)
- Developed independently elsewhere:
  - Baldi and Chauvin, 1992: fingerprint verification
  - Becker and Hinton, 1992 - discovering depth in random-dot stereograms



Convnets: single stage



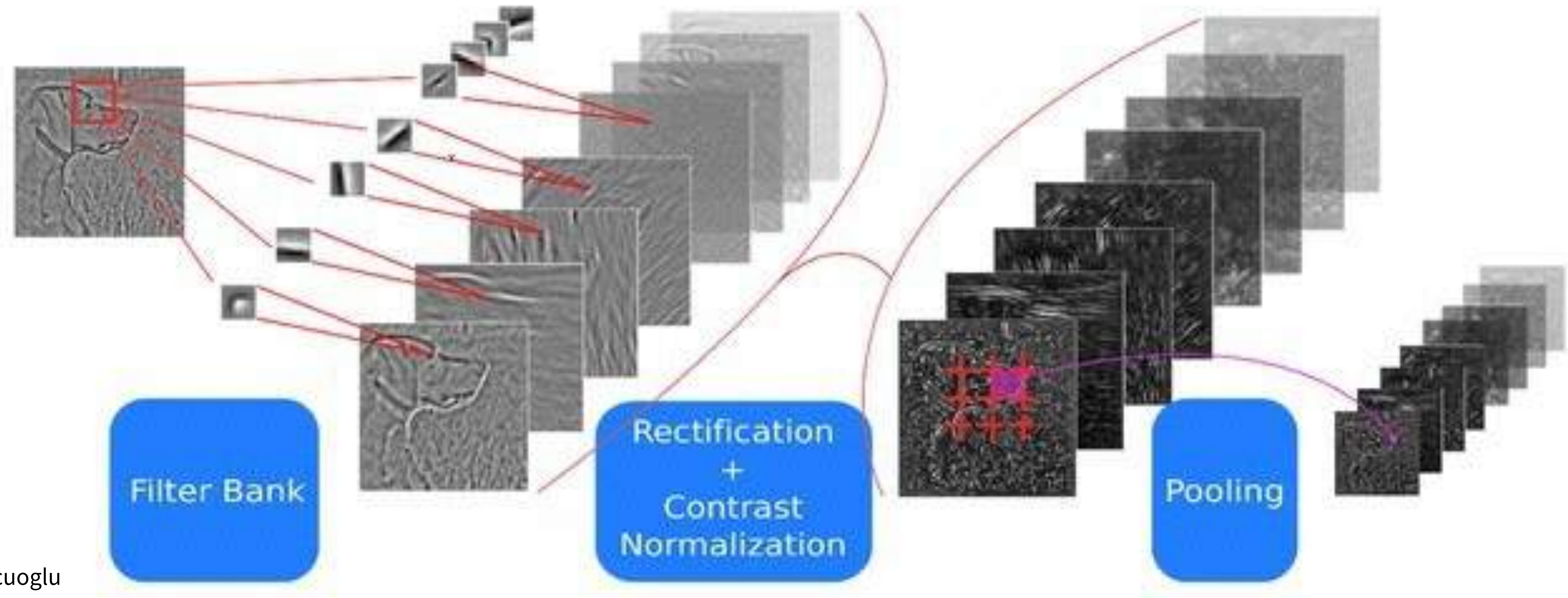
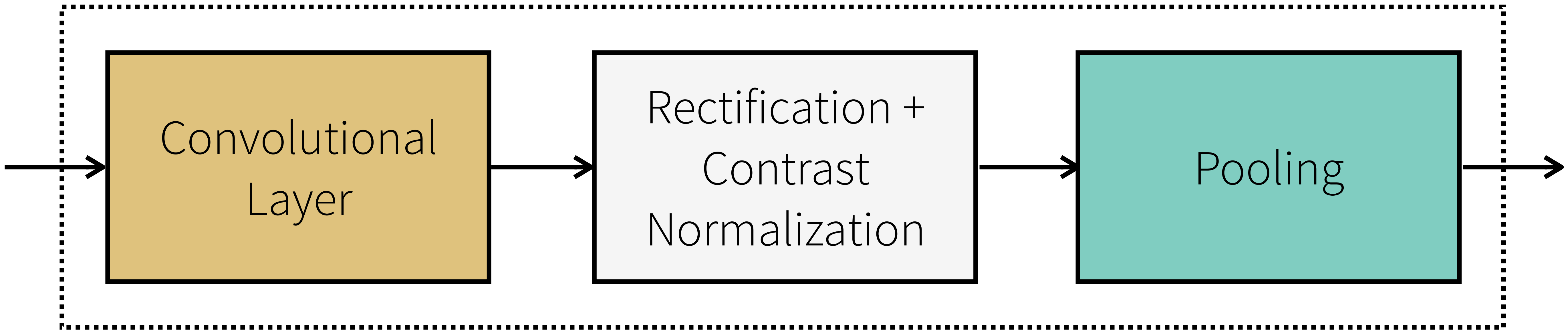


Image credit: Koray Kavukcuoglu

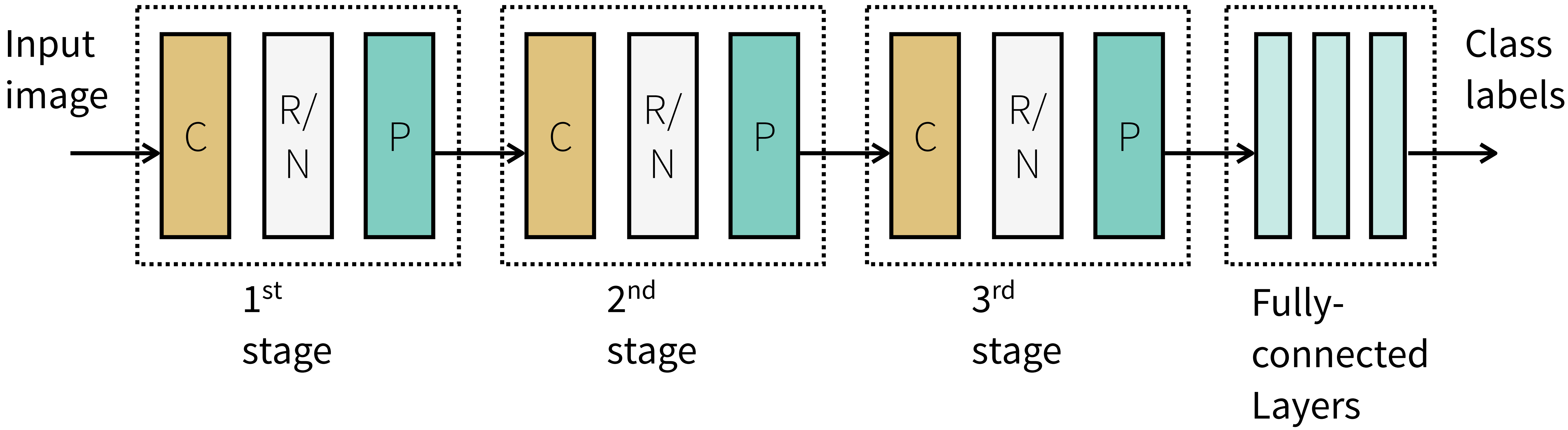
Credit: Marc'Aurelio Ranzato

Convnets: typical architecture

Single stage

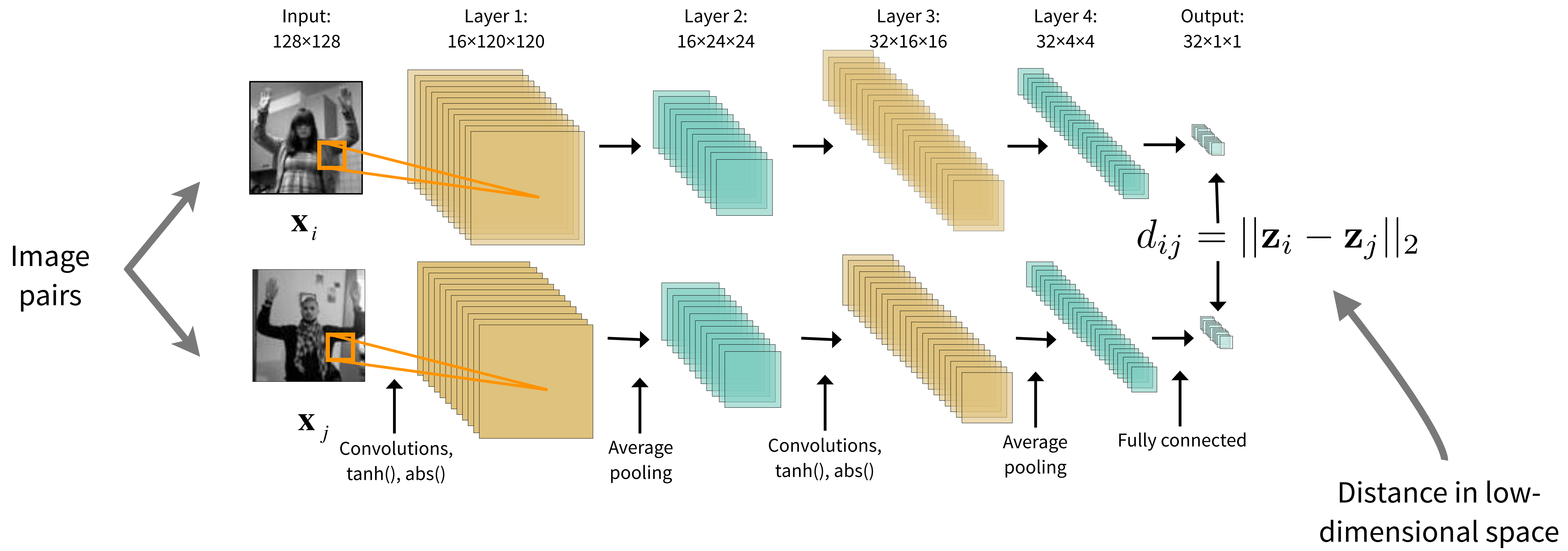


Whole system



Credit: Marc'Aurelio Ranzato

Embedding with a Siamese convnet



What's the objective function?

-needs to pull together semantically similar pairs

-needs to push apart semantically dissimilar pairs

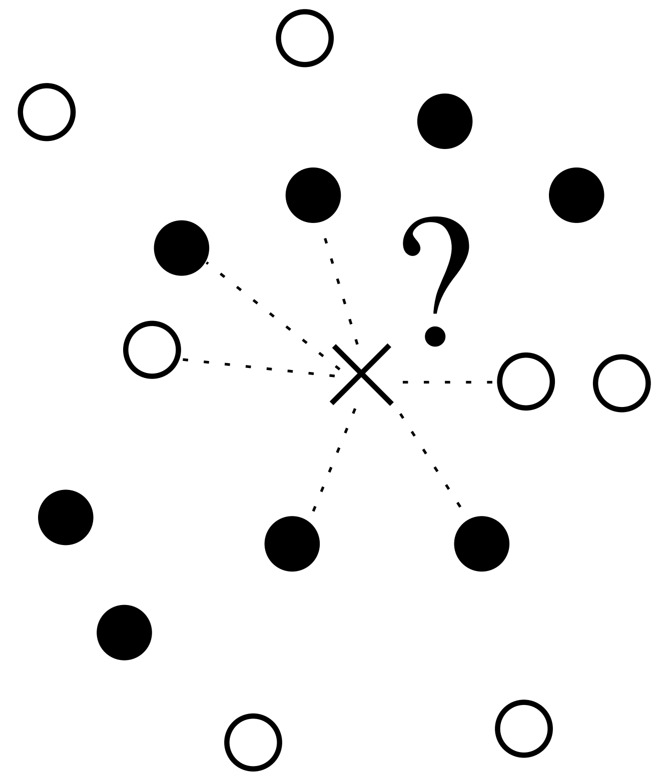
Training Siamese nets

(Bromley, Guyon, LeCun, Sackinger, and Shah 1994)

- Siamese nets can be trained by error backpropagation, just need to define an objective function:
  - Neighbourhood Component Analysis (Goldberger et al. 2004)
  - Dimensionality Reduction by Learning an Invariant Mapping (Hadsell et al. 2006)
  - Triplet-based Criterion (Chechik et al. 2010)
  - Quadruplet-based Criterion (Law et al. 2013)

Neighbourhood components analysis (NCA)

(Goldberger et al. 2004)

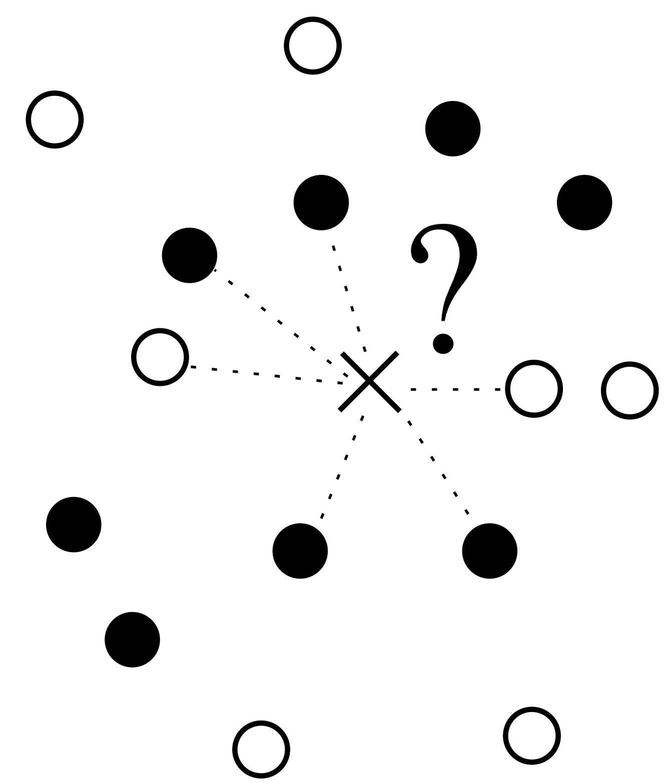


Credit: Sam Roweis

Neighbourhood components analysis (NCA)

(Goldberger et al. 2004)

- Learn a metric which minimizes KNN classification error



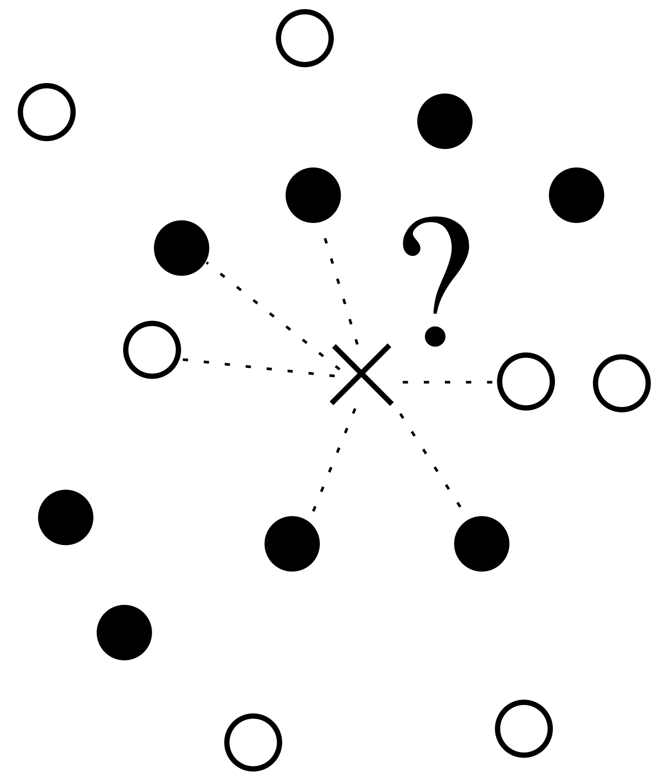
Credit: Sam Roweis

Neighbourhood components analysis (NCA)

(Goldberger et al. 2004)

- Learn a metric which minimizes KNN classification error

- Two problems:

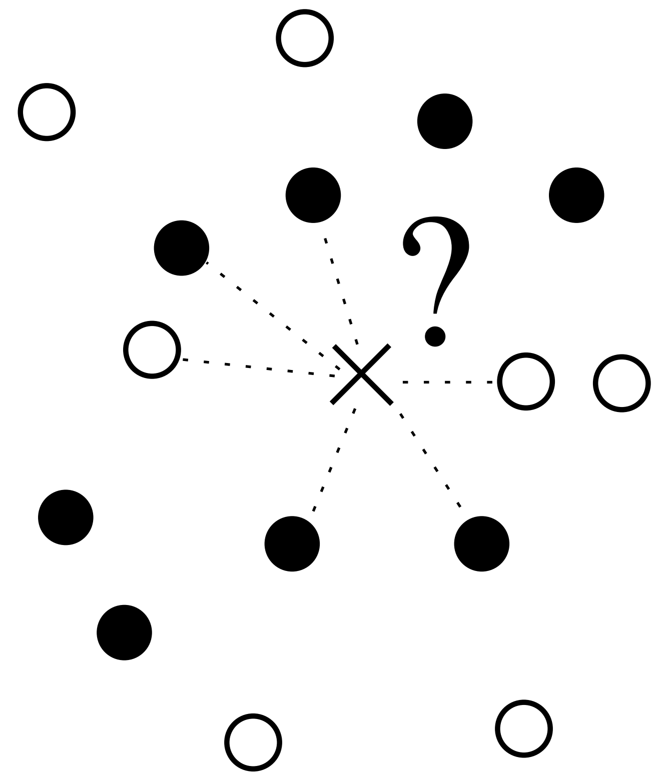


Credit: Sam Roweis

Neighbourhood components analysis (NCA)

(Goldberger et al. 2004)

- Learn a metric which minimizes KNN classification error
- Two problems:
  - Error is a highly discontinuous function of the distance metric

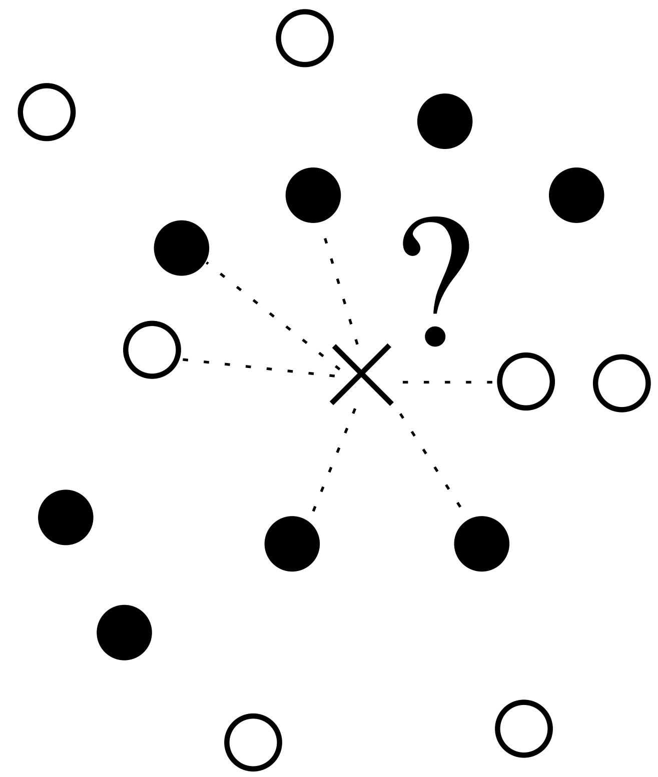


Credit: Sam Roweis

Neighbourhood components analysis (NCA)

(Goldberger et al. 2004)

- Learn a metric which minimizes KNN classification error
- Two problems:
    - Error is a highly discontinuous function of the distance metric
    - We still need to choose K

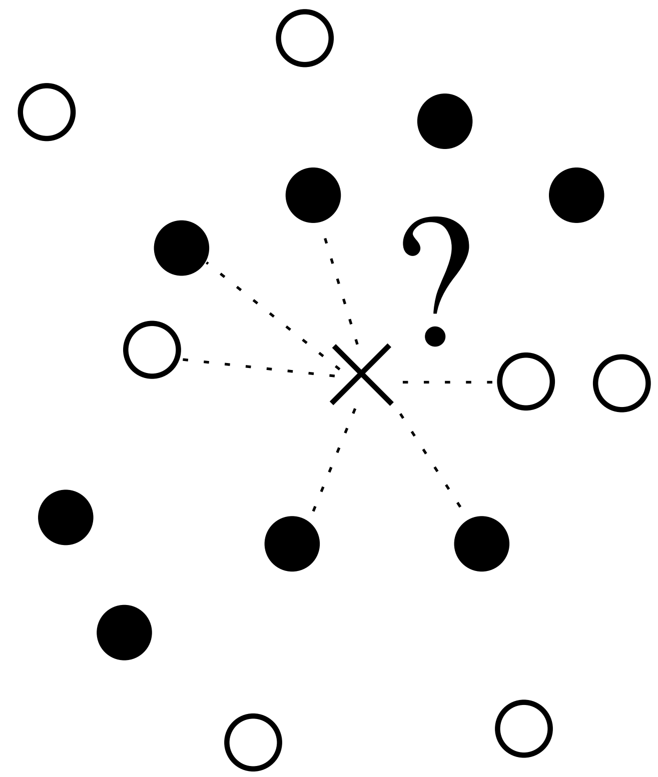


Credit: Sam Roweis

Neighbourhood components analysis (NCA)

(Goldberger et al. 2004)

- Learn a metric which minimizes KNN classification error
- Two problems:
  - Error is a highly discontinuous function of the distance metric
  - We still need to choose K
- Look for a smoother (or at least continuous) cost function



Credit: Sam Roweis

Stochastic nearest neighbour

Figure: Sam Roweis

Stochastic nearest neighbour

- Instead picking from a fixed set of K nearest neighbours, select a single neighbour stochastically

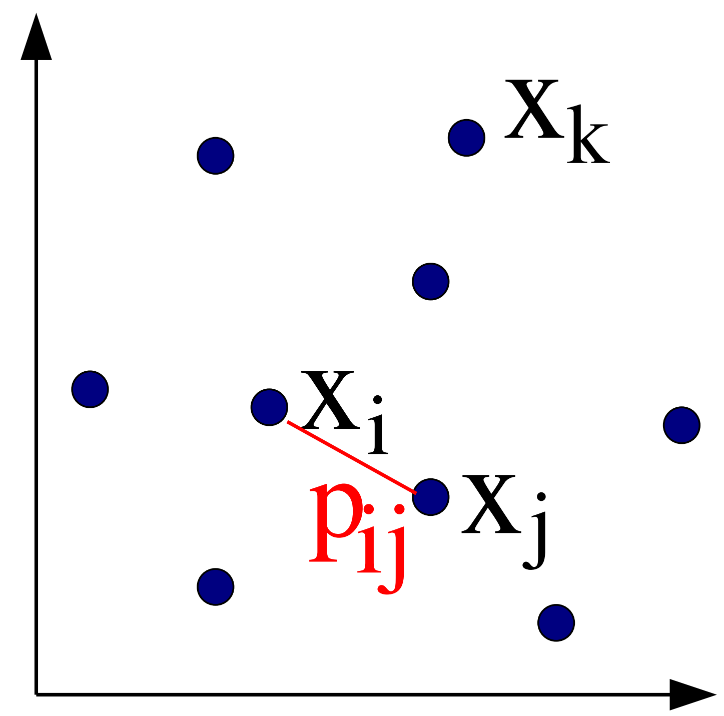


Figure: Sam Roweis

Stochastic nearest neighbour

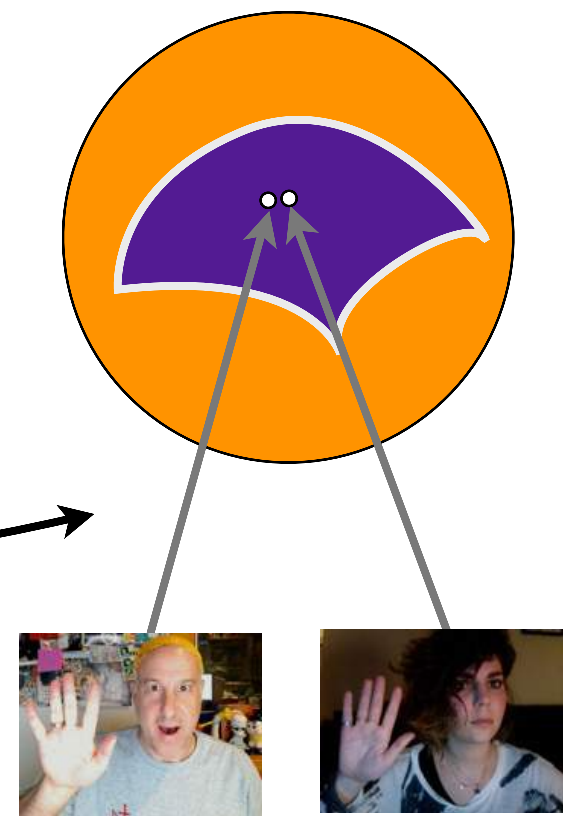
- Instead picking from a fixed set of K nearest neighbours, select a single neighbour stochastically
- Let each point i select other points j as its neighbour with probability p_{ij} based on the softmax of the distance d_{ij} :

$$p_{ij} = \frac{\exp(-d_{ij}^2)}{\sum_{k \neq i} \exp(-d_{ik}^2)}$$

where:

$$d_{ij} = ||\mathbf{z}_i - \mathbf{z}_j||_2$$

$$\mathbf{z}_i = f(\mathbf{x}_i|\theta)$$



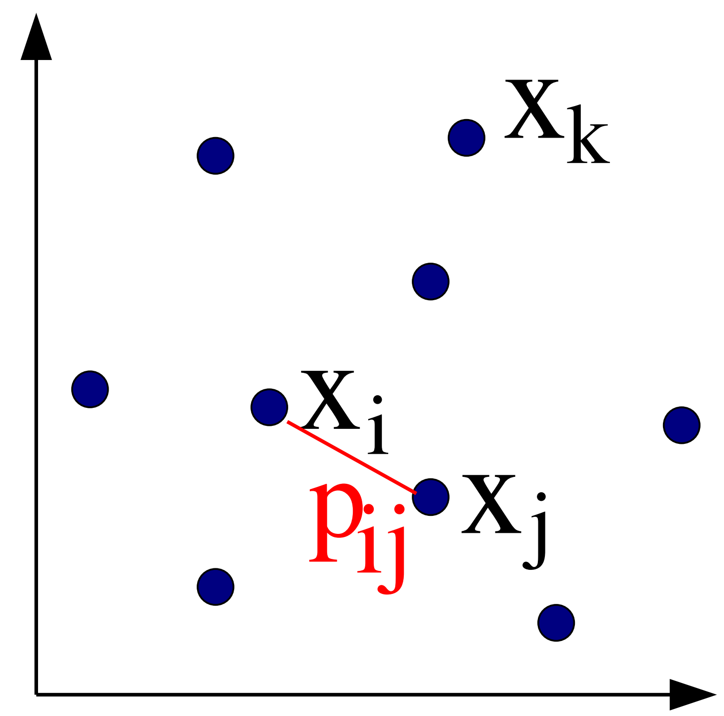


Figure: Sam Roweis

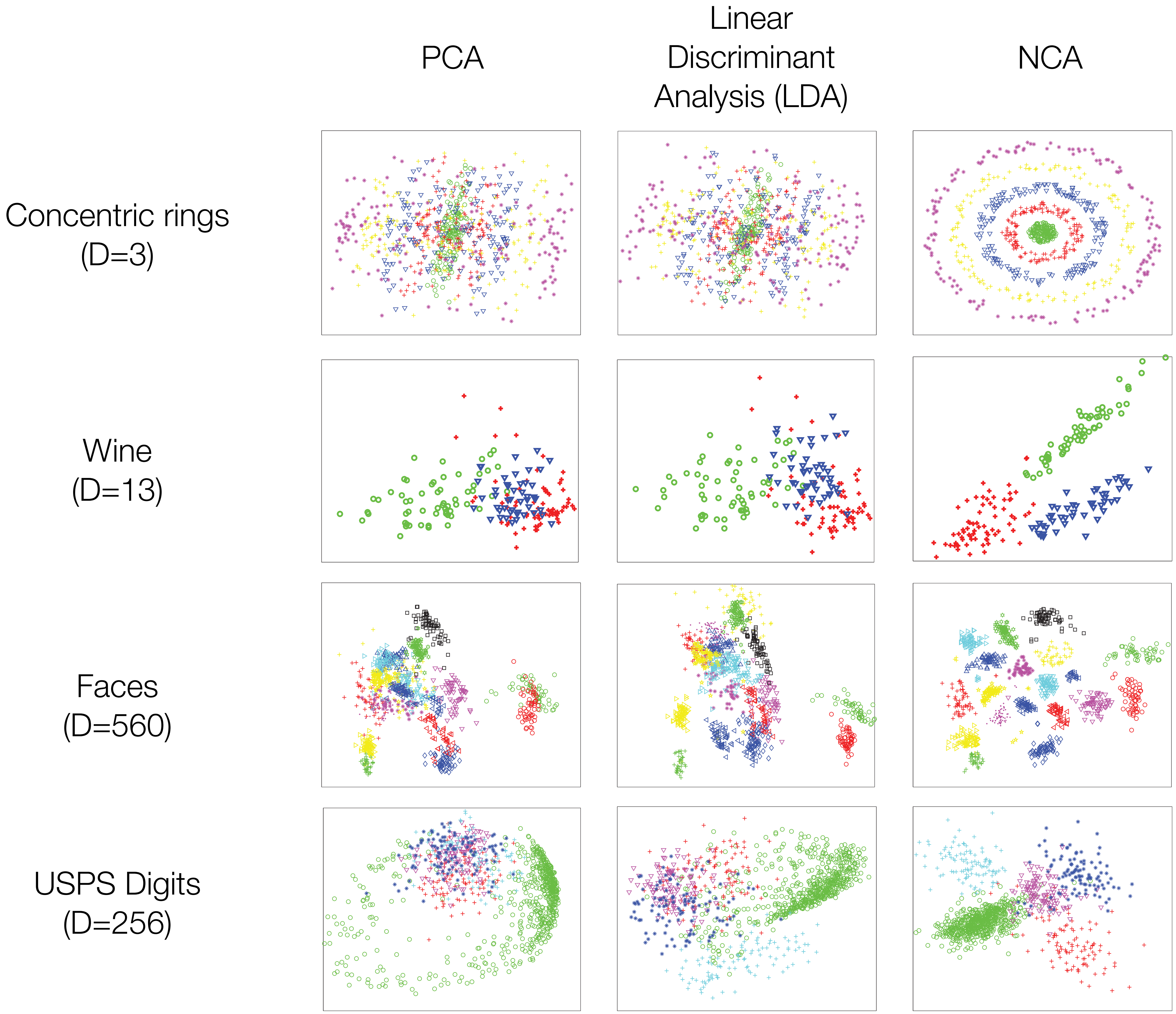
NCA: loss

- Maximize the expected number of points correctly classified under this scheme
- This is much smoother than the actual leave-one-out cross-validation error!
- In fact, it is differentiable w.r.t. parameters of mapping
  - can use SGD or other gradient-based optimizer
- And there is no explicit parameter K
  - See (Tarlow et al. 2013) for $K > 1$ objective

$$L_{\mathrm{NCA}} = -\sum_{i=1}^{N} \sum_{j:y_i=y_j} p_{ij}$$

Minimize loss w.r.t. θ

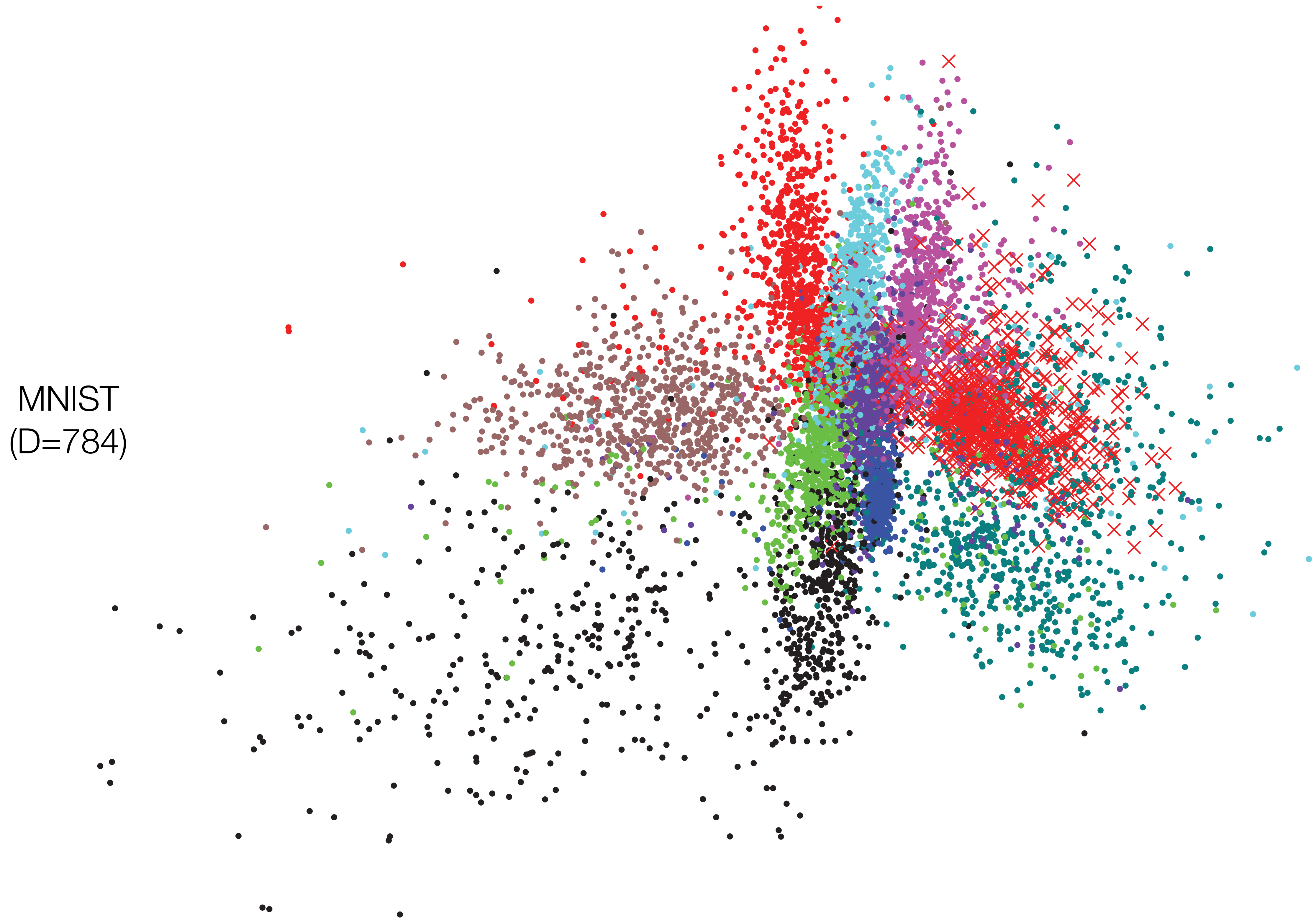
Linear NCA: embeddings



$$f(\mathbf{x}|\theta = A) = A\mathbf{x}$$

Figures: Goldberger et al.

NCA: MNIST



Nonlinear NCA

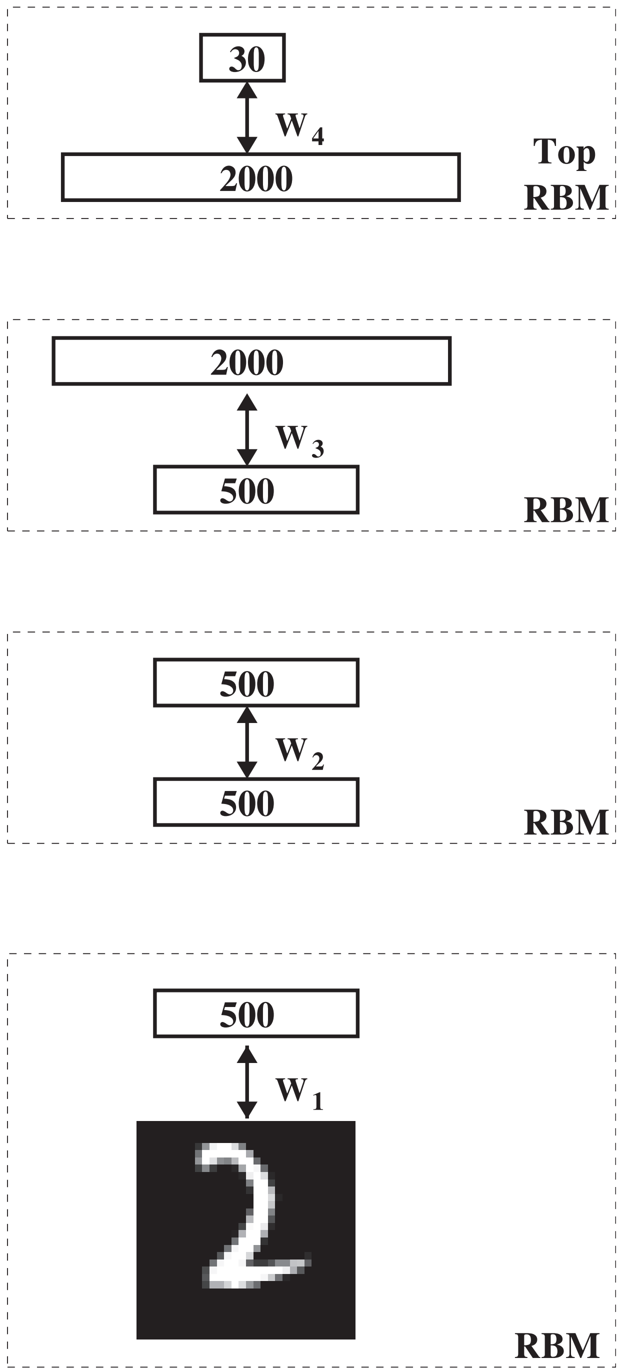
- The original NCA paper (Goldberger et al. 2004) points out that $f(\mathbf{x}_i|\theta)$ need not be a linear mapping
- Salakhutdinov and Hinton (2007) pre-train with an RBM, then fine-tune with the NCA objective
- Can combine the NCA objective with an Autoencoder objective to regularize:

$$C = \lambda L_{\mathrm{NCA}} + (1-\lambda) L_{AE}$$

- Can take advantage of unlabeled data!

Learning nonlinear NCA

Pre-training



Mixed-objective fine-tuning

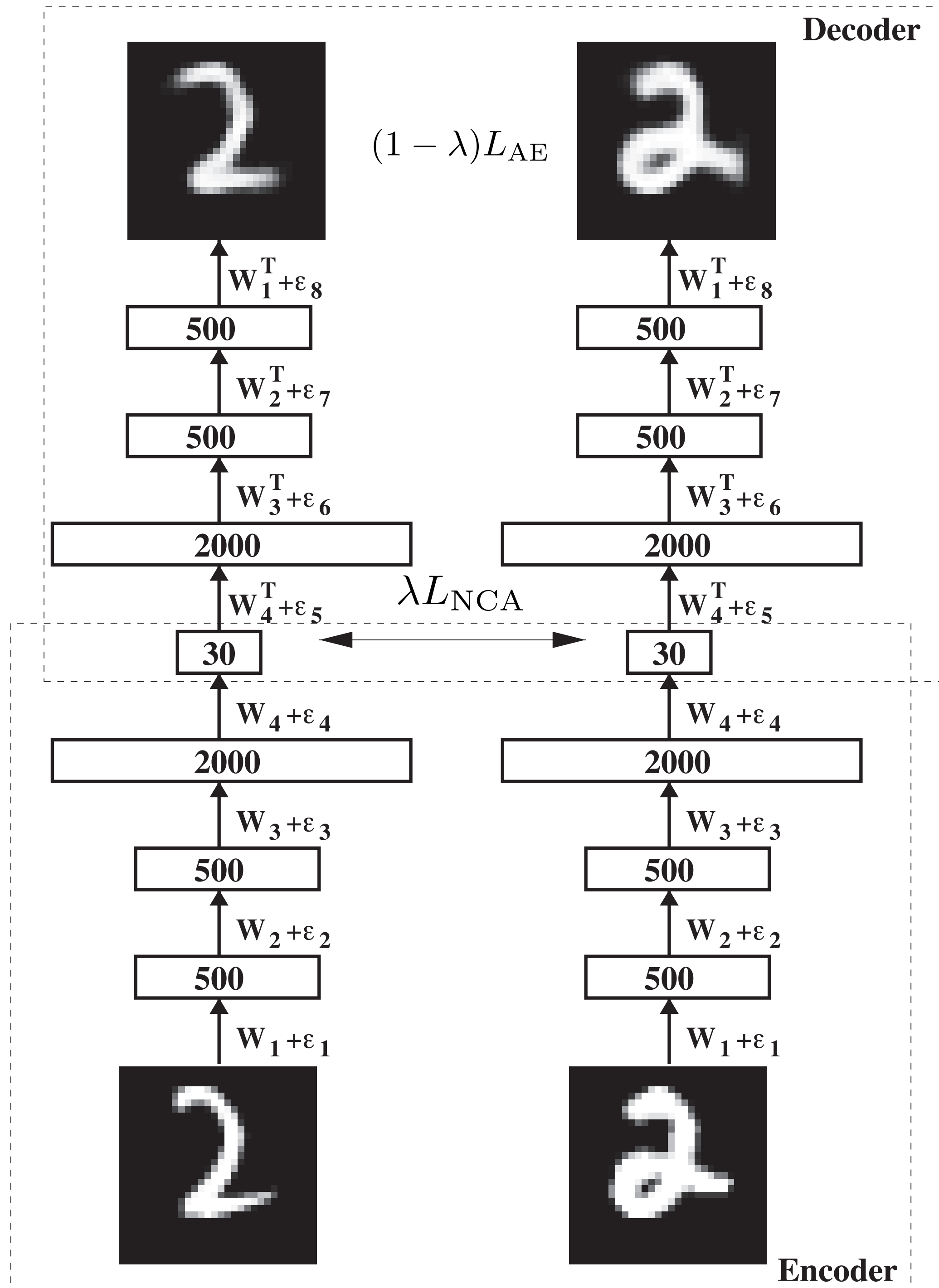
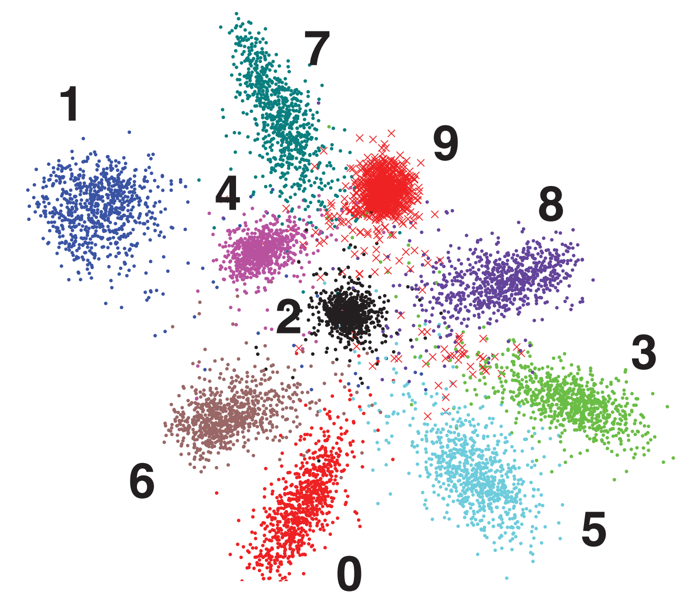


Figure: Salakhutdinov and Hinton

Limitations of NCA

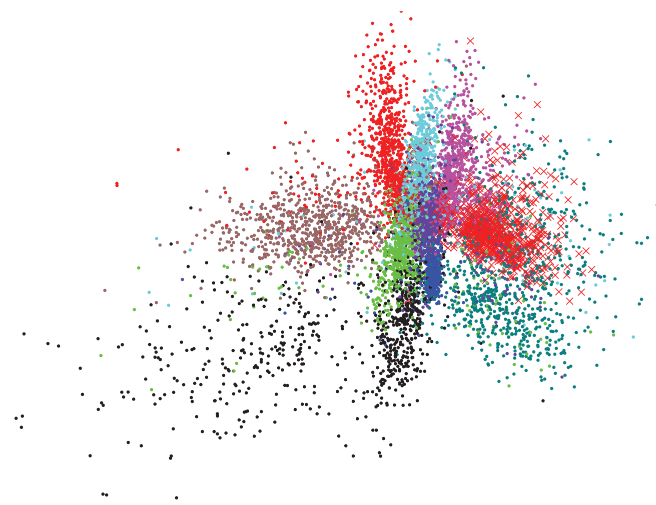
- Despite very nice embeddings (see right) NCA has a quadratic normalization term (must consider all pairs)
    - mini-batch training (approximate)
    - objectives that don't require normalization

Noninear NCA (MNIST)



- What about continuous labels?
    - (Goldberger et al. 2004) describe a "soft" form of NCA that can use continuous labels

Linear NCA (MNIST)



(Figures from R. Salakhutdinov and G. Hinton)

Class-conditional metric learning

(Im and Taylor - In submission)

Daniel Im (here at DLSS!)

Class-conditional metric learning

(Im and Taylor - In submission)

- Optimize Image-to-Class distance (Boiman et al. 2008)

Daniel Im (here at DLSS!)

Class-conditional metric learning

(Im and Taylor - In submission)

- Optimize Image-to-Class distance (Boiman et al. 2008)
- Stochastic neighbour selection rule:

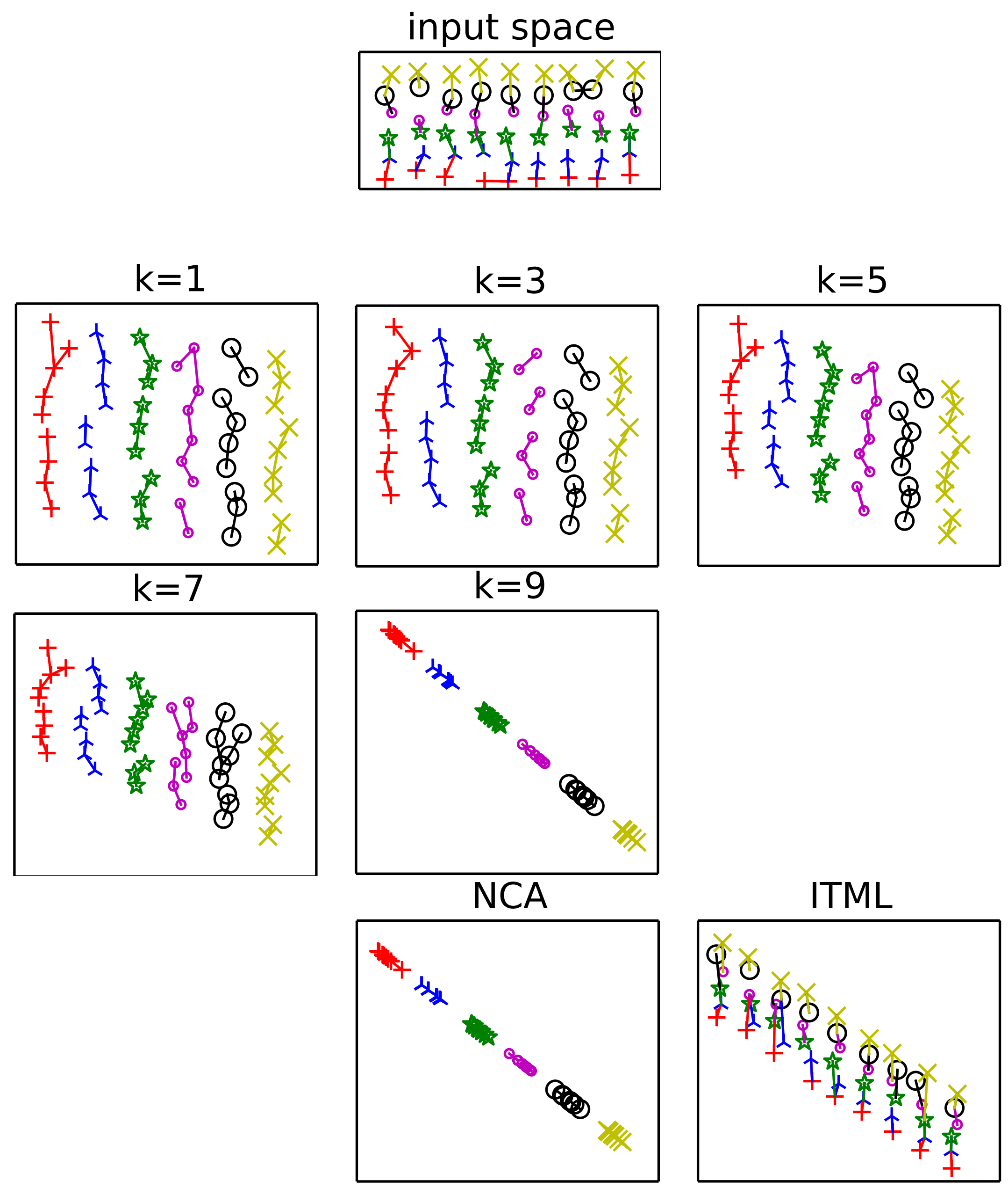
$$p_i^C = \frac{\exp\left(-\frac{1}{k}\sum_{j=1}^{k} ||\mathbf{z}_i - \mathrm{NN}_j^C(\mathbf{z}_i)||^2\right)}{\sum_{C'} \exp\left(-\frac{1}{k}\sum_{j=1}^{k} ||\mathbf{z}_i - \mathrm{NN}_j^{C'}(\mathbf{z}_i)||^2\right)} \ ,$$

Daniel Im (here at DLSS!)

Class-conditional metric learning

(Im and Taylor - In submission)

- Optimize Image-to-Class distance (Boiman et al. 2008)
- Stochastic neighbour selection rule:



$$p_i^C = \frac{\exp\left(-\frac{1}{k}\sum_{j=1}^{k}||\mathbf{z}_i - \mathrm{NN}_j^C(\mathbf{z}_i)||^2\right)}{\sum_{C'}\exp\left(-\frac{1}{k}\sum_{j=1}^{k}||\mathbf{z}_i - \mathrm{NN}_j^{C'}(\mathbf{z}_i)||^2\right)},$$

Daniel Im (here at DLSS!)

DrLIM (Dimensionality reduction by learning an invariant mapping)

s_{ij} is a binary indicator

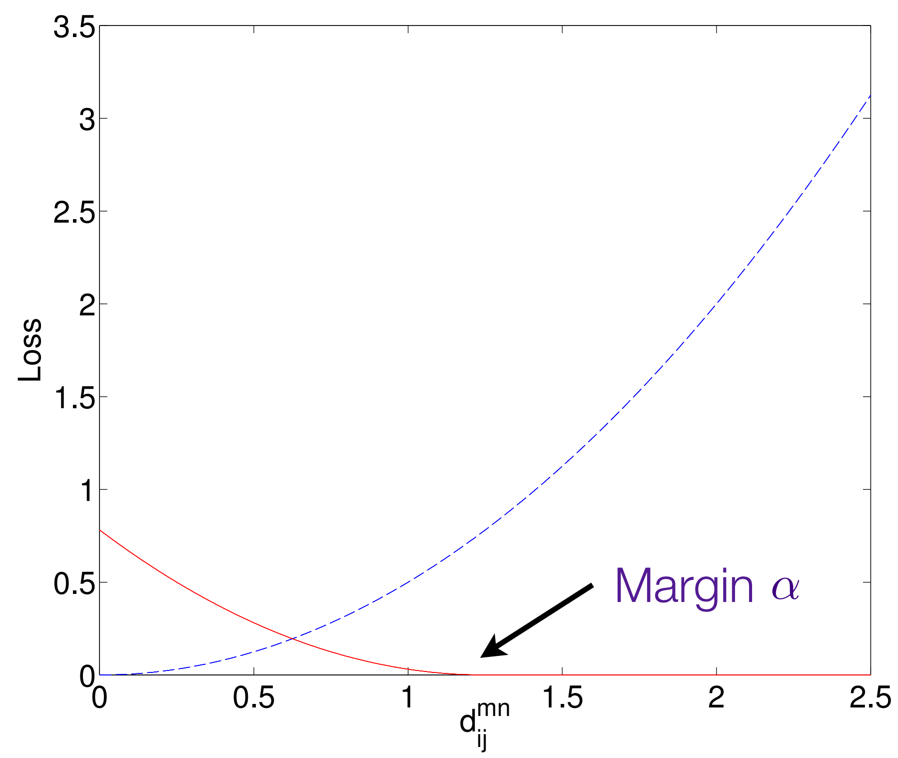
$$L = s_{ij} L_S(\mathbf{x}_i, \mathbf{x}_j) + (1 - s_{ij}) L_D(\mathbf{x}_i, \mathbf{x}_j)$$

Similarity loss

Dissimilarity loss

$$L_S(\mathbf{x}_i, \mathbf{x}_j) = \frac{1}{2}(d_{ij})^2$$

$$L_D(\mathbf{x}_i, \mathbf{x}_j) = \frac{1}{2}\left[\max(0, \alpha - d_{ij})\right]^2$$

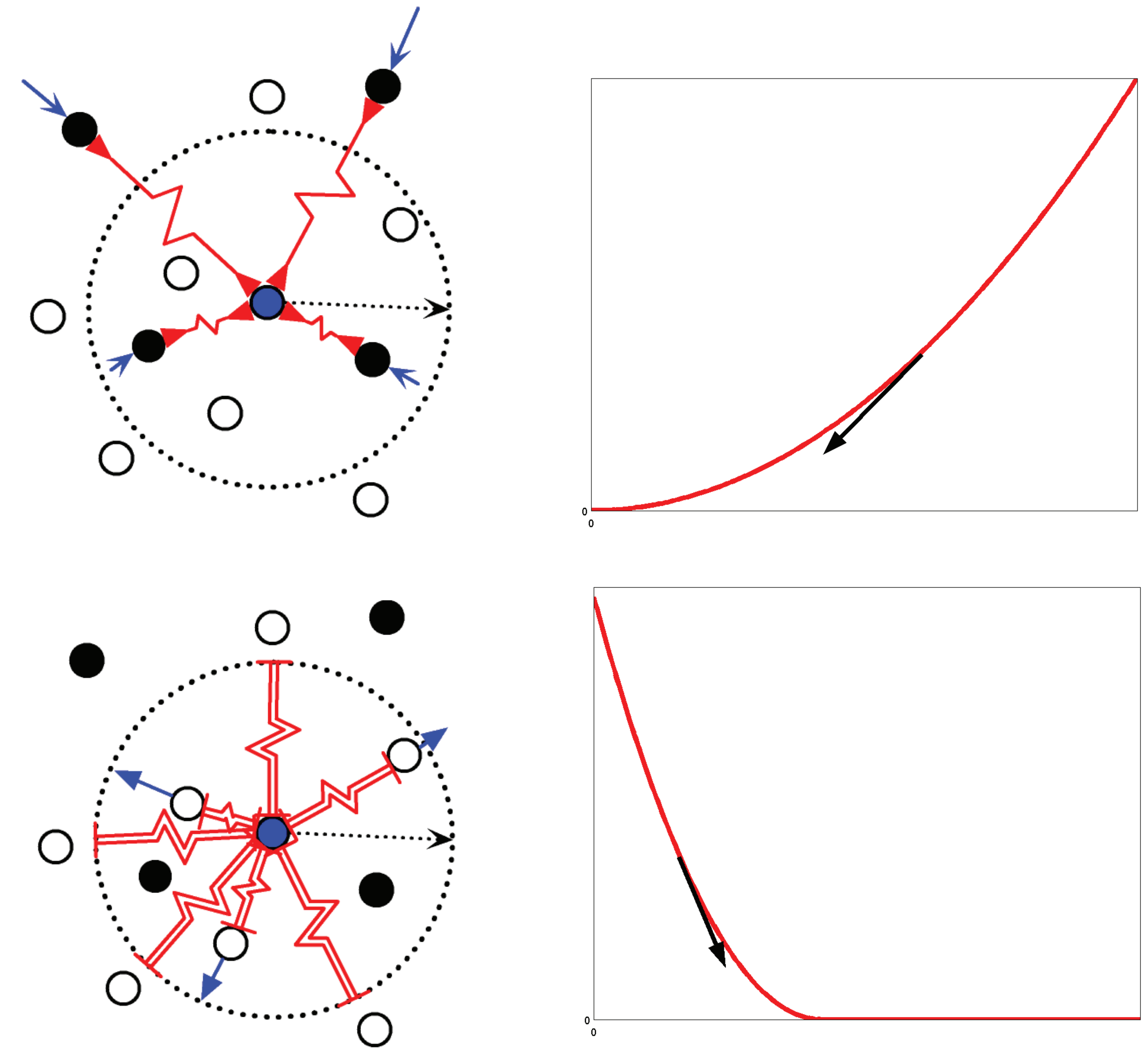


- The similarity loss “pushes together” similar points
- The dissimilarity loss “pulls apart” dissimilar points
    - but only if their distance is within some margin, α

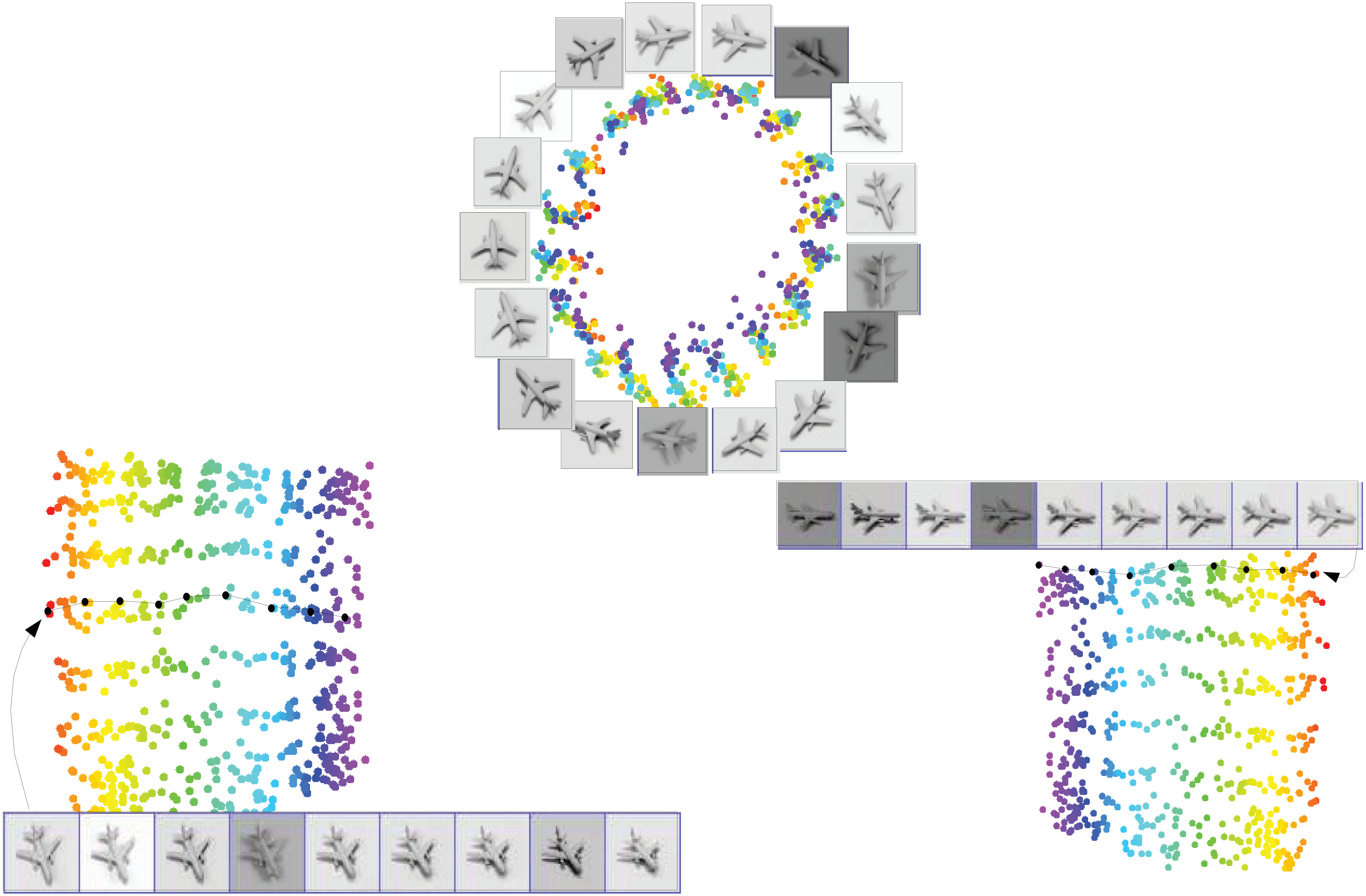
Hadsell, Chopra and LeCun 2006

Spring analogy

- Solid dots are points that are similar to the point in the centre
- Hollow dots are points that are dissimilar to the point in the centre
- Forces acting on the points are shown in blue
    - The length of the arrow represents the strength of the force
- Radius represents the margin, α



Figures from Hadsell et al.



Figures from Hadsell et al.

Triplet-based embedding

(Chechik et al. 2010)

Given a similarity score $S(\mathbf{x}_i, \mathbf{x}_j)$ for inputs $\mathbf{x}_i, \mathbf{x}_j$

We want to learn an embedding $f(\mathbf{x})$ such that

$$D\left(f\left(\mathbf{x}_i\right), f\left(\mathbf{x}_i^+\right)\right) < D\left(f\left(\mathbf{x}_i\right), f\left(\mathbf{x}_i^-\right)\right)$$

$$\underbrace{\forall \mathbf{x}_i, \mathbf{x}_i^+, \mathbf{x}_i^-}_{\text{“triplet”}} \quad \text{such that} \quad S(\mathbf{x}_i, \mathbf{x}_i^+) > S(\mathbf{x}_i, \mathbf{x}_i^-)$$

$D\left(f\left(\mathbf{x}_i\right), f\left(\mathbf{x}_j\right)\right)$ is a distance measure, commonly

$$D\left(f\left(\mathbf{x}_i\right), f\left(\mathbf{x}_j\right)\right) = ||f\left(\mathbf{x}_i\right) - f\left(\mathbf{x}_j\right)||^2$$

Learning fine-grained image similarity with deep ranking

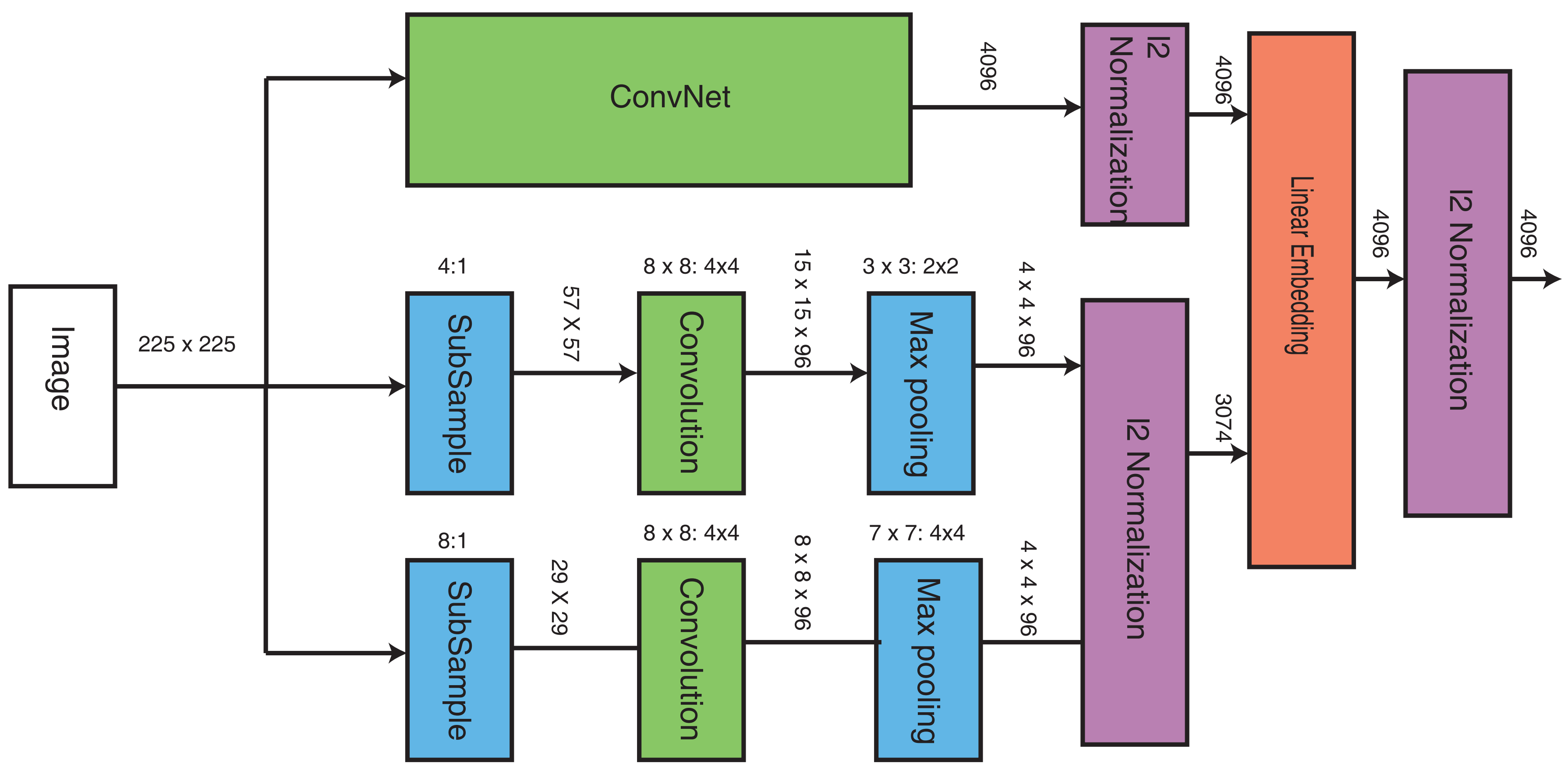
(Wang et al. 2014)

Objective:

$$\min \sum_i \xi_i + \lambda ||\theta||^2$$

$$\text{s.t.: } \max\left(0, g + D\left(f\left(\mathbf{x}_i\right), f\left(\mathbf{x}_i^+\right)\right) - D\left(f\left(\mathbf{x}_i\right), f\left(\mathbf{x}_i^-\right)\right)\right) \leq \xi_i$$

$$\forall \mathbf{x}_i, \mathbf{x}_i^+, \mathbf{x}_i^- \quad \text{s.t.} \quad S(\mathbf{x}_i, \mathbf{x}_i^+) > S(\mathbf{x}_i, \mathbf{x}_i^-)$$



ξ_i penalty

g gap (hyperparameter)

θ weights in network

λ regularization strength (hyperparameter)

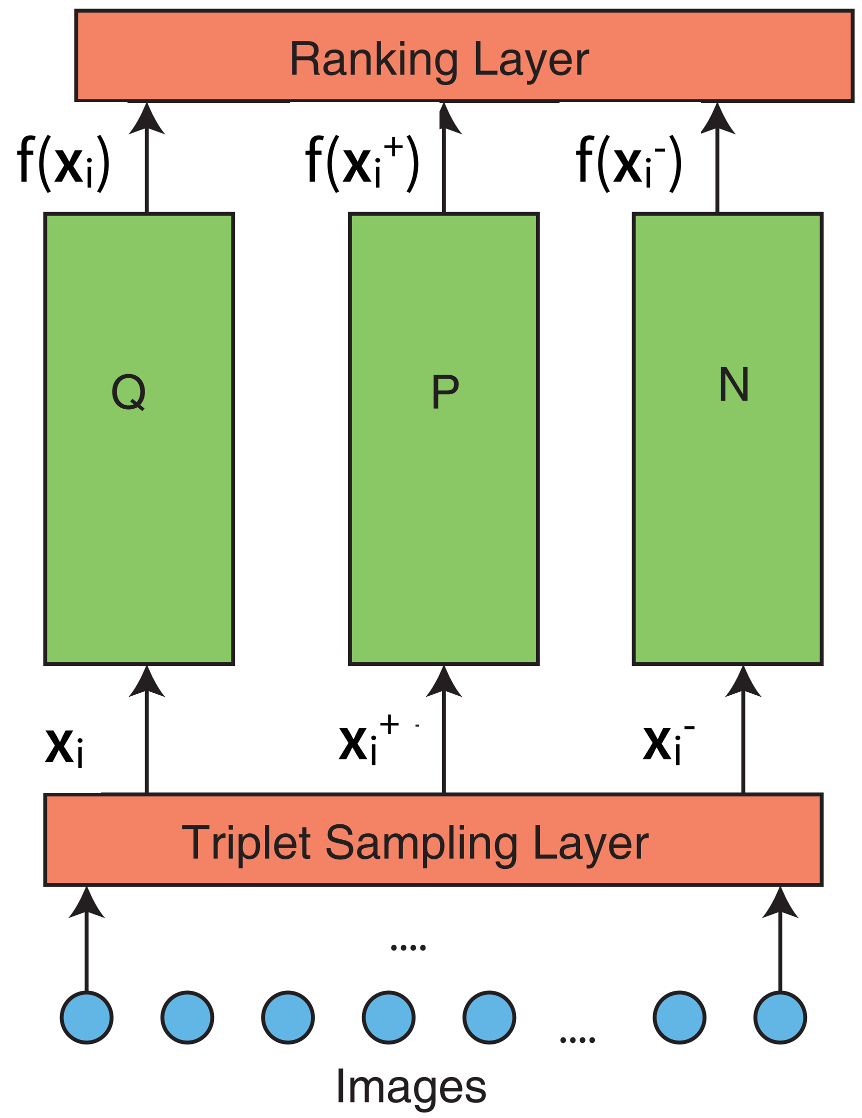
Figures from Wang et al. 2014

How to: triplet sampling

Figures from Wang et al. 2014

How to: triplet sampling

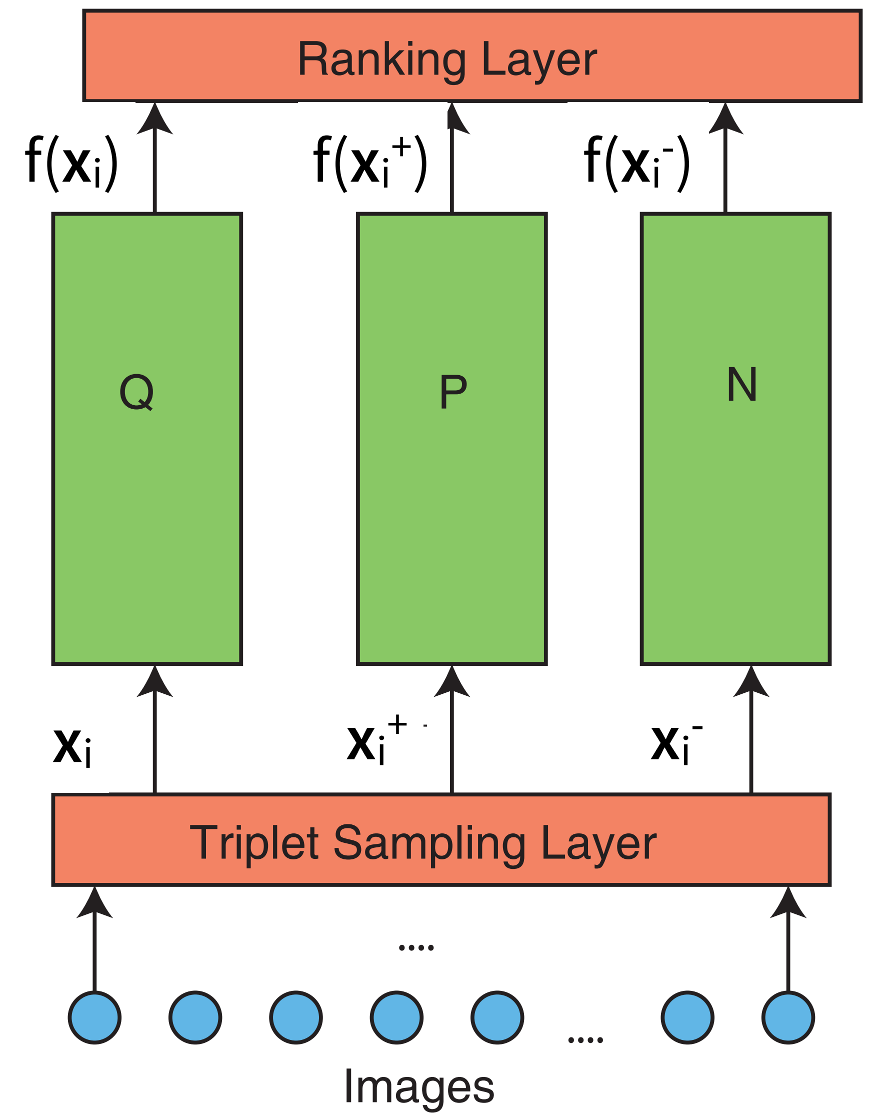
- # of possible triplets increases cubically with # of images
- e.g. 12M images, 1.728 x 10^21 triplets!
- Optimization converges in ~24M triplet samples
- Uniformly sampling triplets is sub-optimal

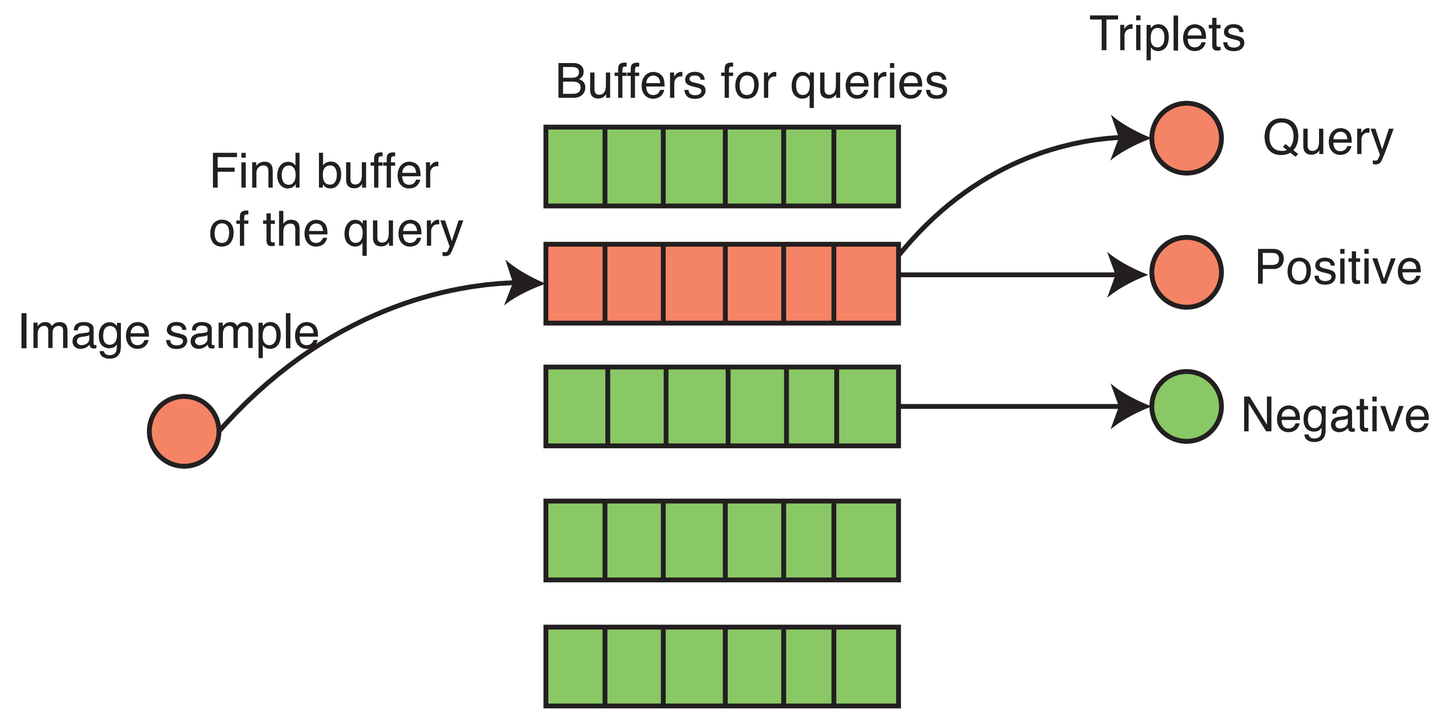


Figures from Wang et al. 2014

How to: triplet sampling

- # of possible triplets increases cubically with # of images
- e.g. 12M images, 1.728 x 10^21 triplets!
- Optimization converges in ~24M triplet samples
- Uniformly sampling triplets is sub-optimal
- Propose an online triplet sampling algorithm (more details in paper):
    - Sample an image according to its "relevance" to a category
    - Sample a positive image with high relevance
    - Sample "out-of-class" negatives uniformly
    - Sample "in-class" relevant negatives but ensure a margin between positive and negative examples





Figures from Wang et al. 2014

Finding similarity data

Finding similarity data

- NCA, DrLIM: binary notion of similarity typically defined by class membership or explicitly constructed neighbourhood graph

Finding similarity data

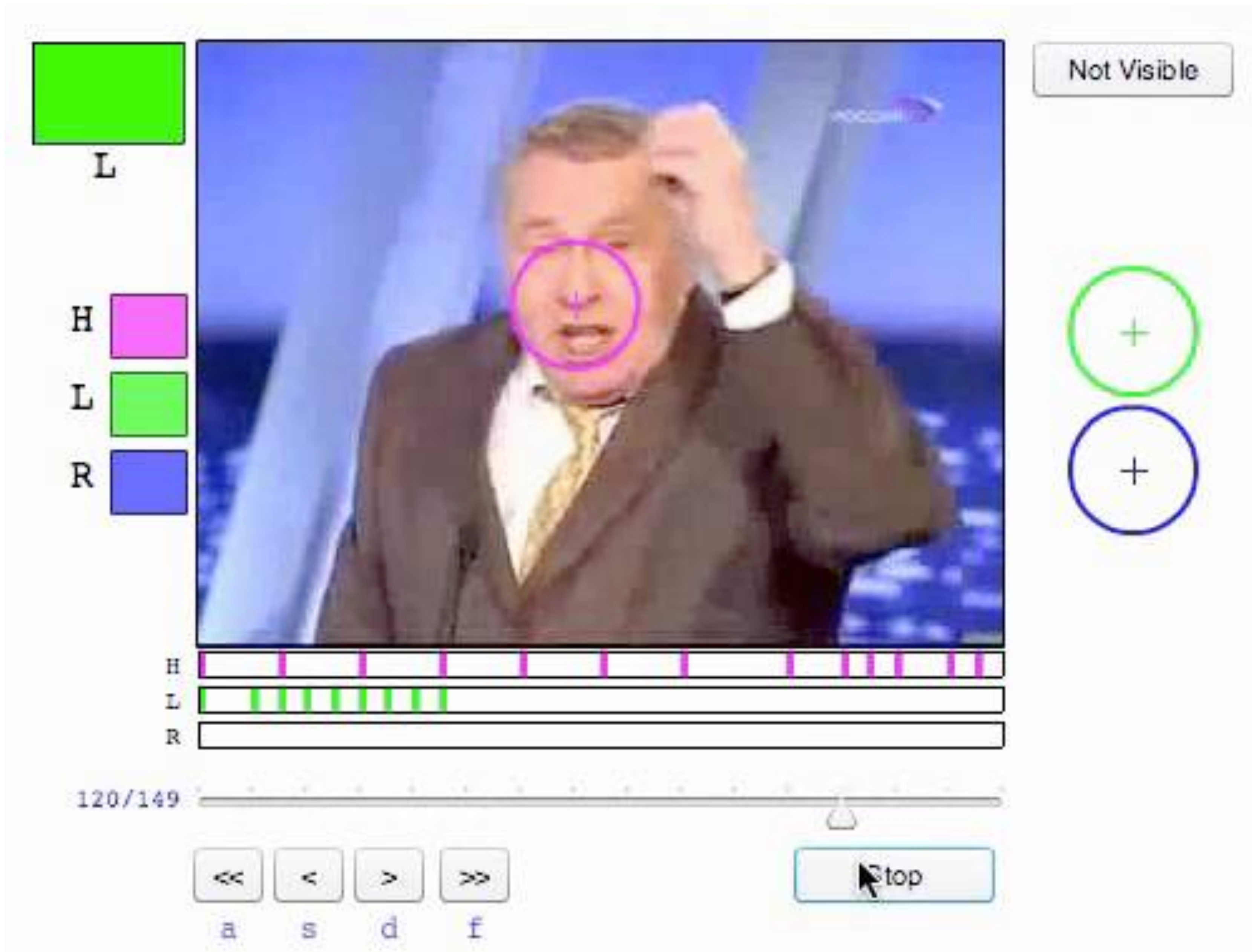
- NCA, DrLIM: binary notion of similarity typically defined by class membership or explicitly constructed neighbourhood graph
- Defining pairwise similarity is difficult and inconsistent across observers; Google used “Golden Feature” - weighted linear combination of 27 features

Finding similarity data

- NCA, DrLIM: binary notion of similarity typically defined by class membership or explicitly constructed neighbourhood graph
- Defining pairwise similarity is difficult and inconsistent across observers; Google used “Golden Feature” - weighted linear combination of 27 features
- Despite crowd-sourcing platforms (e.g. Amazon Mechanical Turk) gathering semantically similar pairs of images is expensive

Hands by hand

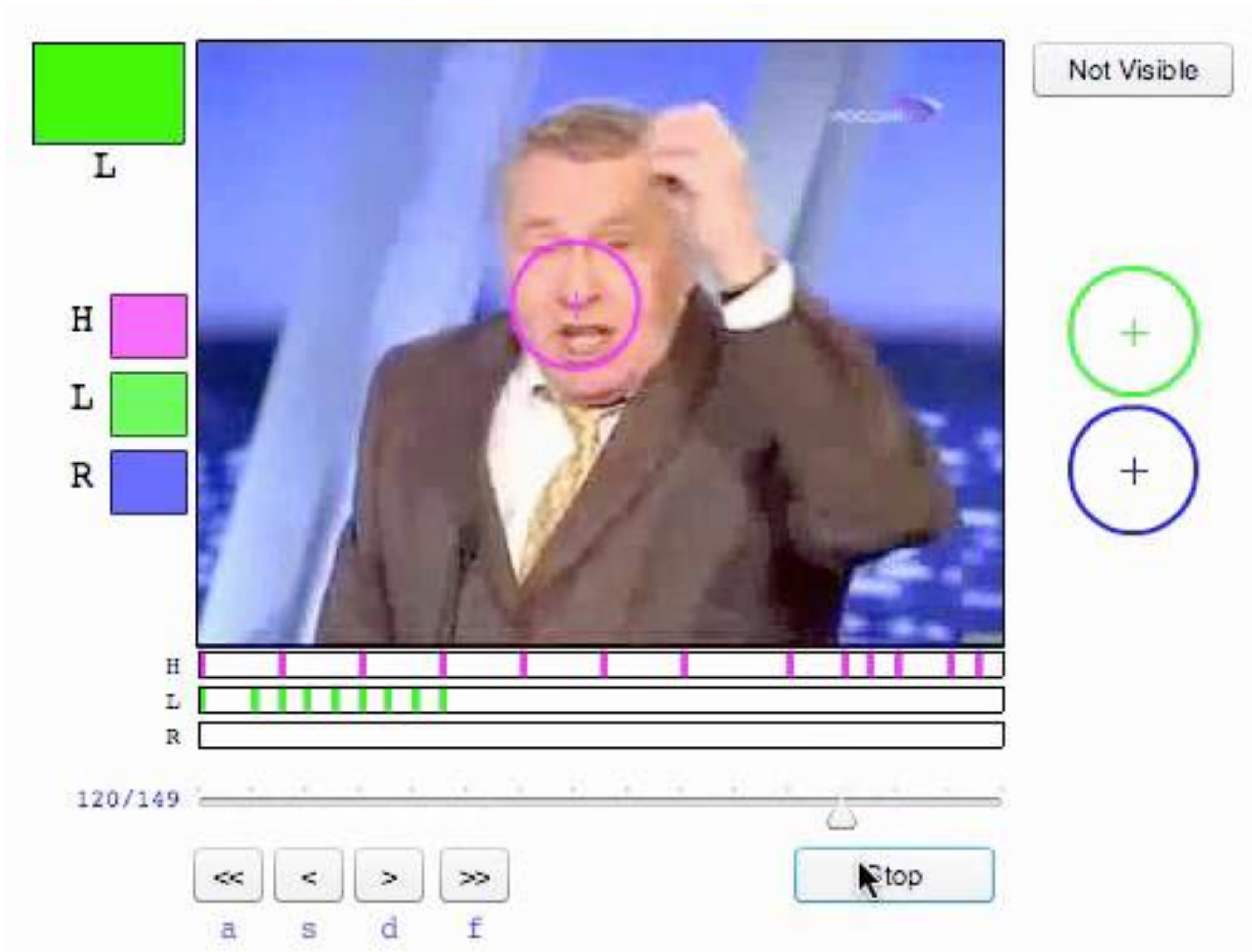
- One solution is to turn to synthetic data (e.g. Shakhnarovich et al. 2003, Jain et al. 2008)
- Difficult to generalize to real (e.g. “YouTube” settings)
- Another solution: ask people to label heads and hands (Spiro et al. 2010) or superimpose articulated skeletons (Bourdev et al. 2009)



(Spiro et al. 2010)

Hands by hand

- One solution is to turn to synthetic data (e.g. Shakhnarovich et al. 2003, Jain et al. 2008)
- Difficult to generalize to real (e.g. “YouTube” settings)
- Another solution: ask people to label heads and hands (Spiro et al. 2010) or superimpose articulated skeletons (Bourdev et al. 2009)



(Spiro et al. 2010)

Pose-sensitive embeddings

(Taylor et al. 2010)

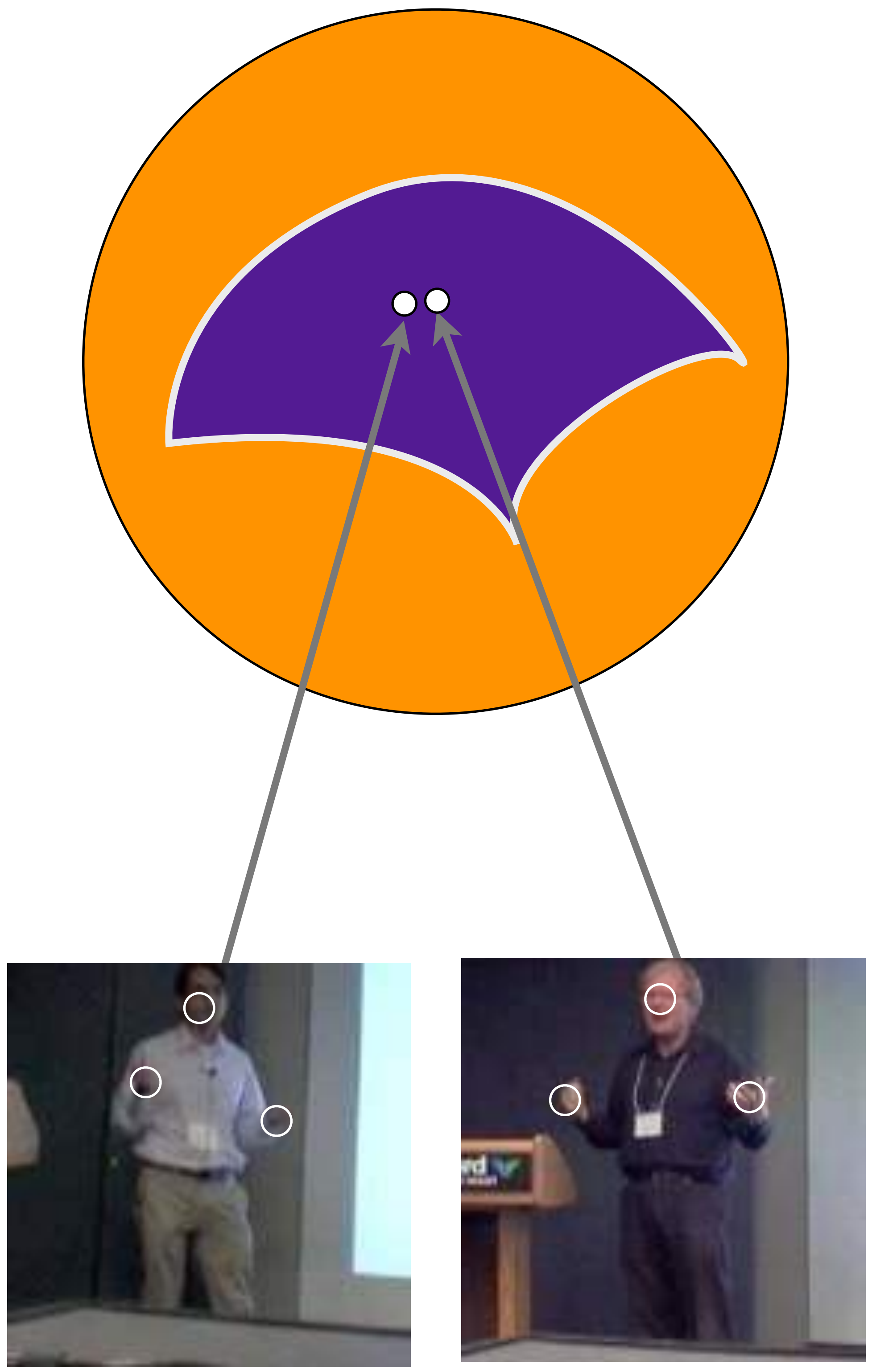
Pose-sensitive embeddings

(Taylor et al. 2010)

Database

Pose-sensitive embeddings

(Taylor et al. 2010)



Pose-sensitive embeddings

(Taylor et al. 2010)

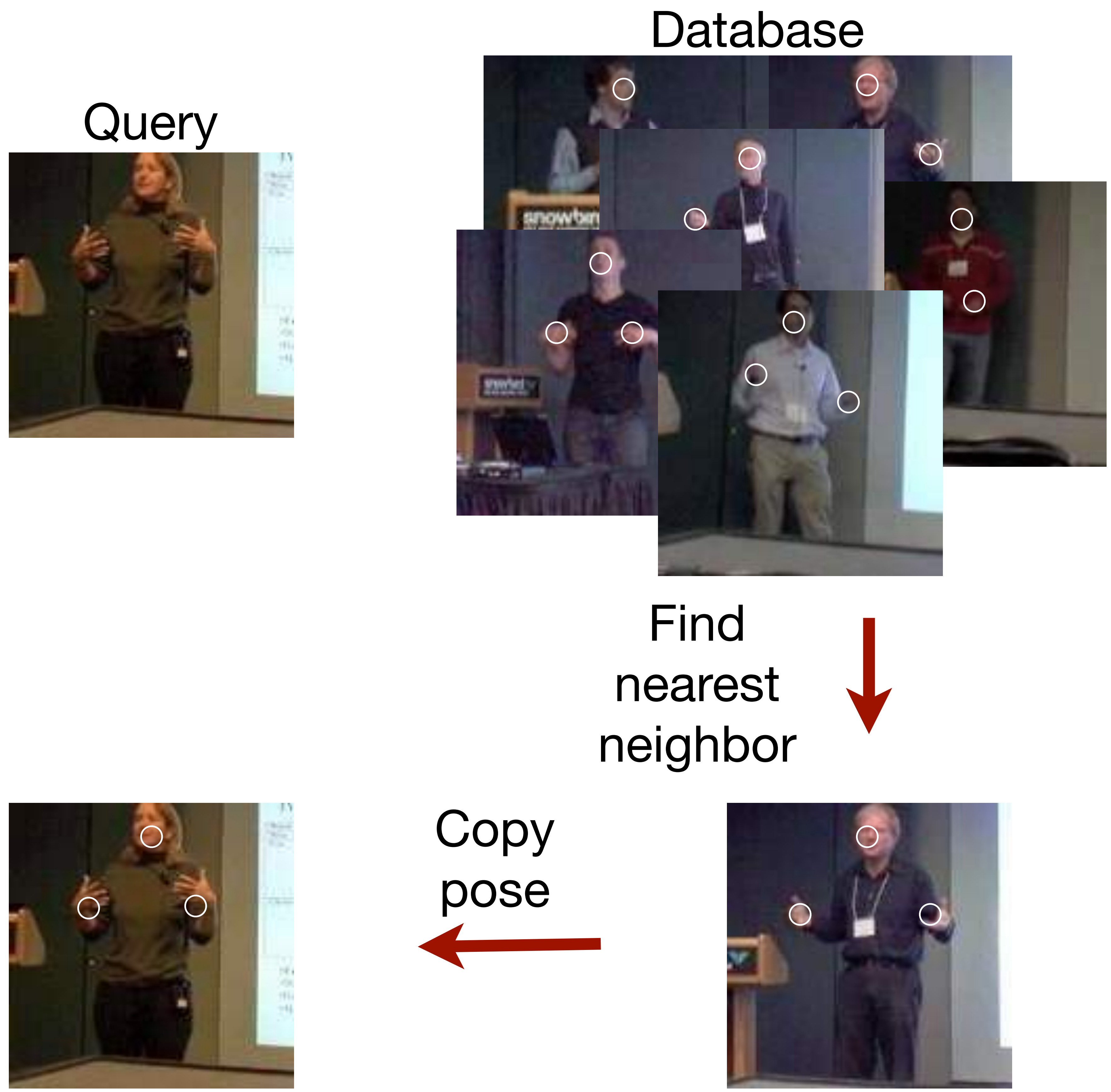
- If we have a database of images labeled with 2D or 3D pose information - we can do non-parametric pose estimation

Database

Pose-sensitive embeddings

(Taylor et al. 2010)

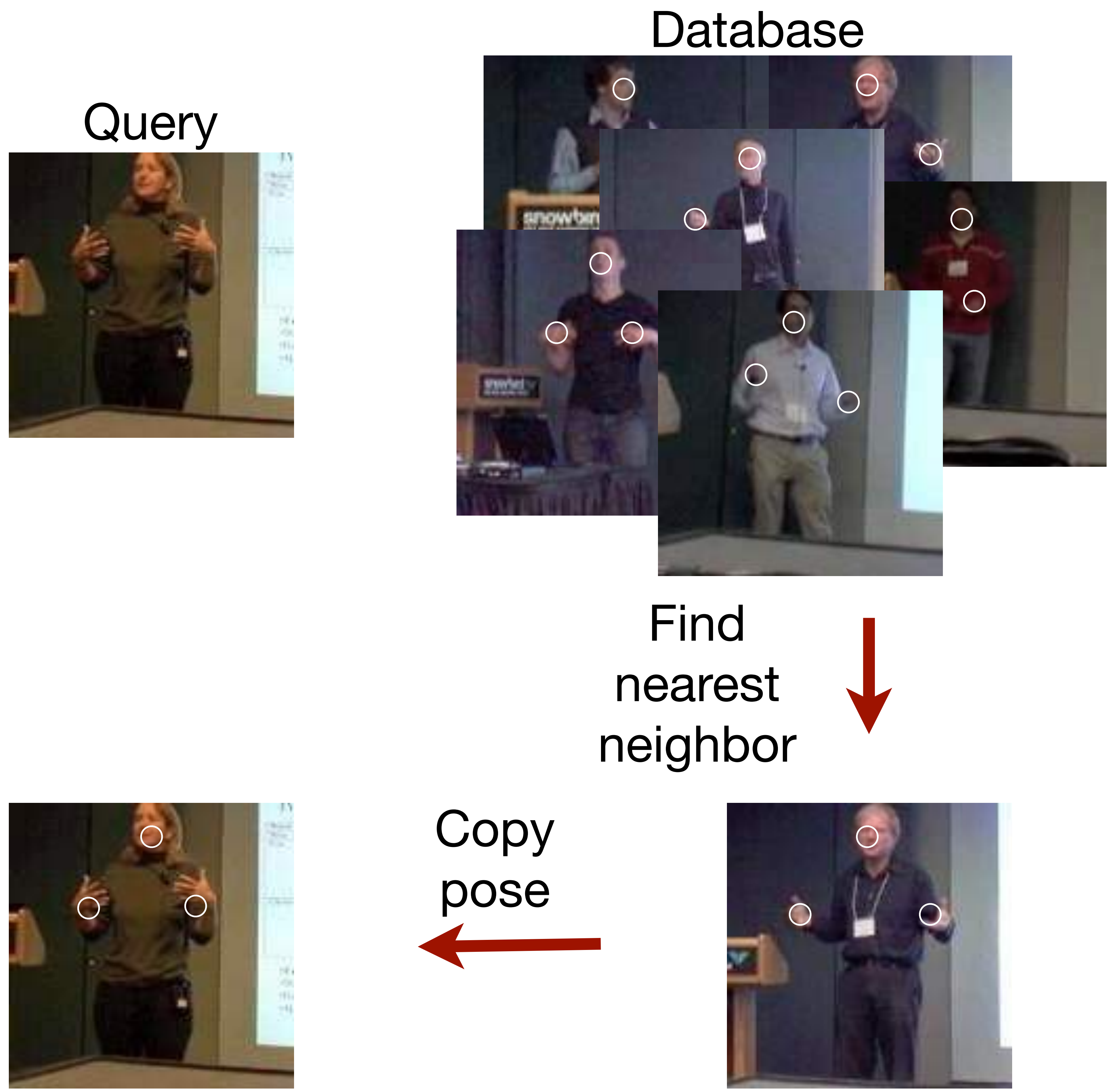
- If we have a database of images labeled with 2D or 3D pose information - we can do non-parametric pose estimation



Pose-sensitive embeddings

(Taylor et al. 2010)

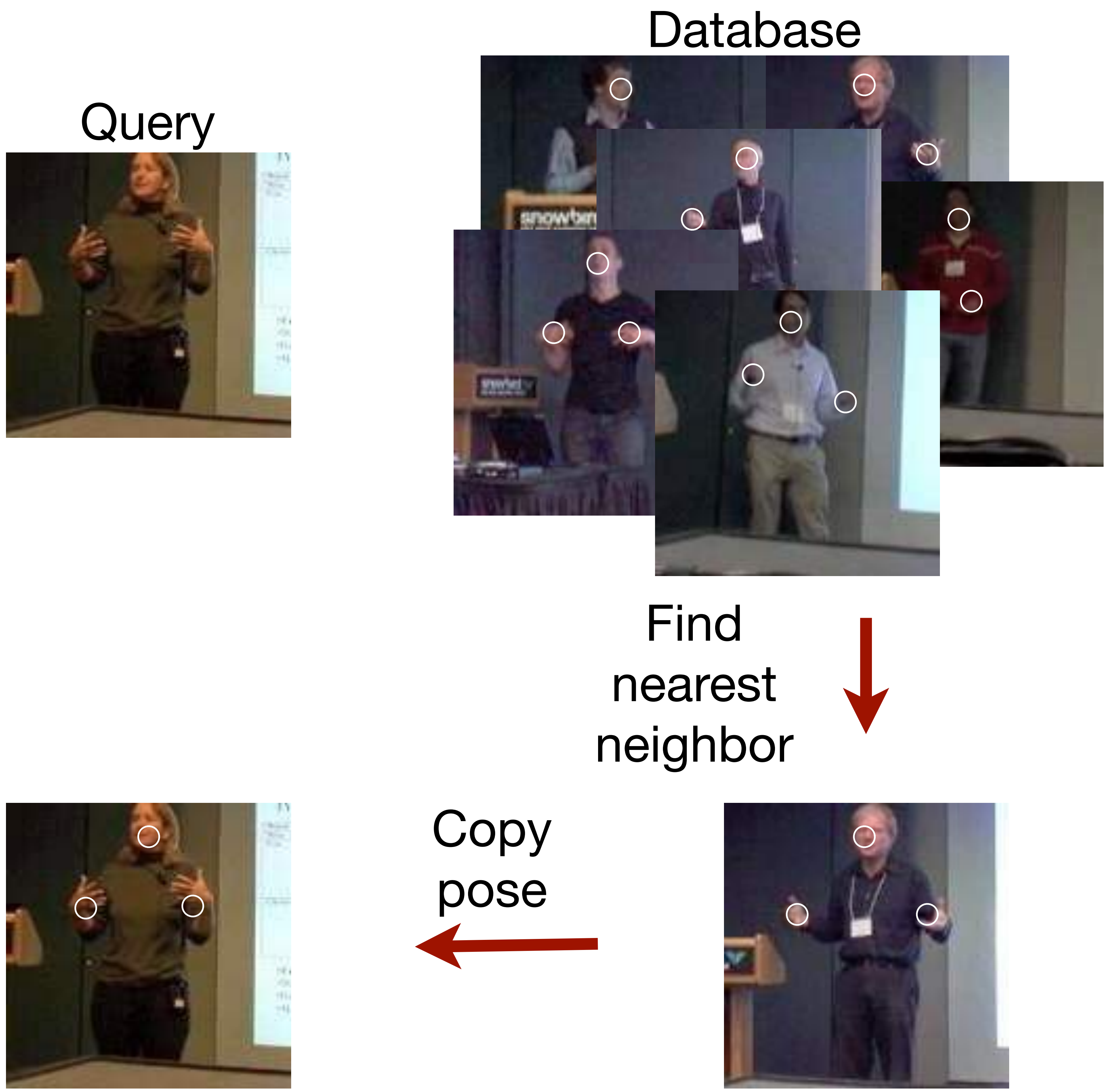
- If we have a database of images labeled with 2D or 3D pose information - we can do non-parametric pose estimation
- Nearest neighbor lookup must be quick (e.g. performed in a low-dimensional space)



Pose-sensitive embeddings

(Taylor et al. 2010)

- If we have a database of images labeled with 2D or 3D pose information - we can do non-parametric pose estimation
- Nearest neighbor lookup must be quick (e.g. performed in a low-dimensional space)
- It also must be informative of pose and invariant to clothing, lighting, scale, and other appearance changes



NCA regression

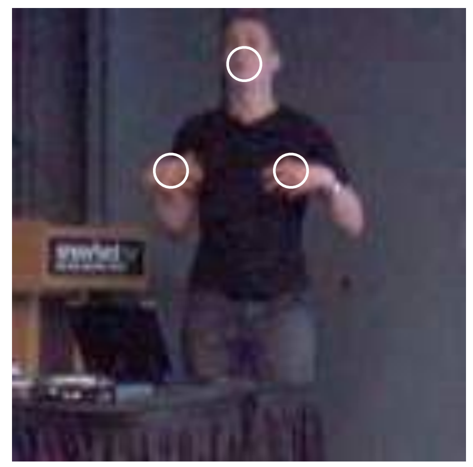
$$L_{\mathrm{NCAR}} = \sum_{i=1}^{N} \sum_{j} p_{ij} ||\mathbf{y}_i - \mathbf{y}_j||_2^2$$

Minimize loss w.r.t.

Pay a high cost for “neighbours” in feature space that are far away in pose space

$\mathbf{x}_i$

$$\mathbf{y}_i = [48.2, 46.3, \ldots, 63.3]^T$$



$\mathbf{x}_j$

$$\mathbf{y}_i = [54.4, 45.8, \ldots, 64.1]^T$$

Snowbird dataset

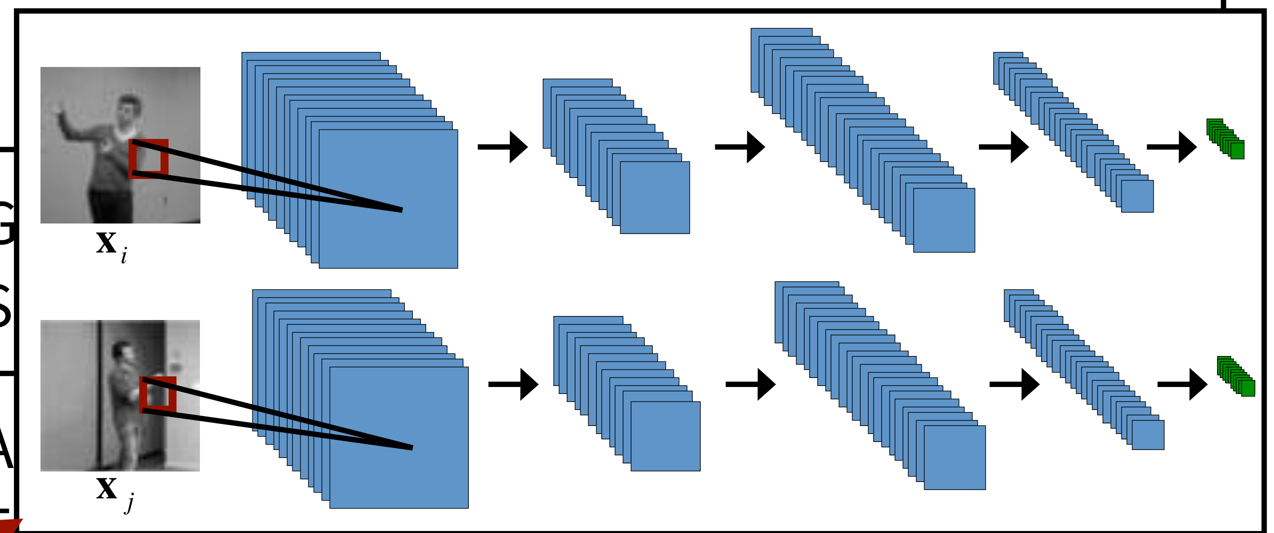
- We digitally recorded all contributing and invited speakers at the 2010 Snowbird workshop
- After each session of talks, blocks of 150 frames were distributed as Human Intelligence Tasks (HITs) on Amazon Mechanical Turk

Comparison of Approaches

Pixel distance	Not practical
GIST	•Global representation of image •Still not practical
Linear NCA regression (NCAR)	•Applied to pre-computed GIST •Fit by conjugate gradient
Convolutional NCAR (C-NCAR)	•Convolutions applied to pixels •Tanh(),Abs(),Average downsampling
DrLIM Regression (DrLIMR)	•Similar to NCAR but adds an explicit contrastive loss
Convolutional DrLIMR (C-DrLIMR)	•Similar to C-NCAR but adds an explicit contrastive loss

Comparison of Approaches

Pixel distance	
GIST	•G •S
Linear NCA regression (NCAR)	•A •Fit by conjugate gradient
Convolutional NCAR (C-NCAR)	•Convolutions applied to pixels •Tanh(),Abs(),Average downsampling
DrLIM Regression (DrLIMR)	•Similar to NCAR but adds an explicit contrastive loss
Convolutional DrLIMR (C-DrLIMR)	•Similar to C-NCAR but adds an explicit contrastive loss



Comparison of Approaches

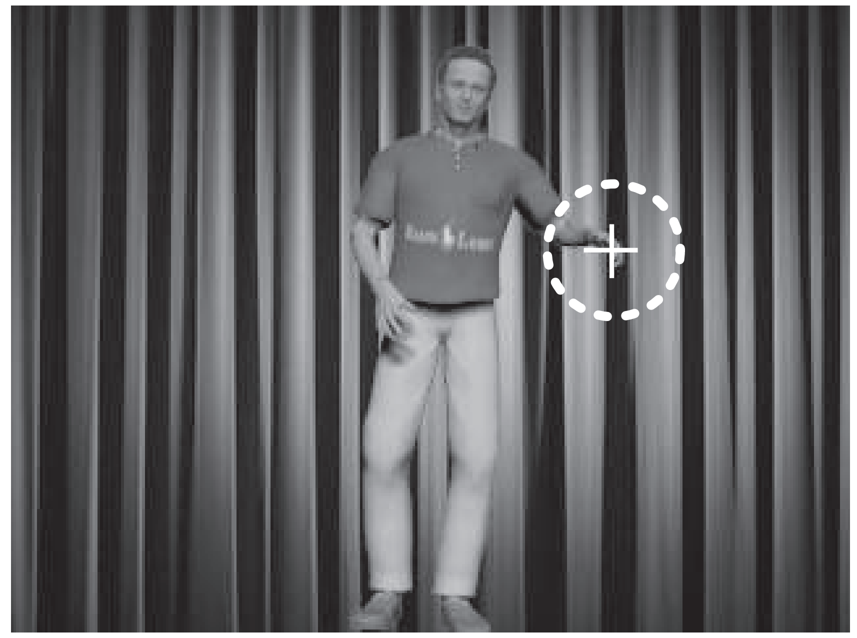
Pixel distance	Not practical
GIST	•Global representation of image •Still not practical
Linear NCA regression (NCAR)	•Applied to pre-computed GIST •Fit by conjugate gradient
Convolutional NCAR (C-NCAR)	•Convolutions applied to pixels •Tanh(),Abs(),Average downsampling
DrLIM Regression (DrLIMR)	•Similar to NCAR but adds an explicit contrastive loss
Convolutional DrLIMR (C-DrLIMR)	•Similar to C-NCAR but adds an explicit contrastive loss

Results (qualitative)

- Both Pixel-based matching and GIST focus on scene content, lighting
- Our method learns invariance to background, focuses on pose
- Though trained on hands relative to head, seems to capture something more substantial about body pose

Results (quantitative)

Embedding	Input	Code size	Err-SY	Err-RE
None	Pixels	16384	32.86	25.12
None	GIST	512	47.41	25.3
PCA	GIST	128	47.17	24.85
PCA	GIST	32	48.99	25.74
NCAR	GIST	32	34.21	24.93
NCAR	LCN+GIST	32	32.9	23.15
S-DrLIM	GIST	32	37.8	25.19
Boost-SSC	LCN+GIST	32	34.8	22.65
C-NCAR	LCN	32	28.95	**16.41**
C-DRLIM	LCN	32	**25.4**	19.61



25.4 pixel error

16.4 pixel error

MPII Human Pose

(Andriluka et al. 2014)

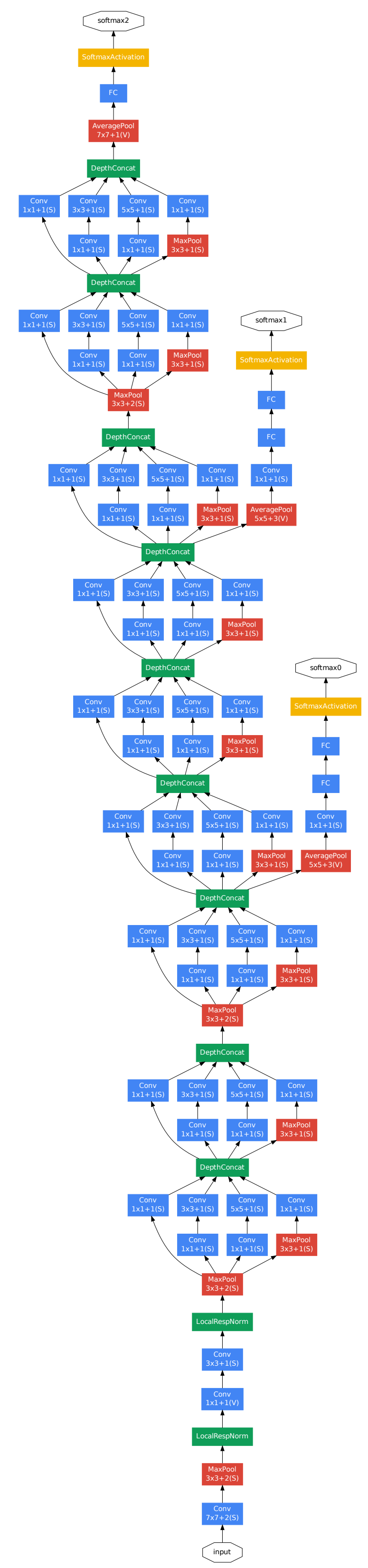
- Addresses appearance variability and complexity
- YouTube as a data source
- Many activities, indoor and outdoor scenes, variety of imaging conditions

Dataset	#training	#test	img. type
Full body pose datasets			
Parse [16]	100	205	diverse
LSP [12]	1,000	1,000	sports (8 types)
PASCAL Person Layout [6]	850	849	everyday
Sport [21]	649	650	sports
UIUC people [21]	346	247	sports (2 types)
LSP extended [13]	10,000	-	sports (3 types)
FashionPose [2]	6,530	775	fashion blogs
J-HMDB [11]	31,838	-	diverse (21 act.)
Upper body pose datasets			
Buffy Stickmen [8]	472	276	TV show (Buffy)
ETHZ PASCAL Stickmen [3]	-	549	PASCAL VOC
Human Obj. Int. (HOI) [23]	180	120	sports (6 types)
We Are Family [5]	350 imgs.	175 imgs.	group photos
Video Pose 2 [18]	766	519	TV show (Friends)
FLIC [17]	6,543	1,016	feature movies
Sync. Activities [4]	-	357 imgs.	dance / aerobics
Armlets [9]	9,593	2,996	PASCAL VOC/Flickr
MPII Human Pose (this paper)	**28,821**	**11,701**	diverse (491 act.)

Pose embeddings

(Mori et al. 2015)

- Similar to (Taylor et al. 2010), but uses:
  - MPII database: 2D locations of 16 body joints
  - Triplet-style learning
  - Modern, “Inception”-style convnet



Can we avoid explicit labeling of body parts?

Weakly-supervised embeddings

(Taylor et al. 2011)

Weakly-supervised embeddings

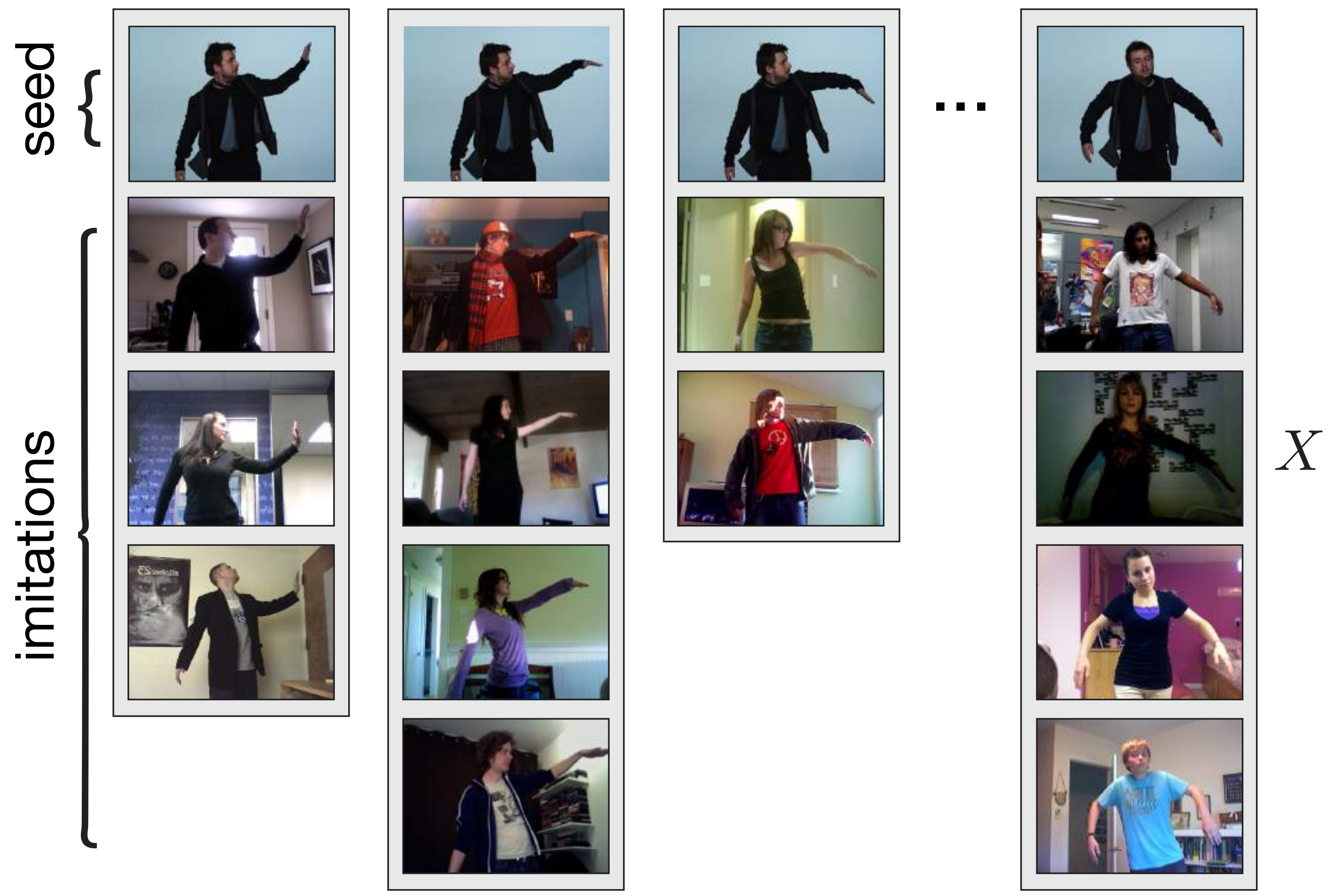
(Taylor et al. 2011)

- Have people imitate frames from a video:
  - imitated frames, though different in appearance, should be embedded nearby

Weakly-supervised embeddings

(Taylor et al. 2011)

- Have people imitate frames from a video:
  - imitated frames, though different in appearance, should be embedded nearby



Weakly-supervised embeddings

(Taylor et al. 2011)

- Have people imitate frames from a video:
  - imitated frames, though different in appearance, should be embedded nearby
- Use *temporal coherence* as a similarity signal:
  - i.e. frames which are close together in time should be embedded nearby

Zero-shot learning (Nourouzi et al. 2014)

Test Image	Softmax Baseline [7]	DeViSE [6]	ConSE (10)
	wig fur coat Saluki, gazelle hound Afghan hound, Afghan stole	water spaniel tea gown bridal gown, wedding gown spaniel tights, leotards	business suit **dress, frock** hairpiece, false hair, postiche swimsuit, swimwear, bathing suit kit, outfit
	ostrich, Struthio camelus black stork, Ciconia nigra vulture crane peacock	heron owl, bird of Minerva, bird of night hawk bird of prey, raptor, raptorial bird finch	**ratite, ratite bird, flightless bird** peafowl, bird of Juno common spoonbill New World vulture, cathartid Greek partridge, rock partridge
	sea lion plane, carpenter's plane cowboy boot loggerhead, loggerhead turtle goose	elephant turtle turtleneck, turtle, polo-neck flip-flop, thong handcart, pushcart, cart, go-cart	California sea lion **Steller sea lion** Australian sea lion South American sea lion eared seal
	hamster broccoli Pomeranian capuchin, ringtail weasel	**golden hamster, Syrian hamster** rhesus, rhesus monkey pipe shaker American mink, Mustela vison	**golden hamster, Syrian hamster** rodent, gnawer Eurasian hamster rhesus, rhesus monkey rabbit, coney, cony
 **(farm machine)**	thresher, threshing machine tractor harvester, reaper half track snowplow, snowplough	truck, motortruck skidder tank car, tank automatic rifle, machine rifle trailer, house trailer	flatcar, flatbed, flat truck, motortruck tracked vehicle bulldozer, dozer wheeled vehicle
 **(alpaca, Lama pacos)**	Tibetan mastiff titi, titi monkey koala, koala bear, kangaroo bear llama chow, chow chow	kernel littoral, litoral, littoral zone, sands carillon Cabernet, Cabernet Sauvignon poodle, poodle dog	dog, domestic dog domestic cat, house cat schnauzer Belgian sheepdog domestic llama, Lama peruana

Zero-shot learning (Nourouzi et al. 2014)

- Can you exploit a trained word embedding model (Mikolov et al. 2013) and a trained object recognition model (Krizhevsky et al. 2012) to label images from unseen classes?

Test Image	Softmax Baseline [7]	DeViSE [6]	ConSE (10)
	wig fur coat Saluki, gazelle hound Afghan hound, Afghan stole	water spaniel tea gown bridal gown, wedding gown spaniel tights, leotards	business suit **dress, frock** hairpiece, false hair, postiche swimsuit, swimwear, bathing suit kit, outfit
	ostrich, Struthio camelus black stork, Ciconia nigra vulture crane peacock	heron owl, bird of Minerva, bird of night hawk bird of prey, raptor, raptorial bird finch	**ratite, ratite bird, flightless bird** peafowl, bird of Juno common spoonbill New World vulture, cathartid Greek partridge, rock partridge
	sea lion plane, carpenter's plane cowboy boot loggerhead, loggerhead turtle goose	elephant turtle turtleneck, turtle, polo-neck flip-flop, thong handcart, pushcart, cart, go-cart	California sea lion **Steller sea lion** Australian sea lion South American sea lion eared seal
	hamster broccoli Pomeranian capuchin, ringtail weasel	**golden hamster, Syrian hamster** rhesus, rhesus monkey pipe shaker American mink, Mustela vison	**golden hamster, Syrian hamster** rodent, gnawer Eurasian hamster rhesus, rhesus monkey rabbit, coney, cony
 **(farm machine)**	thresher, threshing machine tractor harvester, reaper half track snowplow, snowplough	truck, motortruck skidder tank car, tank automatic rifle, machine rifle trailer, house trailer	flatcar, flatbed, flat truck, motortruck tracked vehicle bulldozer, dozer wheeled vehicle
 **(alpaca, Lama pacos)**	Tibetan mastiff titi, titi monkey koala, koala bear, kangaroo bear llama chow, chow chow	kernel littoral, litoral, littoral zone, sands carillon Cabernet, Cabernet Sauvignon poodle, poodle dog	dog, domestic dog domestic cat, house cat schnauzer Belgian sheepdog domestic llama, Lama peruana

Zero-shot learning (Nourouzi et al. 2014)

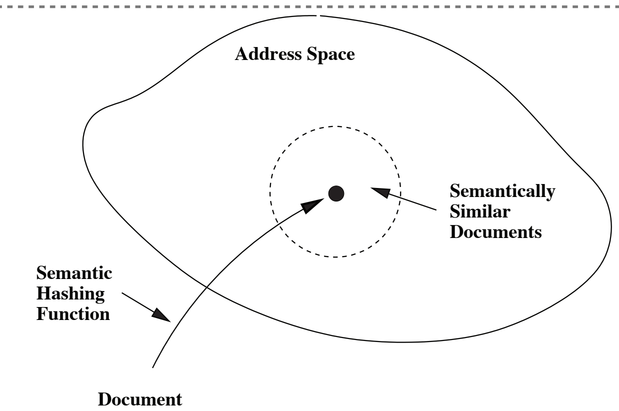
- Can you exploit a trained word embedding model (Mikolov et al. 2013) and a trained object recognition model (Krizhevsky et al. 2012) to label images from unseen classes?
- Let softmax output of recognition model for top T classes determine convex combination of semantic word embeddings

Test Image	Softmax Baseline [7]	DeViSE [6]	ConSE (10)
	wig fur coat Saluki, gazelle hound Afghan hound, Afghan stole	water spaniel tea gown bridal gown, wedding gown spaniel tights, leotards	business suit **dress, frock** hairpiece, false hair, postiche swimsuit, swimwear, bathing suit kit, outfit
	ostrich, Struthio camelus black stork, Ciconia nigra vulture crane peacock	heron owl, bird of Minerva, bird of night hawk bird of prey, raptor, raptorial bird finch	**ratite, ratite bird, flightless bird** peafowl, bird of Juno common spoonbill New World vulture, cathartid Greek partridge, rock partridge
	sea lion plane, carpenter's plane cowboy boot loggerhead, loggerhead turtle goose	elephant turtle turtleneck, turtle, polo-neck flip-flop, thong handcart, pushcart, cart, go-cart	California sea lion **Steller sea lion** Australian sea lion South American sea lion eared seal
	hamster broccoli Pomeranian capuchin, ringtail weasel	**golden hamster, Syrian hamster** rhesus, rhesus monkey pipe shaker American mink, Mustela vison	**golden hamster, Syrian hamster** rodent, gnawer Eurasian hamster rhesus, rhesus monkey rabbit, coney, cony
 **(farm machine)**	thresher, threshing machine tractor harvester, reaper half track snowplow, snowplough	truck, motortruck skidder tank car, tank automatic rifle, machine rifle trailer, house trailer	flatcar, flatbed, flat truck, motortruck tracked vehicle bulldozer, dozer wheeled vehicle
 **(alpaca, Lama pacos)**	Tibetan mastiff titi, titi monkey koala, koala bear, kangaroo bear llama chow, chow chow	kernel littoral, litoral, littoral zone, sands carillon Cabernet, Cabernet Sauvignon poodle, poodle dog	dog, domestic dog domestic cat, house cat schnauzer Belgian sheepdog domestic llama, Lama peruana

Summary

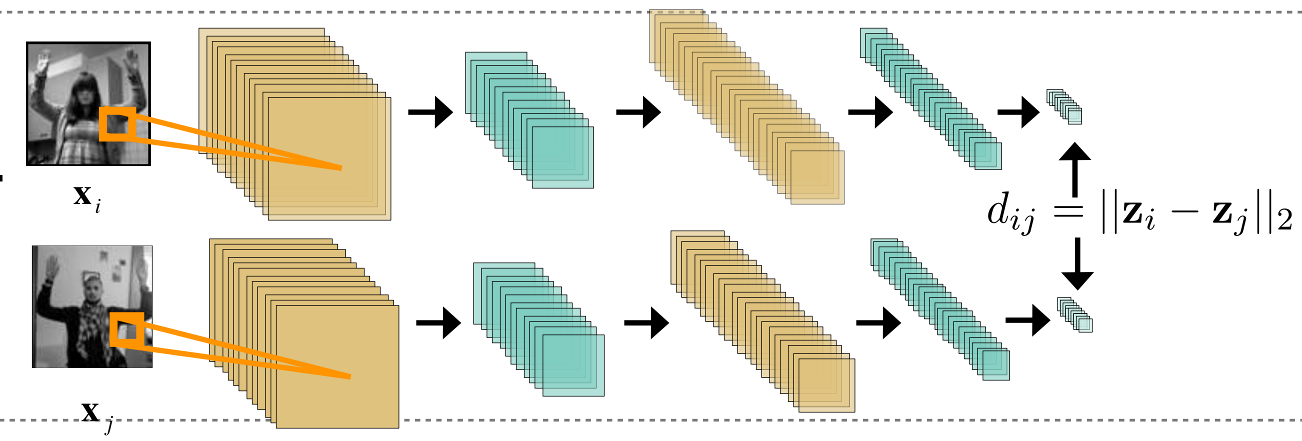
Unsupervised

Learn similarity structure completely from unlabeled data.
Difficult to ensure that similar examples map to similar codes.



Supervised

Use labels or neighbourhood graph to inform map.
Often, this information is not available!

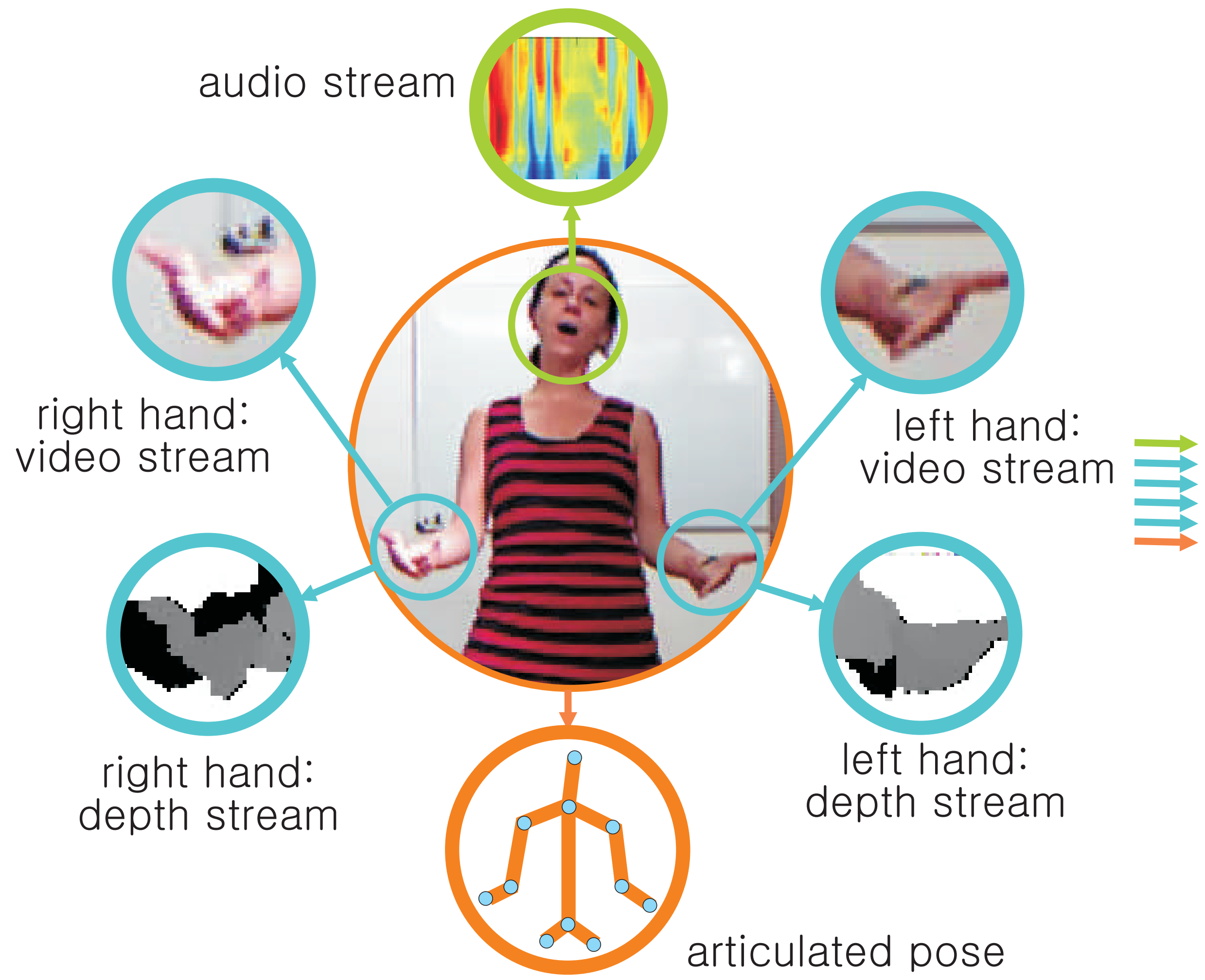


Weakly supervised

Use of temporal coherence to guide learning.
Application to zero-shot learning.

Where to go from here?

- Architectural improvements, (e.g. going deeper, more efficient use of parameters, multi-scale pathways, etc.), will continue to make impact
- Databases will only continue to grow, so efficiency of search (e.g. Hashing) will be important
- Approaches will roll out to domains beyond images, audio and text



Multi-modal learning (next talk)

Image: Neverova et al. (2015)

Thank You!

Montreal
Guelph
Toronto
New York
Quebec City
Presque Isle
Sudbury
North Bay
Trois-Rivières
Victoriaville
Drummondville
Algonquin Provincial Park
Ottawa
Granby
Sherbrooke
Saint-Jean-sur-Richelieu
Maine
Bangor
Kingston
Vermont
Markham
Mississauga
Portland
Rochester
New Hampshire
Hamilton
London
Buffalo
New York
Albany
Manchester
Chatham
Massachusetts
Boston
Springfield
Providence
Cleveland
Connecticut
Akron
New Haven
Bridgeport
Canton
Pittsburgh
Pennsylvania
Allentown
Rhode Island
Altoona
Trenton
Columbus
Philadelphia
Toms River