Embeddings and Uncertainty

Oct. 4, 2019

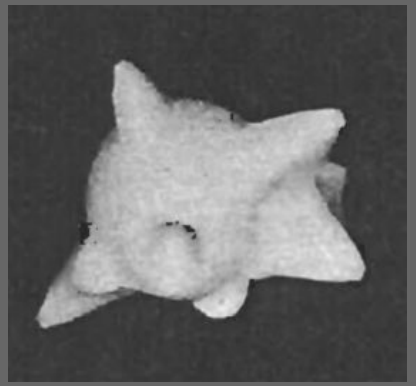
Andrew Gallagher
agallagher@google.com

Seong Joon Oh, Joseph Roth, Jiyan Pan, Florian Schroff, Kevin Murphy, Mike Mozer, Caroline Pantofaru, Michael Nechyba, Cusuh Ham, Zhou Zhou, Abinash Behera

longing for certainty … is in every human mind. But certainty generally is illusion.

Oliver Wendell Holmes

Would you recognize this person if you needed to?



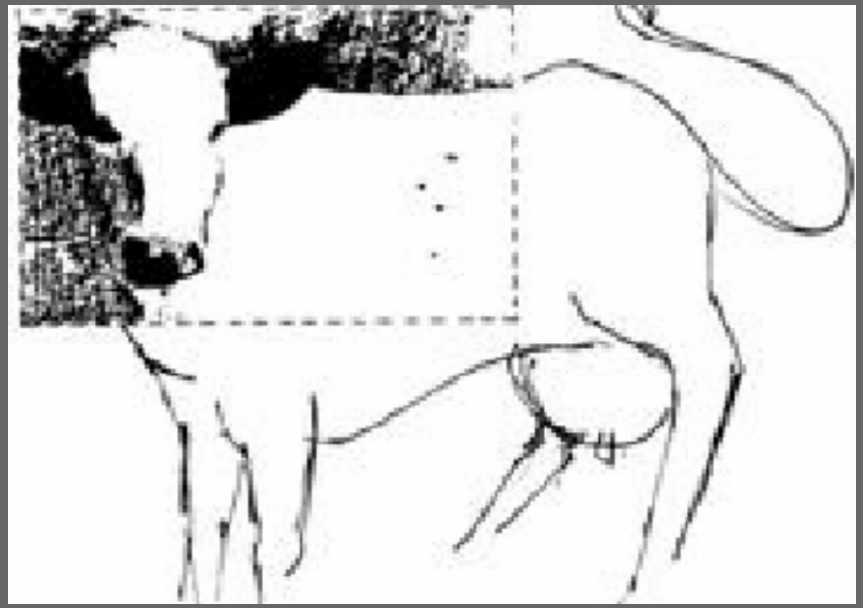
What about one of these?

Bulthoff and Edelman, Psychophysical support for a two-dimensional view interpolation theory of object recognition, 1992. (link)

How about this?

Bulthoff, Object Recognition in Man and Machine (link)

How about this?



Bulthoff, Object Recognition in Man and Machine (link)

Introduction

Embeddings

Uncertainty Representations (2)

Discussion

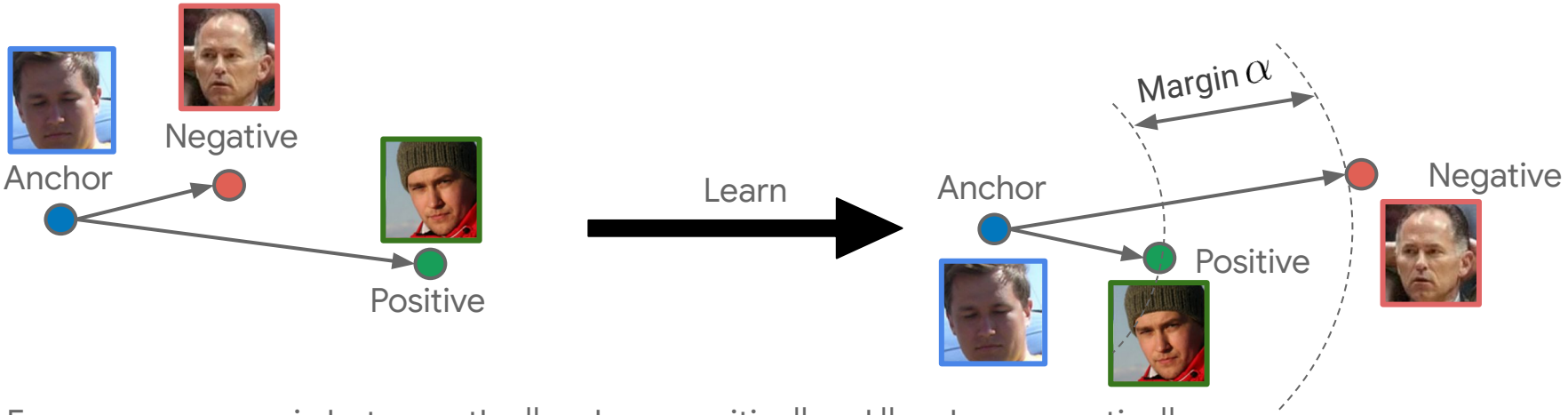
Introduction

Embeddings

Uncertainty Representations (2)

Discussion

FaceNet

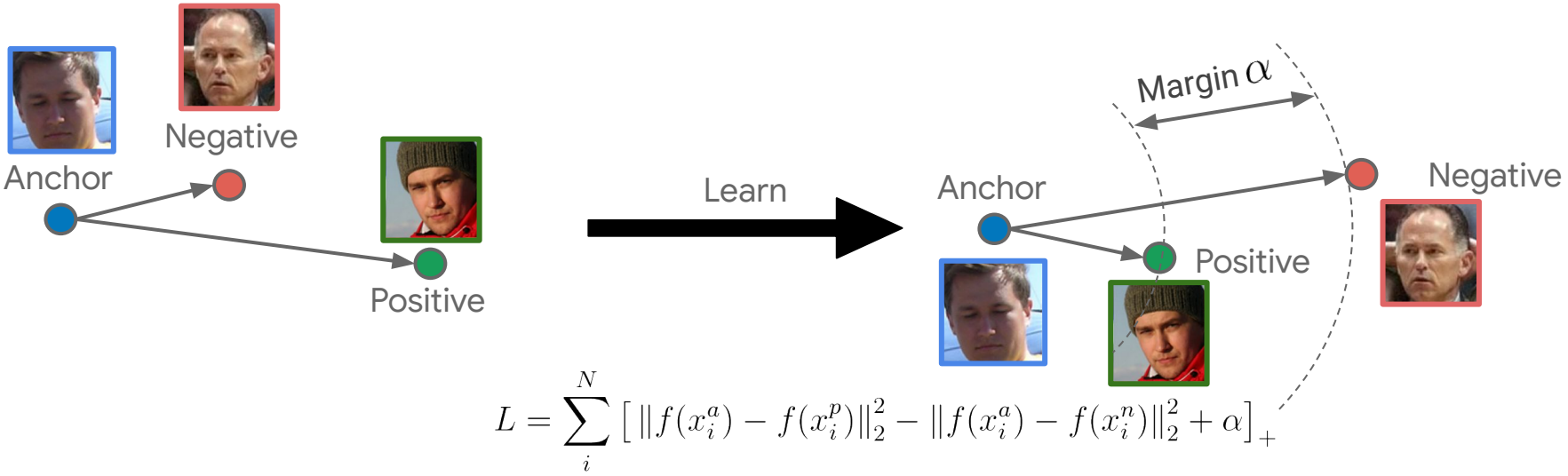


Encourages a margin between the ||anchor - positive|| and ||anchor - negative||.

F. Schroff, D. Kalenichenko, J. Philbin, FaceNet: A Unified Embedding for Face Recognition and Clustering
https://arxiv.org/abs/1503.03832

slide content: F. Schroff and D. Kalenichenko

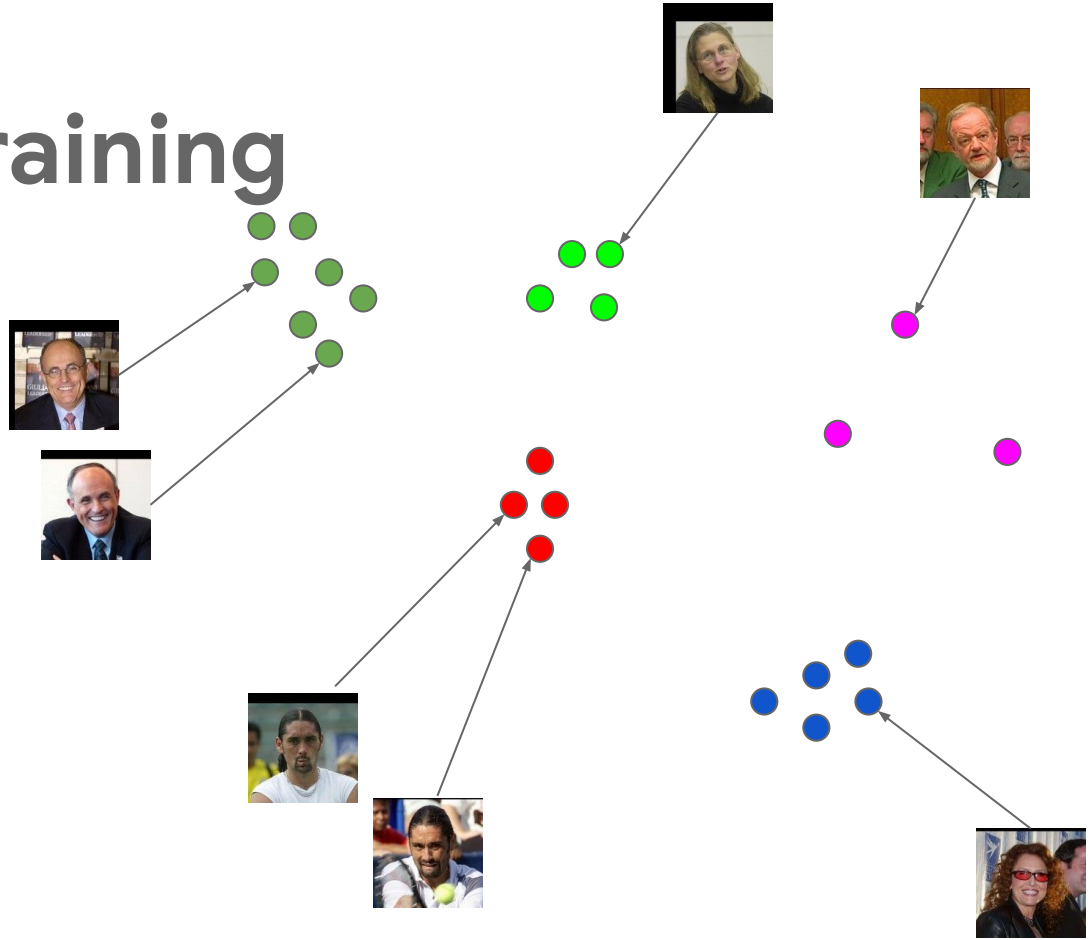
FaceNet



$$L = \sum_{i}^{N} \left[\|f(x_i^a) - f(x_i^p)\|_2^2 - \|f(x_i^a) - f(x_i^n)\|_2^2 + \alpha \right]_+$$

slide content: F. Schroff and D. Kalenichenko

Training

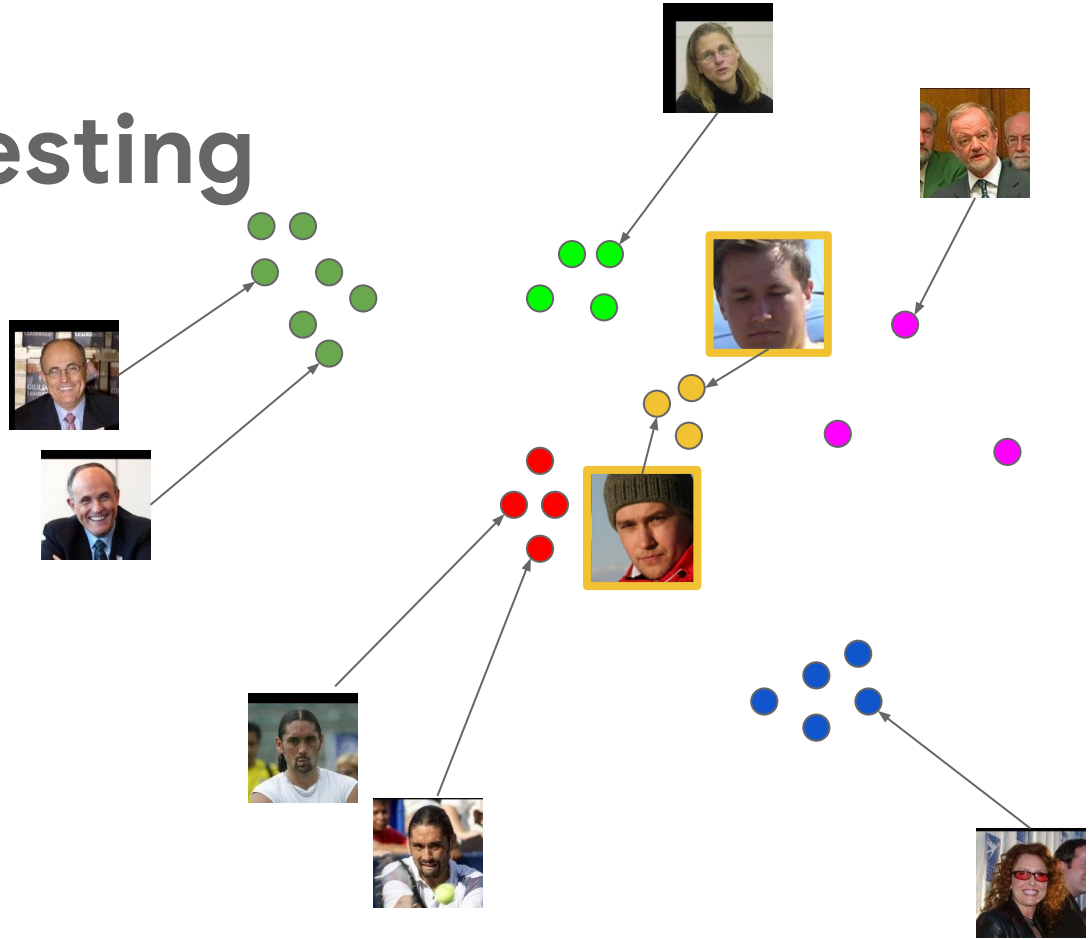


Training

Testing

?

Testing



Introduction

Embeddings

Uncertainty Representations (1)

Discussion

Are two images the same class or not?

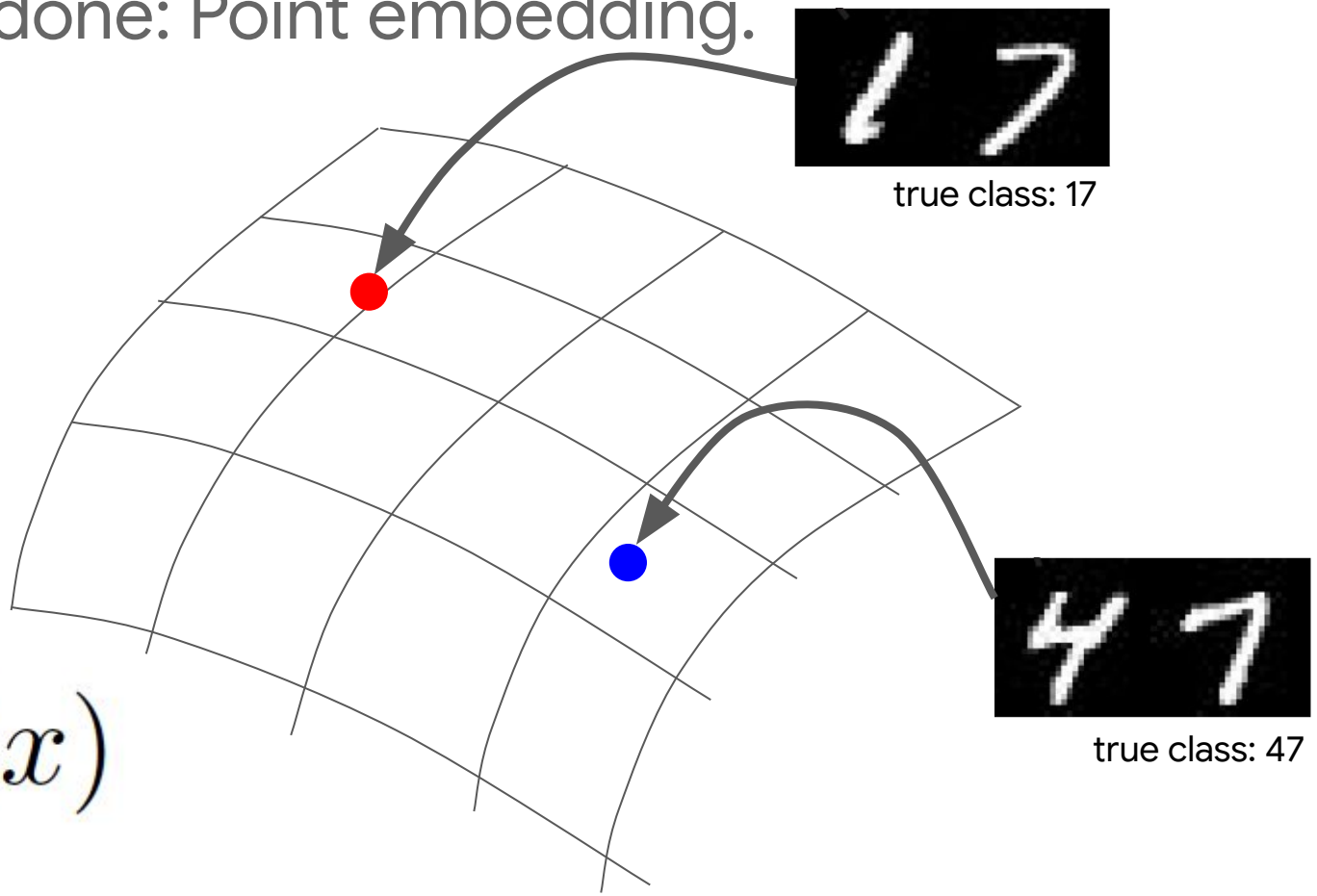
$$p(m|x, x')$$

We typically employ:

$$z = f(x)$$

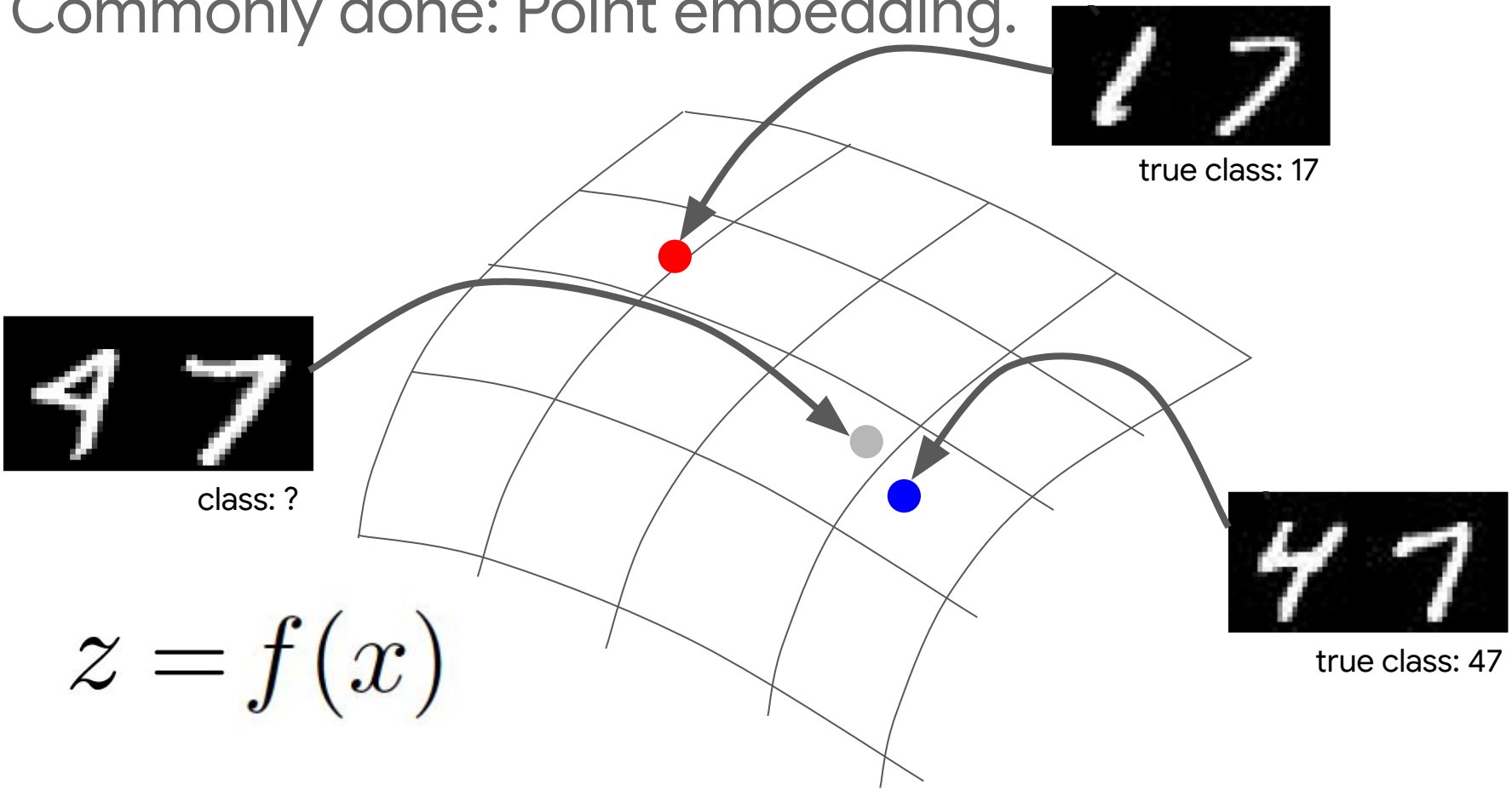
Seong Joon Oh, Kevin Murphy, Jiyan Pan, Joseph Roth, Florian Schroff, Andrew Gallagher, Modeling Uncertainty with Hedged Instance Embedding, https://arxiv.org/abs/1810.00319

Commonly done: Point embedding.



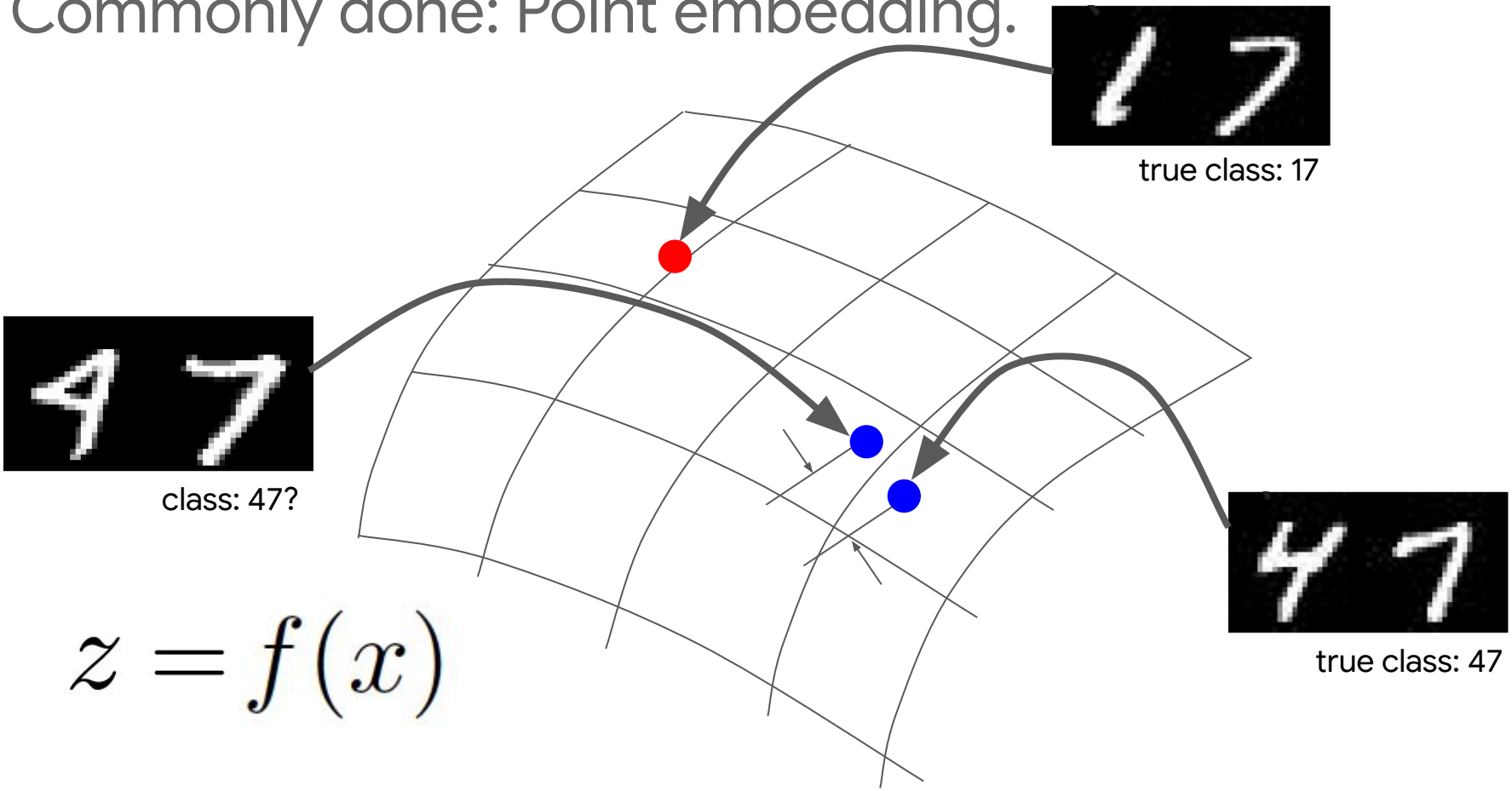
$$z = f(x)$$

Commonly done: Point embedding.



$$z = f(x)$$

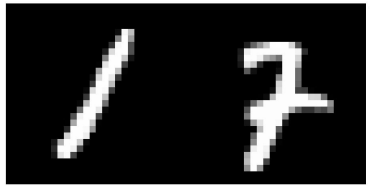
Commonly done: Point embedding.



$$z = f(x)$$

Commonly done: Point embedding.

true class: 17



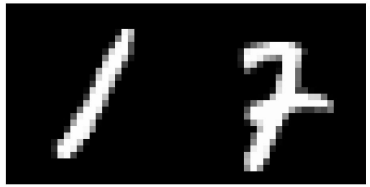
class: ?

true class: 47

$$z = f(x)$$

Commonly done: Point embedding.

true class: 17



class: 17?

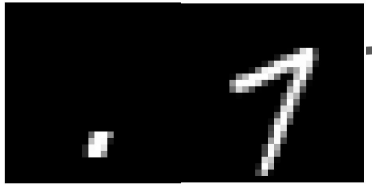
true class: 47

$$z = f(x)$$

Commonly done: Point embedding.

true class: 17

?

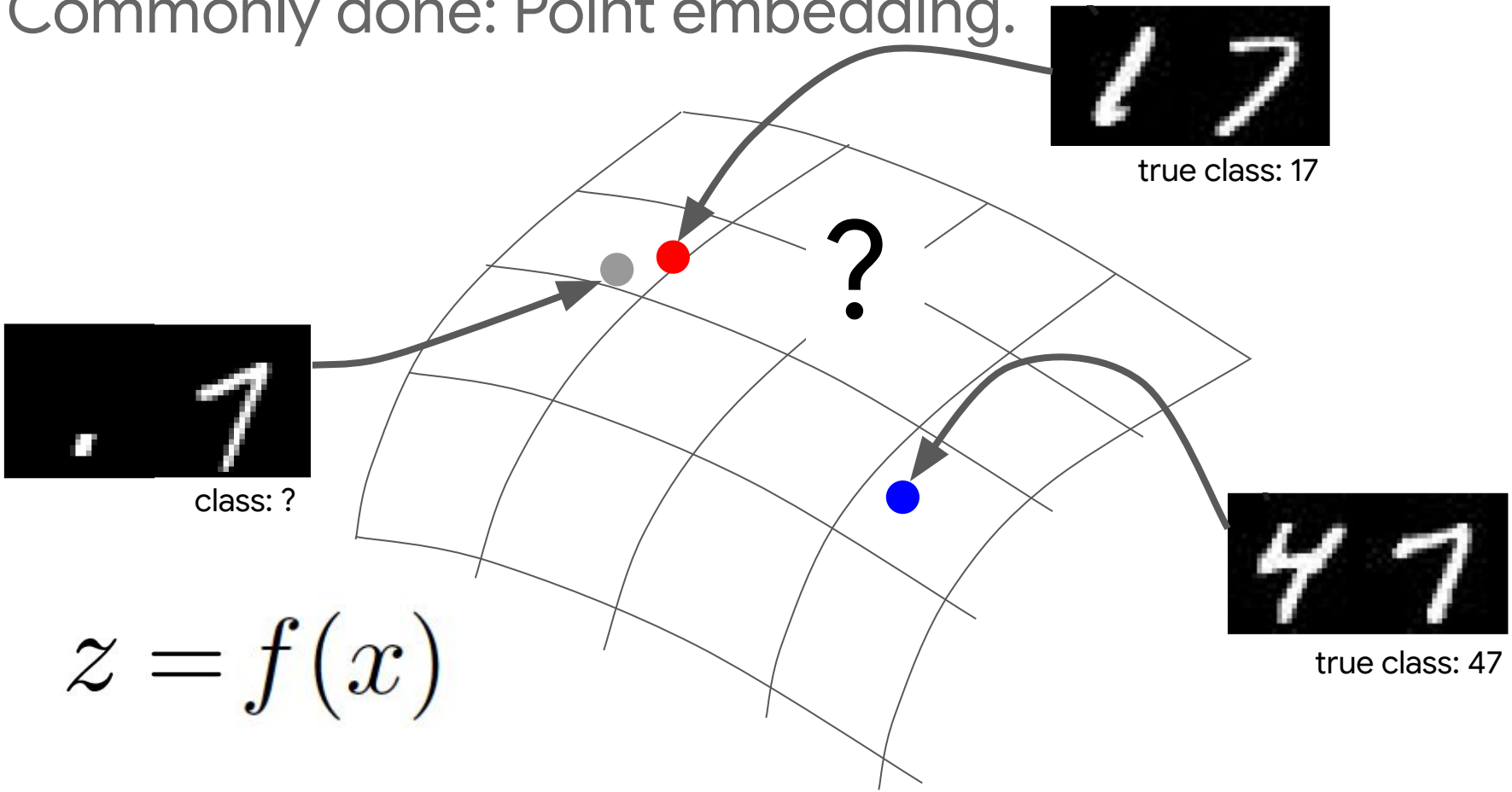


class: ?

true class: 47

$$z = f(x)$$

Commonly done: Point embedding.

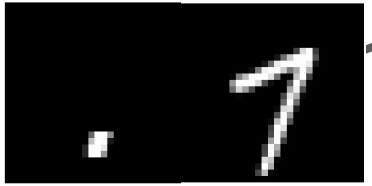


$$z = f(x)$$

Commonly done: Point embedding.

true class: 17

?

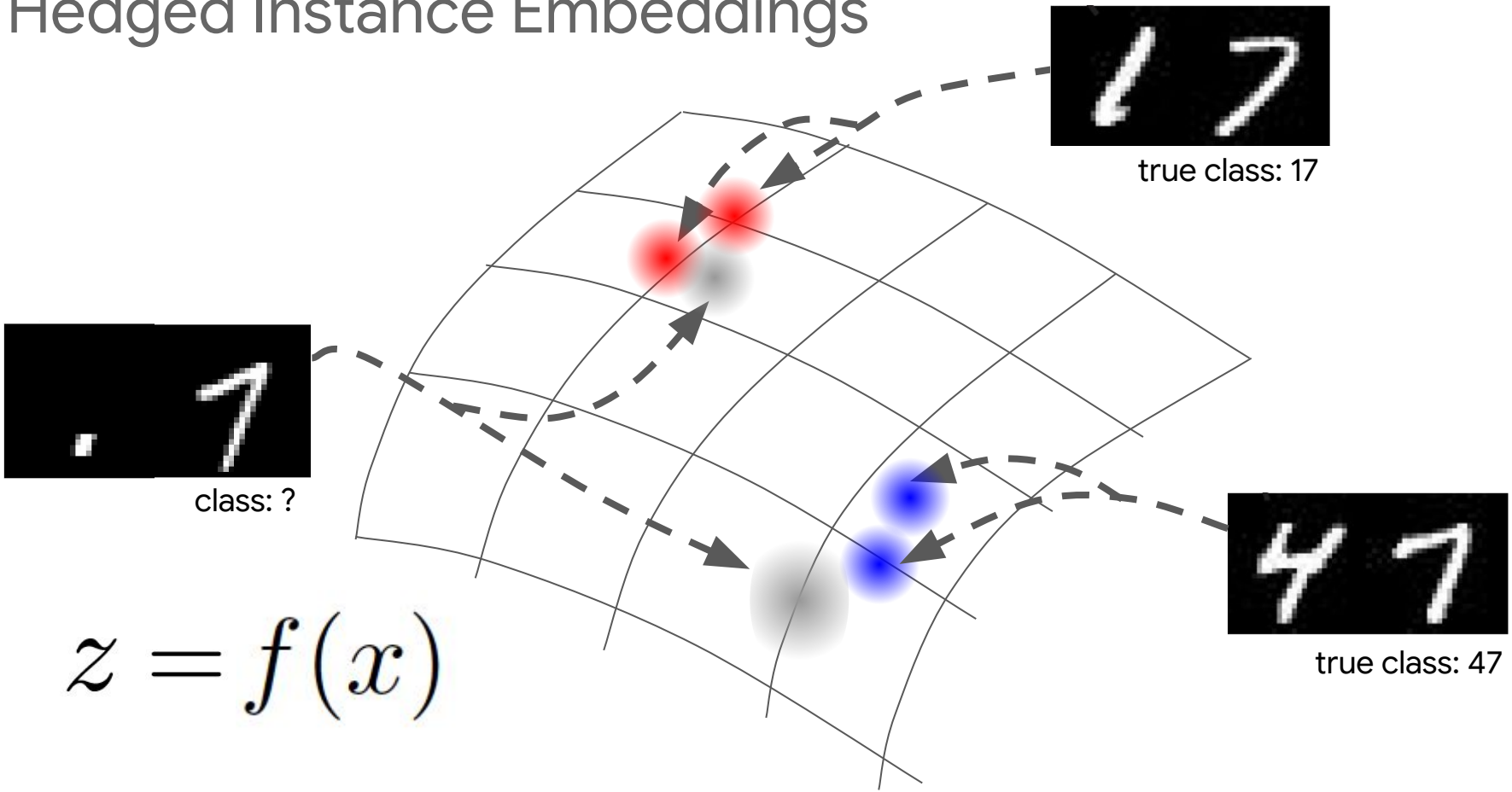


class: ?

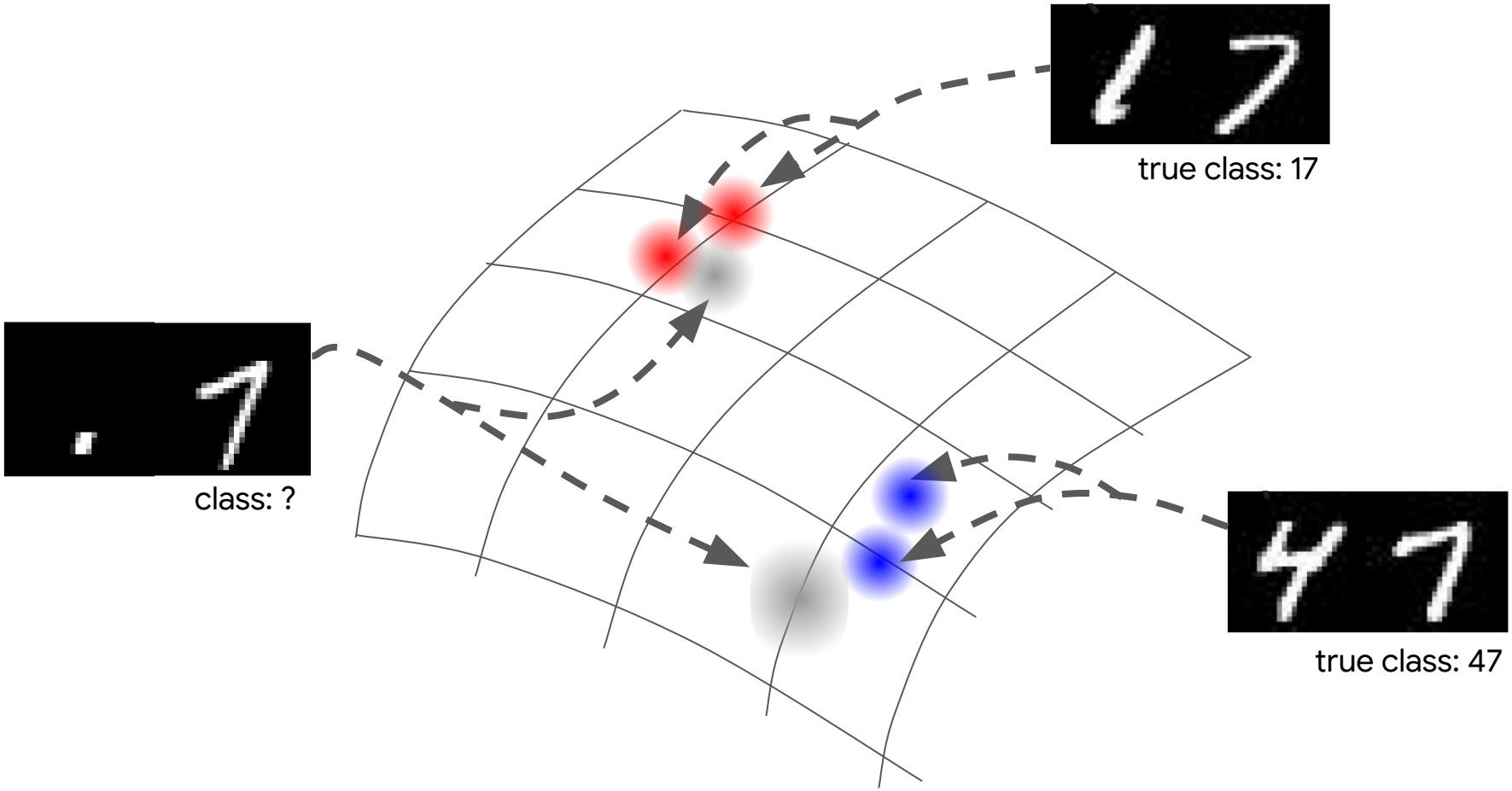
true class: 47

$$z = f(x)$$

Hedged Instance Embeddings



$$z = f(x)$$


true class: 17
class: ?
true class: 47

Metric embedding as distributions.

Formulate as discriminative task: given pair, predict match.

$$p(m|x, x')$$

Introduce embedding as latent variable:

$$p(m|x, x') = \int p(m|z, z')p(z|x)p(z'|x')\,\mathrm{d}z\mathrm{d}z'$$

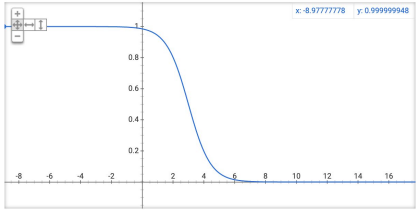
Embedding distributions must be:

- **parameterizable**
- **sampleable**

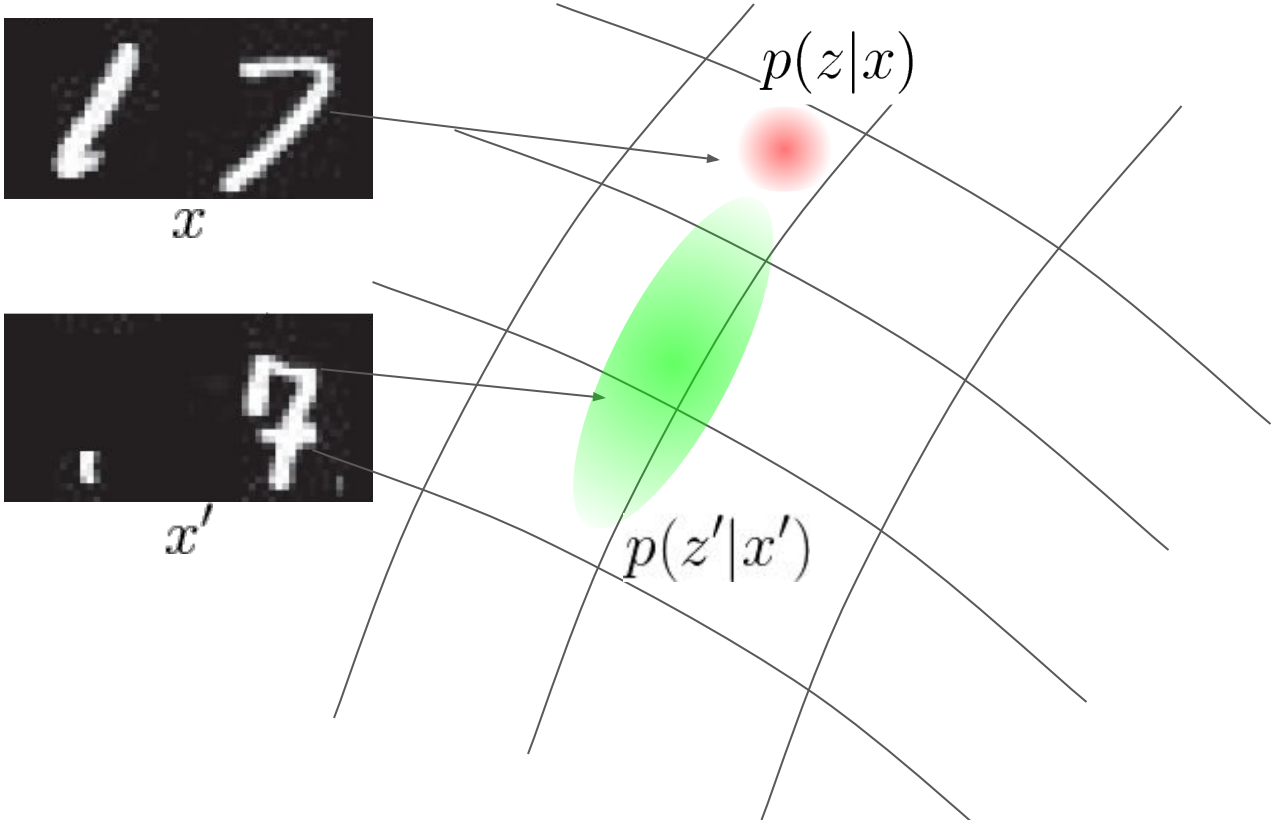
Match probability estimation.

$$p(m|x,x') = \int p(m|z,z')p(z|x)p(z'|x')\,\mathrm{d}z\mathrm{d}z'$$

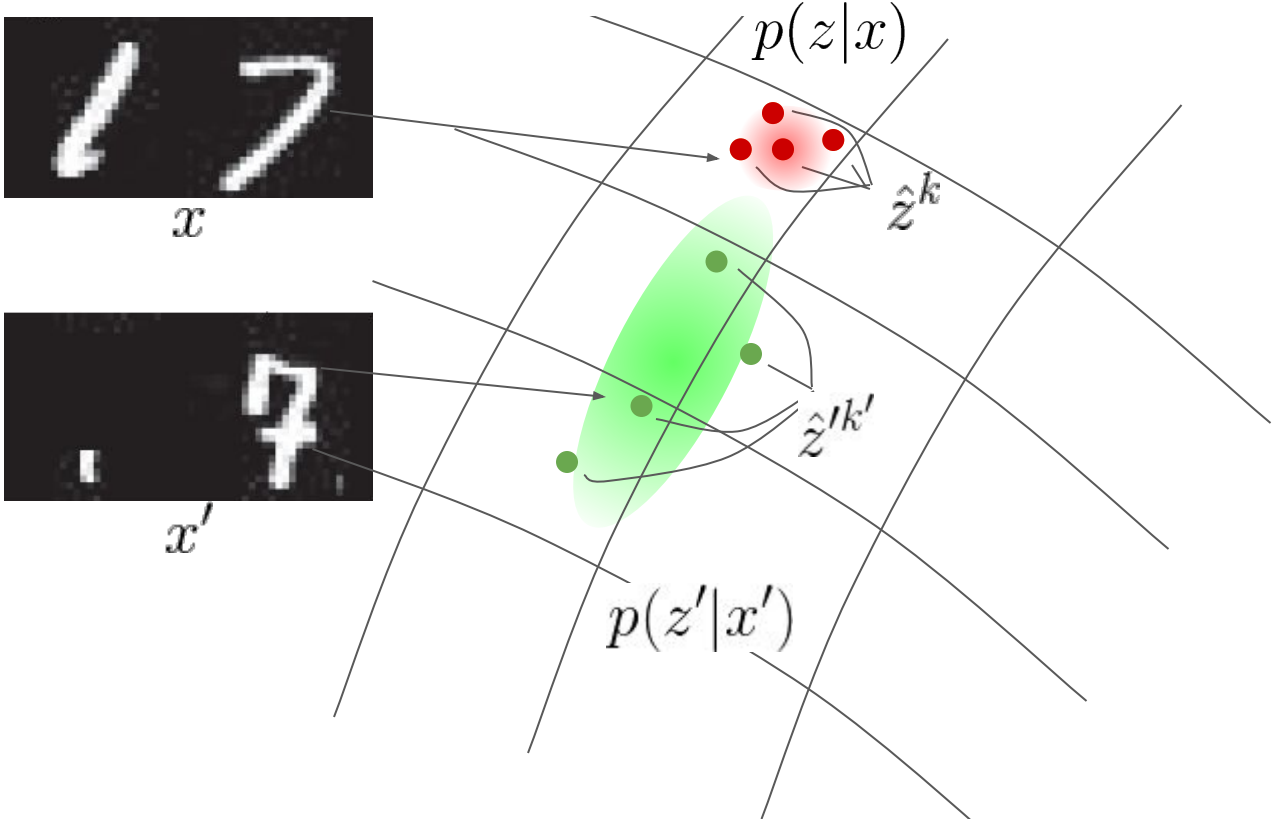
$$p(m|z,z') := s(a||z-z'||_2 + b)$$



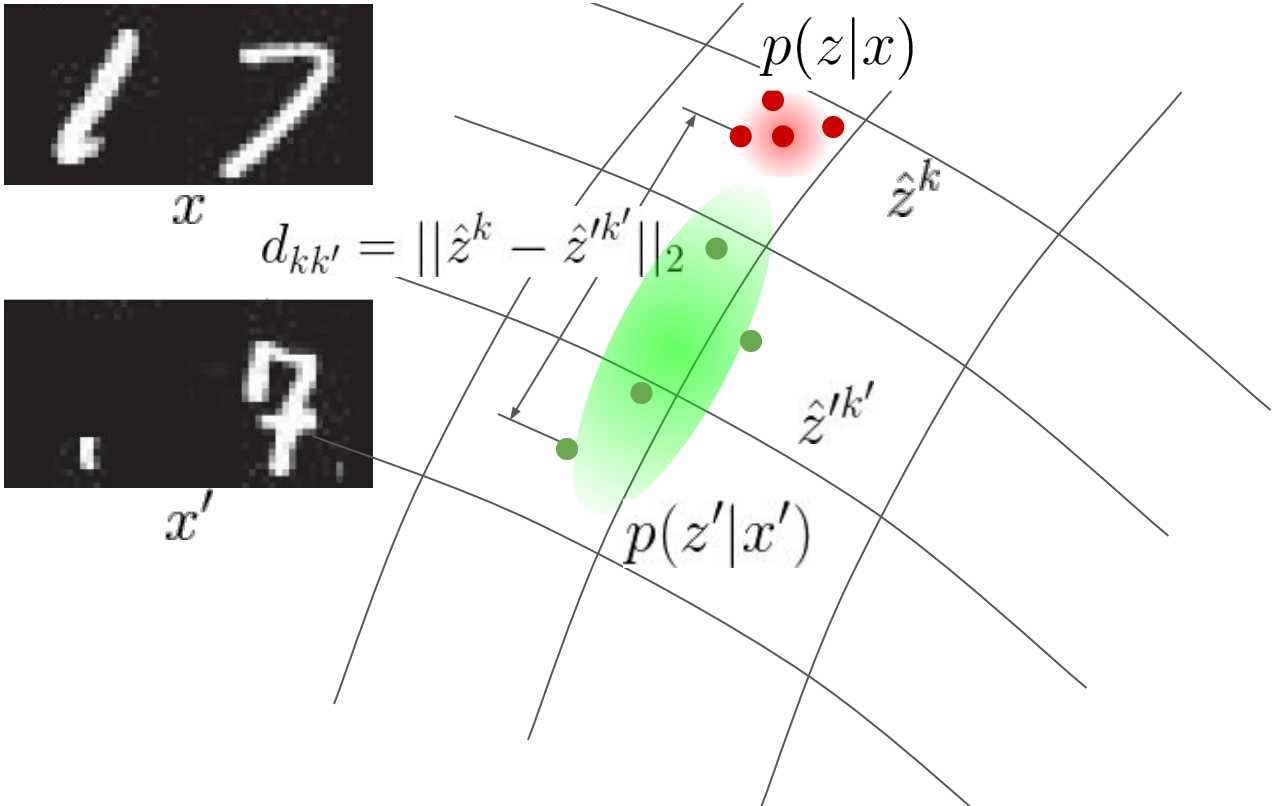
Computing $p(m|x, x')$



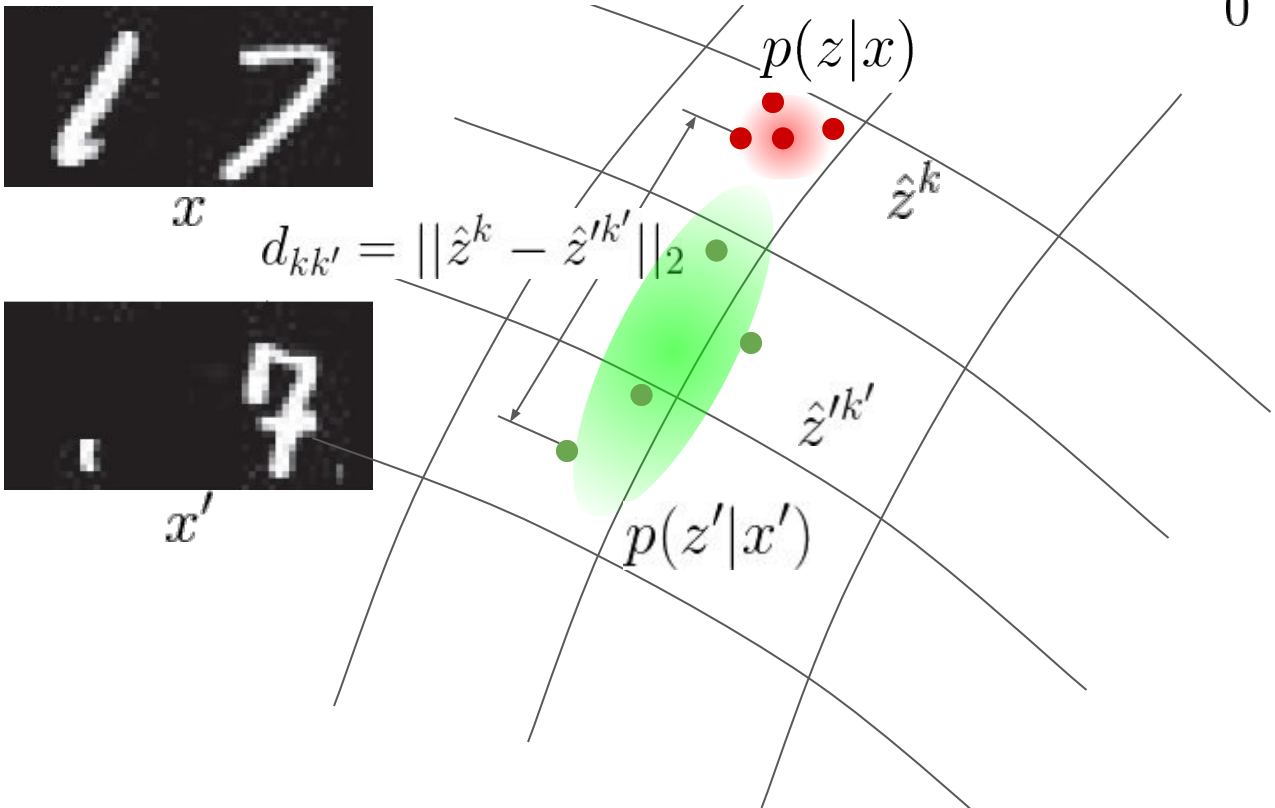
Computing $p(m|x, x')$



Computing $p(m|x, x')$



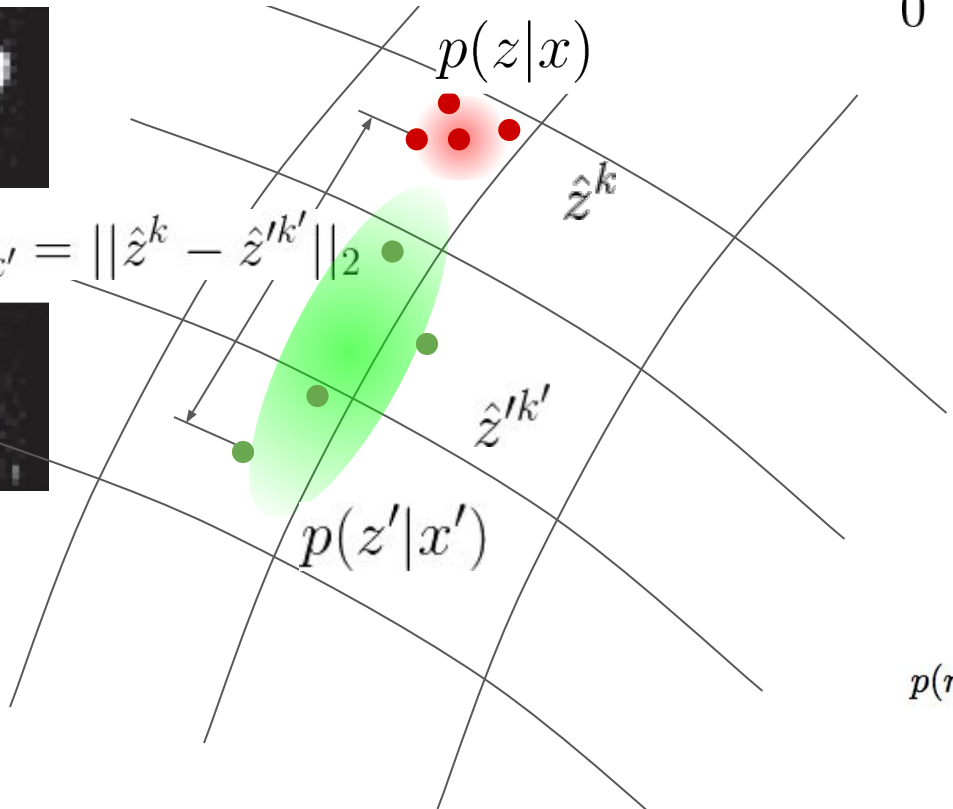
Computing $p(m|x, x')$

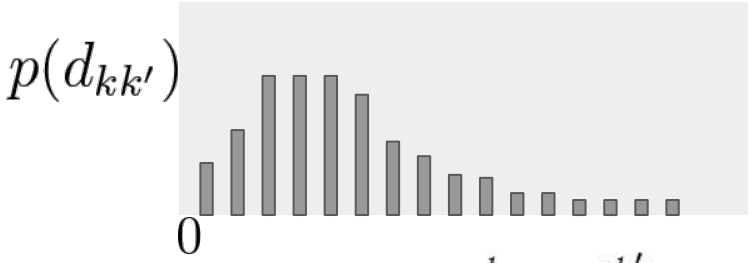


Computing $p(m|x, x')$

x

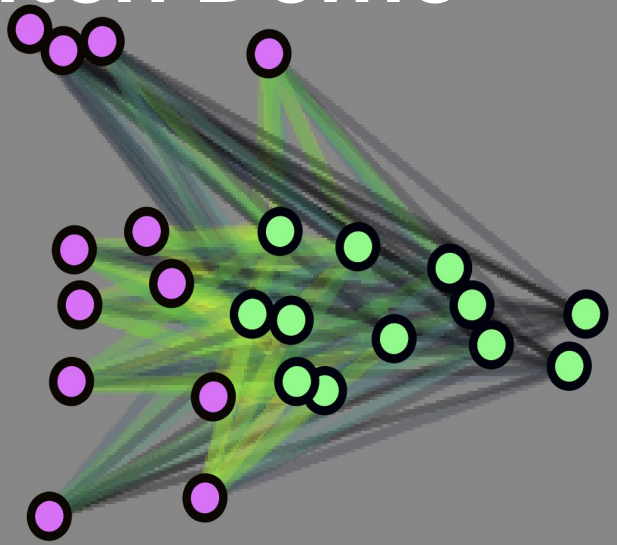
x'

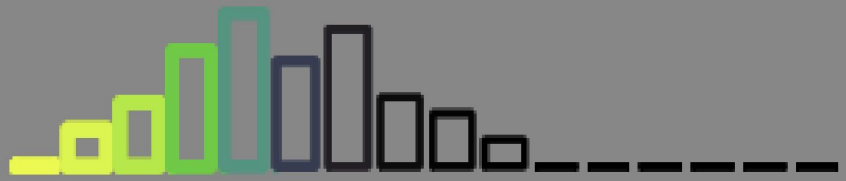




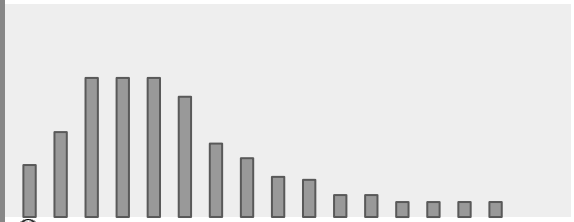
$$p(m|x, x') \approx \frac{1}{K^2} \sum_{k,k'=1}^{K} p(m|\hat{z}^k, \hat{z}'^{k'})$$

Scratch Demo



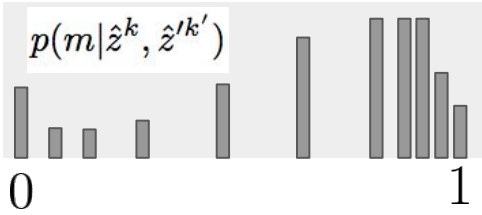


0.52

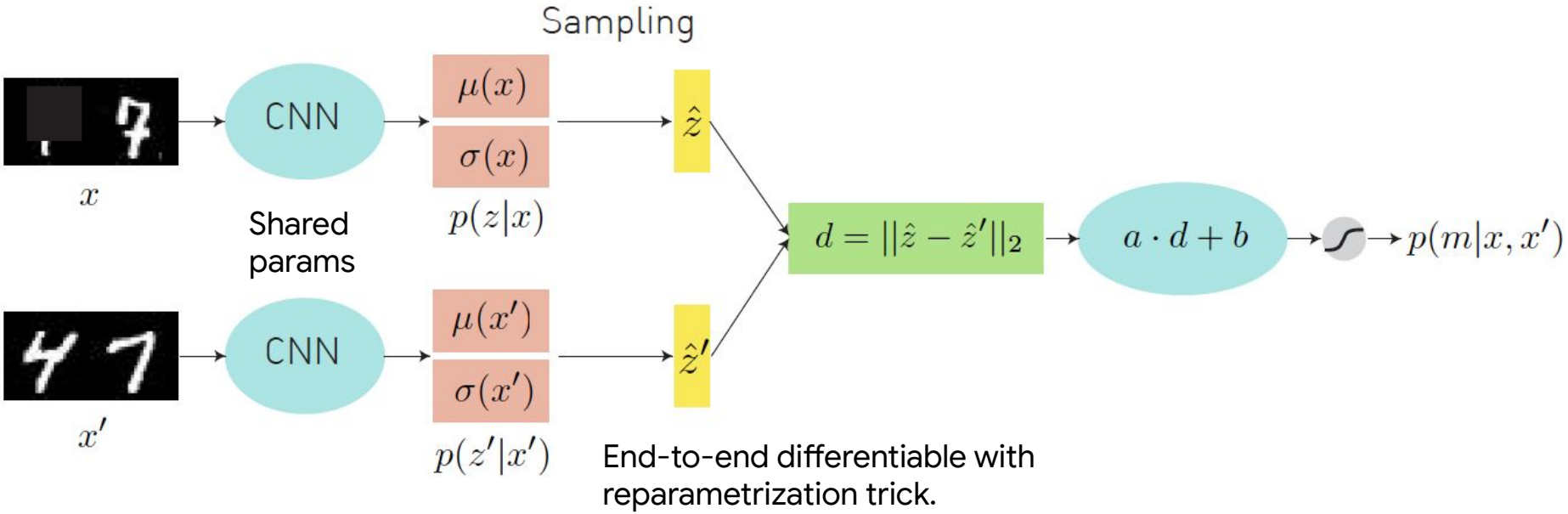


$$d_{kk'} = ||\hat{z}^k - \hat{z}'^{k'}||_2$$

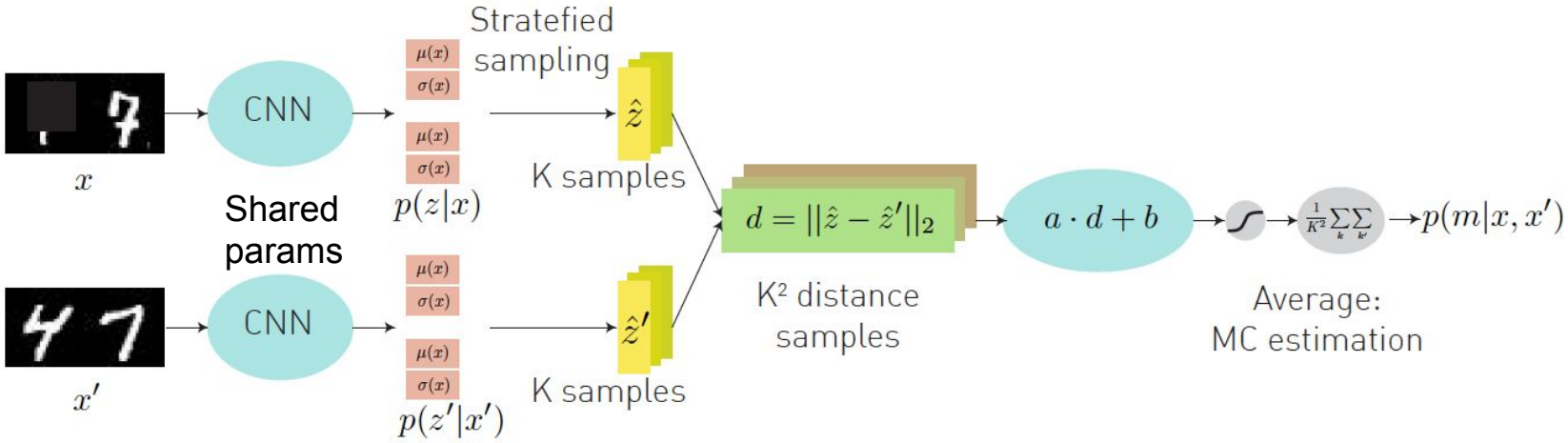
sigmoid match likelihood



$$p(m|x, x') \approx \frac{1}{K^2} \sum_{k,k'=1}^{K} p(m|\hat{z}^k, \hat{z}'^{k'})$$


Sampling
CNN
$\mu(x)$
$\sigma(x)$
$\hat{z}$
x
$p(z|x)$
Shared
params
$d = ||\hat{z} - \hat{z}'||_2$
$a \cdot d + b$
$p(m|x, x')$
CNN
$\mu(x')$
$\sigma(x')$
$\hat{z}'$
x'
$p(z'|x')$
End-to-end differentiable with
reparametrization trick.

MoG embedding with multiple samples case.



Training objective - Variational Information Bottleneck

Derived from the Variational Information Bottleneck [from Bayesflow team]:

$$L_m := -\mathbb{E}_{z\sim p(z|x),z'\sim p(z'|x')}\left[\log p(m|z,z')\right] + \beta\cdot\left[\mathrm{KL}(p(z|x)||r(z)) + \mathrm{KL}(p(z'|x')||r(z'))\right]$$

Log likelihood term: Binary cross-entropy loss for match prediction.

KL term: Controls the compression level for z. Unit Gaussian.

Uncertainty measure: Self-mismatch.

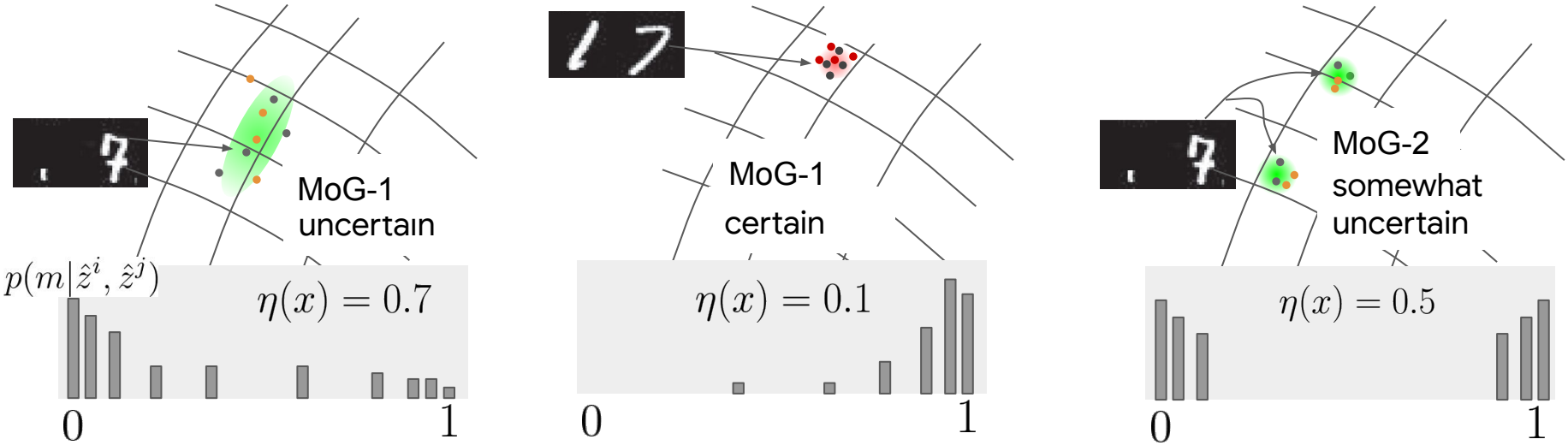
We desire a measure $\eta(x)$ that, given input x, one can guess its performance on downstream tasks (e.g., verification, recognition).

$$\eta(x) := 1 - p(m|x, x)$$

Uncertainty measure: Self-mismatch.

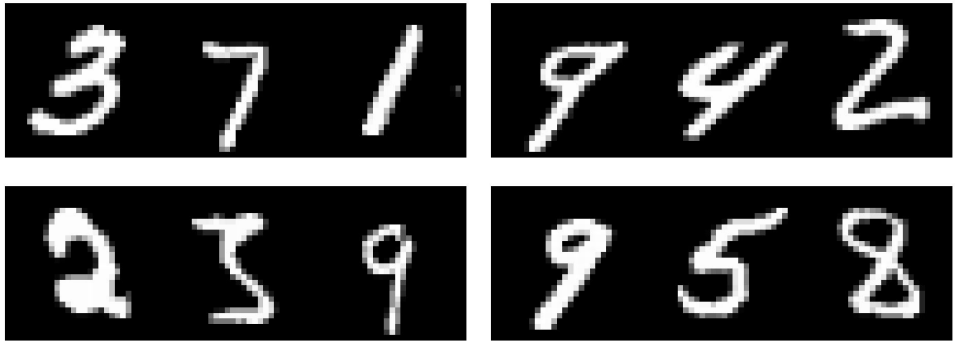
We desire a measure $\eta(x)$ that, given input x, one can guess its performance on downstream tasks (e.g., verification, recognition).

$$\eta(x) := 1 - p(m|x, x)$$



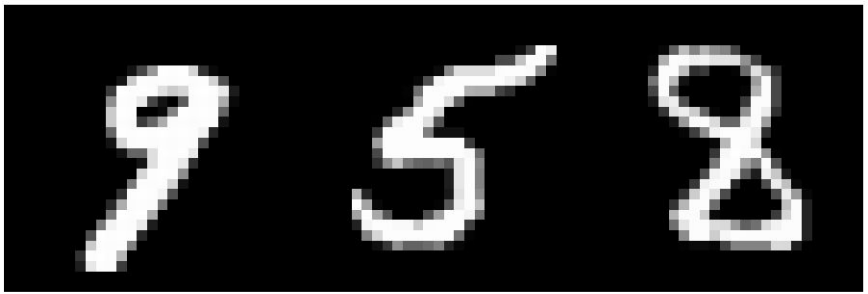
Dataset and Experiments

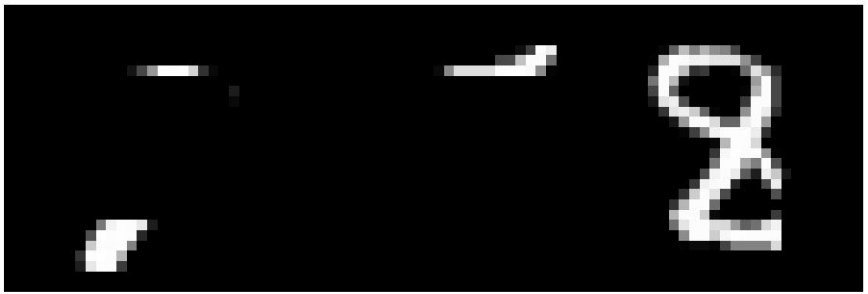
N-digit MNIST Dataset [open source!]



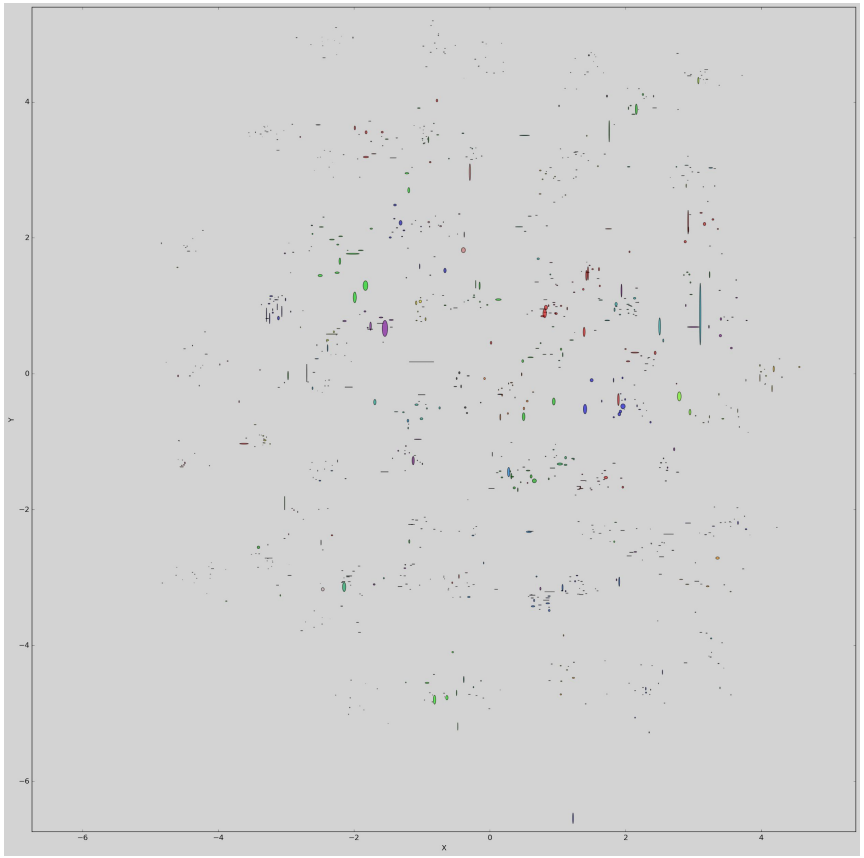
Number Digits	Total Classes	Training Classes	Unseen Test Classes	Seen Test Classes	Training Images	Test Images
2	100	70	30	70	100 000	10 000
3	1000	700	300	700	100 000	10 000

Experimental setup.

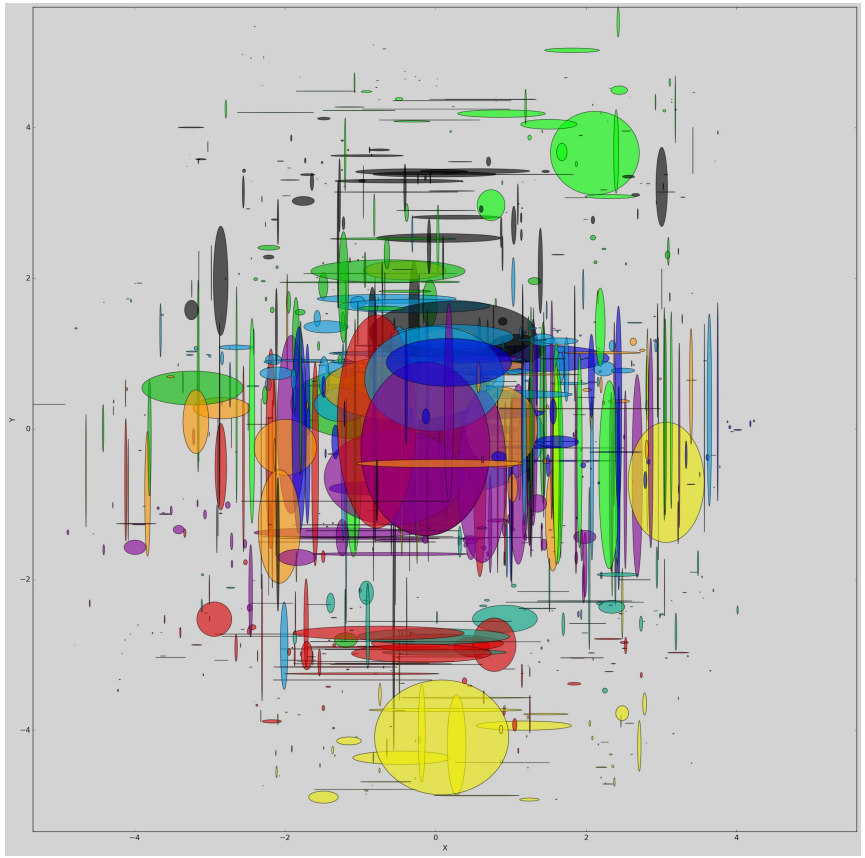




2 Digit MNIST ⟶ 2 dimension Hedged Instance Embeddings (MoG-1)

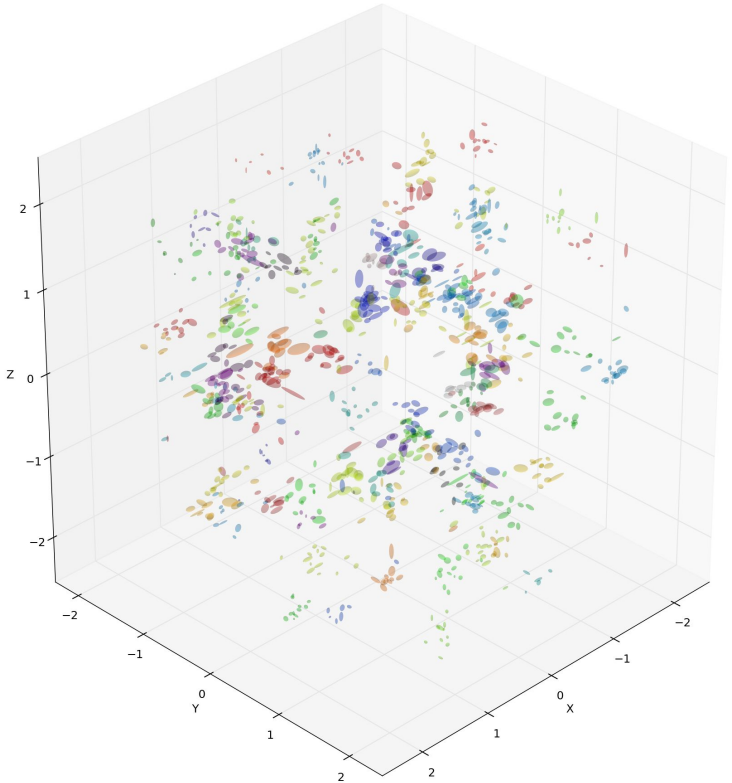


Clean images

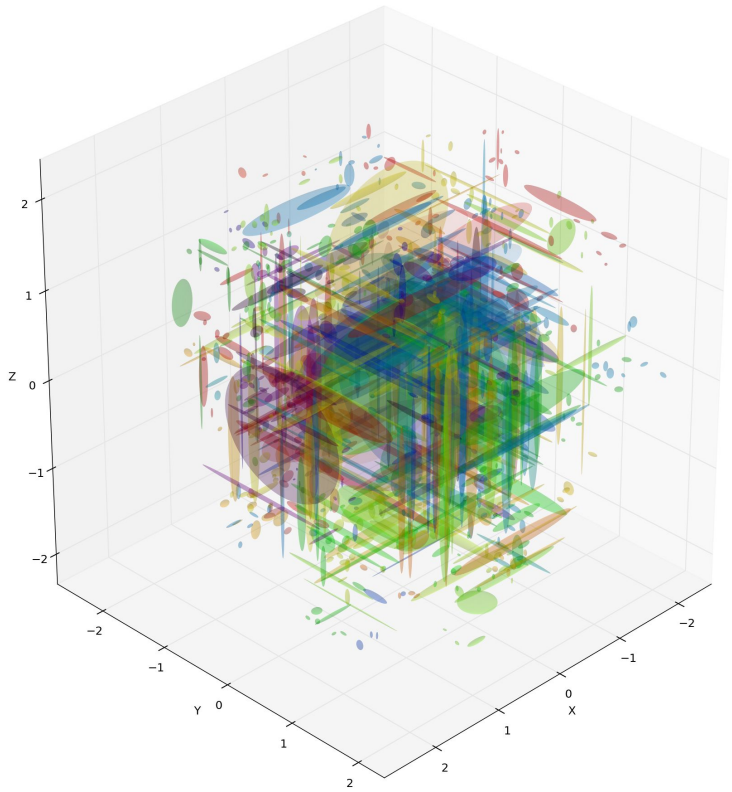


Corrupt images

3 Digit MNIST ⟶ 3 dimension Hedged Instance Embeddings (MoG-1)



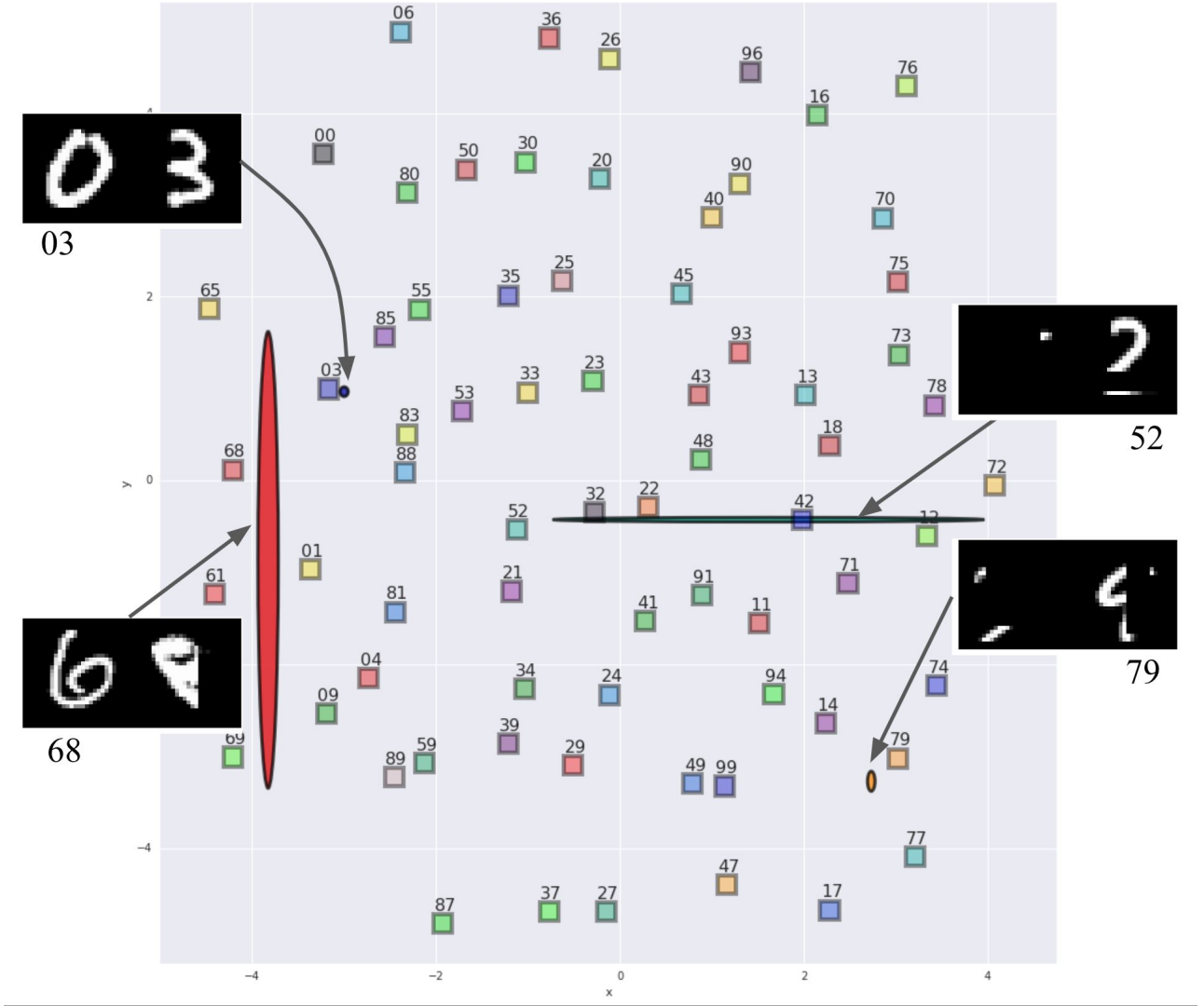
Clean images



Corrupt images

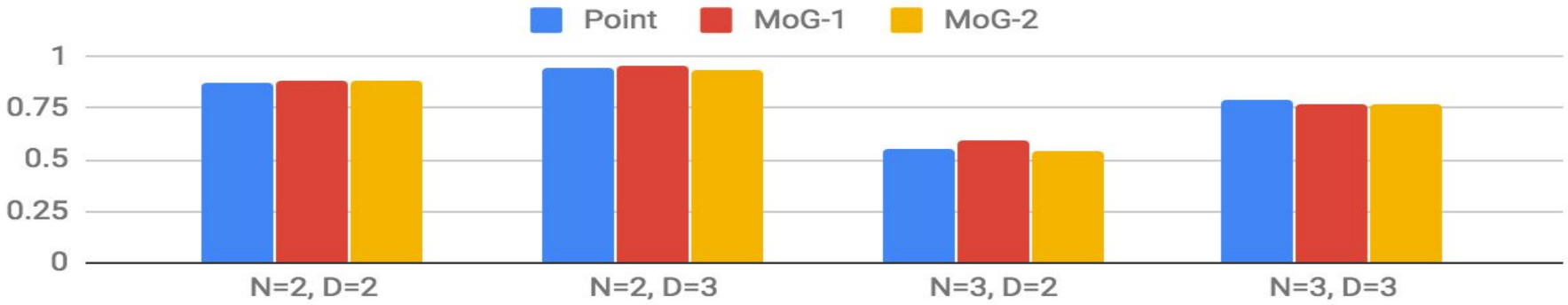
2 Digit ⟶ 2 Dim

Hedged Instance Embeddings (MoG-1)



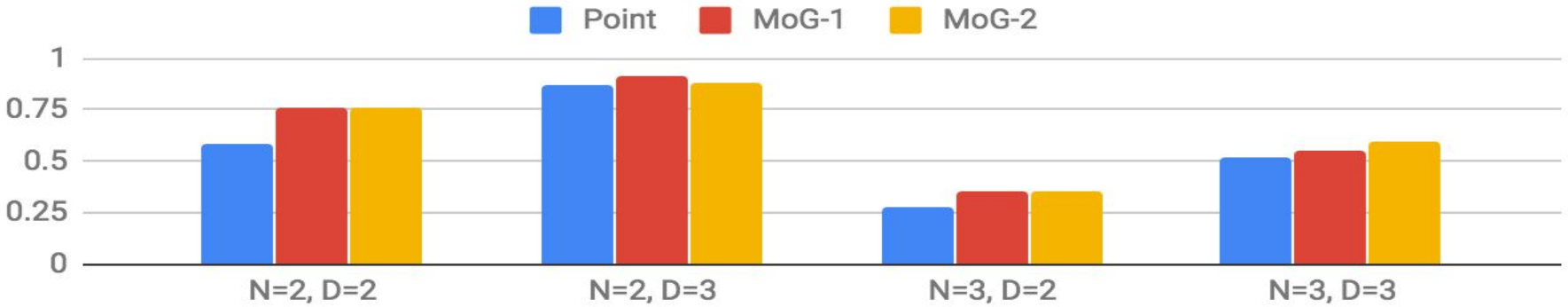
Task Performance

	$N=2, D=2$			$N=2, D=3$			$N=3, D=2$			$N=3, D=3$		
	point	MoG-1	MoG-2	point	MoG-1	MoG-2	point	MoG-1	MoG-2	point	MoG-1	MoG-2
Verification AP												
clean	0.987	0.989	0.990	0.996	0.996	0.996	0.978	0.981	0.976	0.987	0.989	0.991
corrupt	0.880	0.907	0.912	0.913	0.926	0.932	0.886	0.899	0.904	0.901	0.922	0.925
KNN Accuracy												
clean	0.871	0.879	0.888	0.942	0.953	0.939	0.554	0.591	0.540	0.795	0.770	0.766
corrupt	0.583	0.760	0.757	0.874	0.909	0.885	0.274	0.350	0.351	0.522	0.555	0.598

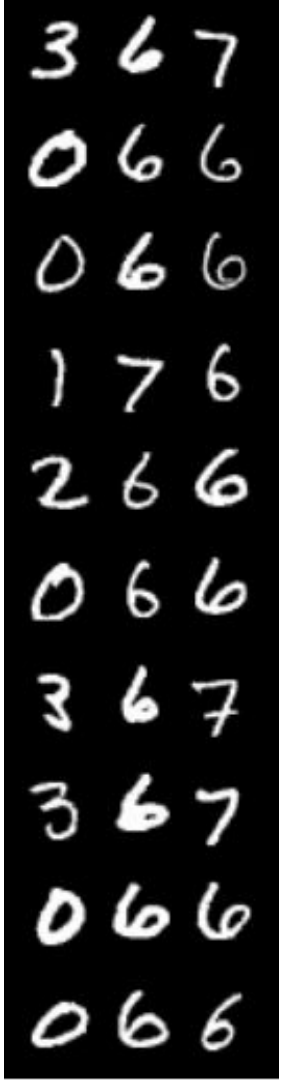


Task Performance

	$N=2, D=2$			$N=2, D=3$			$N=3, D=2$			$N=3, D=3$		
	point	MoG-1	MoG-2	point	MoG-1	MoG-2	point	MoG-1	MoG-2	point	MoG-1	MoG-2
Verification AP												
clean	0.987	0.989	0.990	0.996	0.996	0.996	0.978	0.981	0.976	0.987	0.989	0.991
corrupt	0.880	0.907	0.912	0.913	0.926	0.932	0.886	0.899	0.904	0.901	0.922	0.925
KNN Accuracy												
clean	0.871	0.879	0.888	0.942	0.953	0.939	0.554	0.591	0.540	0.795	0.770	0.766
corrupt	0.583	0.760	0.757	0.874	0.909	0.885	0.274	0.350	0.351	0.522	0.555	0.598



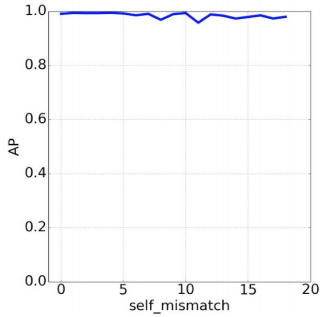
Most certain



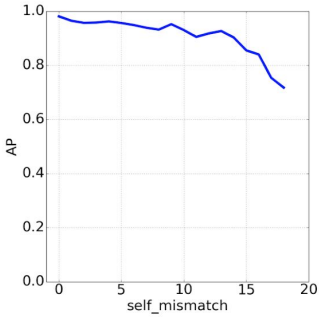
Least certain

Uncertainty Measure

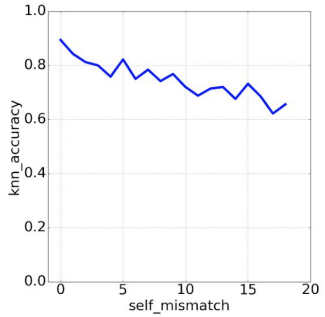
	$N = 2, D = 2$		$N = 3, D = 3$		$N = 3, D = 2$	
	MoG-1	MoG-2	MoG-1	MoG-2	MoG-1	MoG-2
AP Correlation						
clean	0.74 ± 0.03	0.43 ± 0.06	0.51 ± 0.04	0.39 ± 0.02	0.63 ± 0.05	0.28 ± 0.04
corrupt	0.81 ± 0.04	0.79 ± 0.08	0.85 ± 0.04	0.79 ± 0.04	0.82 ± 0.03	0.76 ± 0.03
KNN Correlation						
clean	0.71 ± 0.06	0.57 ± 0.06	0.74 ± 0.07	0.54 ± 0.03	0.76 ± 0.03	0.29 ± 0.12
corrupt	0.47 ± 0.09	0.43 ± 0.10	0.67 ± 0.05	0.34 ± 0.12	0.49 ± 0.06	0.50 ± 0.08



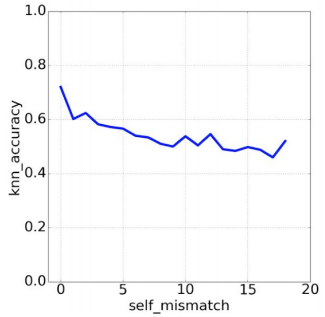
(a) AP for clean test



(b) AP for corrupt test

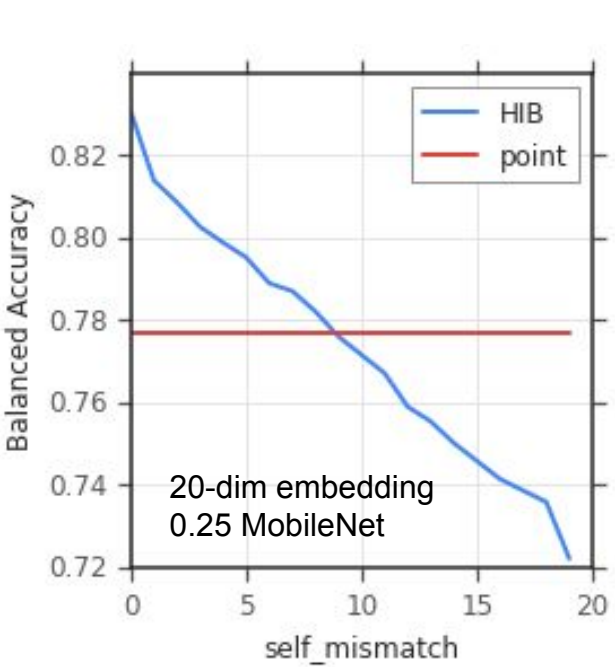


(c) KNN for clean test

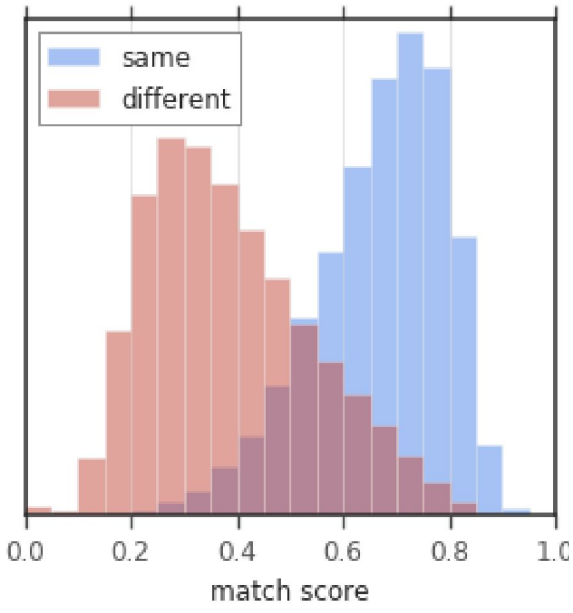


(d) KNN for corrupt test

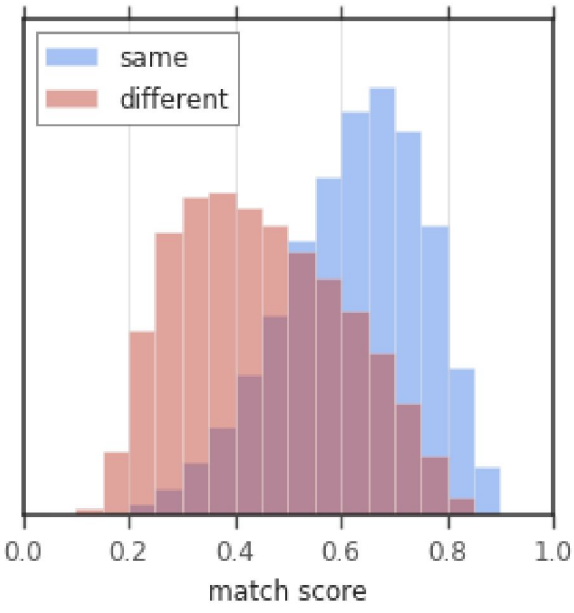
PetNet with Uncertainty



High Certainty Samples



Low Certainty Samples



Uncertain Pet *

roger901

Certain Pet *

unbunt

* Not the actual photos, but similar.

Introduction

Embeddings

Uncertainty Representations (2)

Discussion

Deep Convolutional Neural Network Features and the Original Image

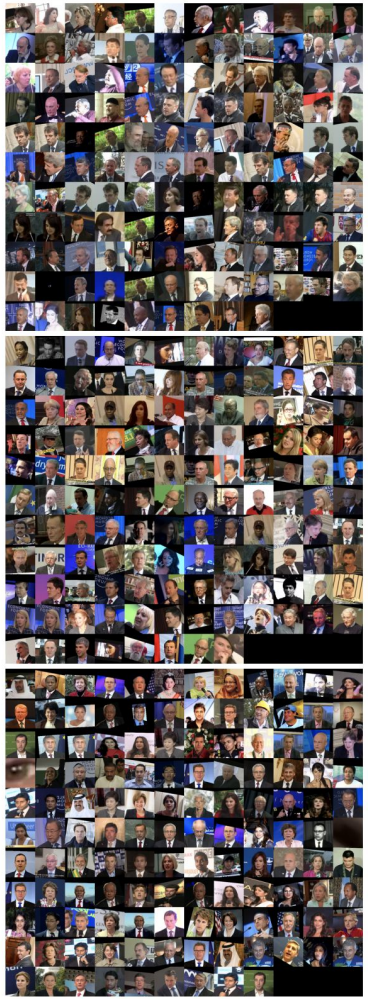
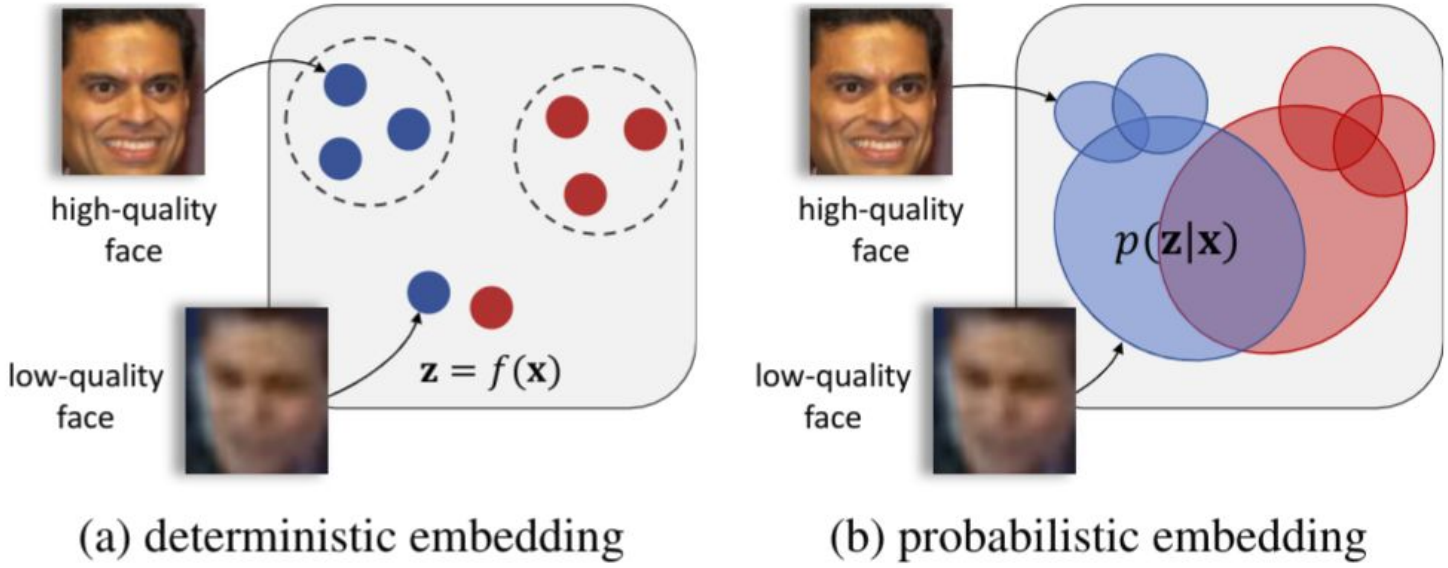


Fig. 6. Images (n=129) sampled at the 20th (top), 50th (middle), and 90th (bottom) percentiles of ranked distances from the origin. Face image quality seems to increase with distance from the center of the DCNN feature space.

Why is L2-Norm Useful?



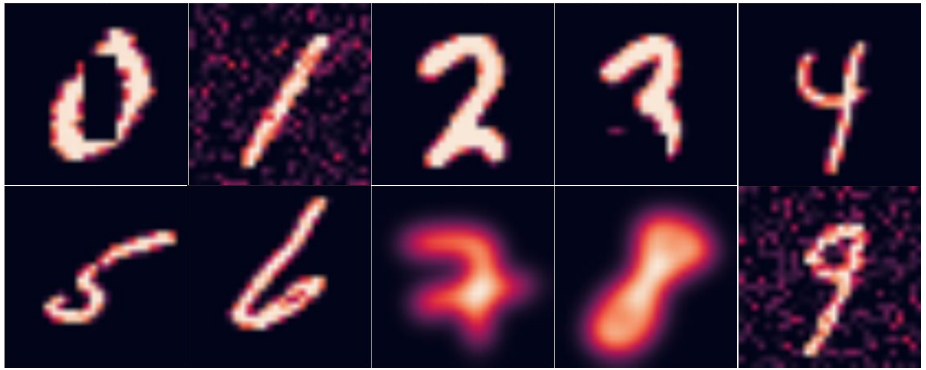
(a) deterministic embedding

(b) probabilistic embedding

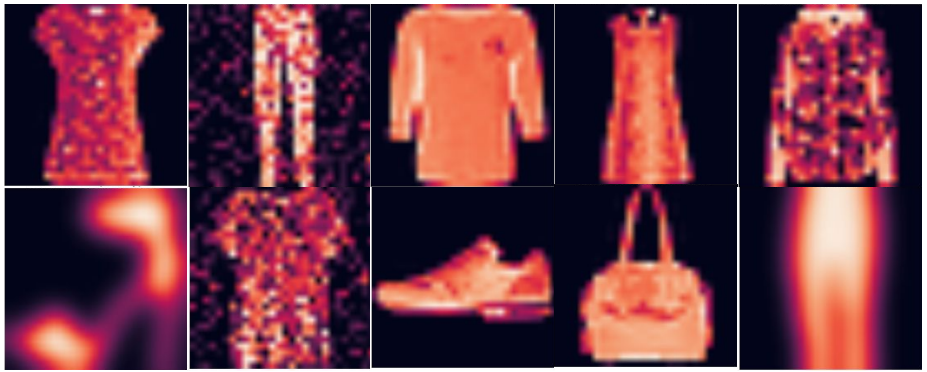
Probabilistic Face Embeddings, Shi & Jain, 2019, https://arxiv.org/abs/1904.09658.

Datasets

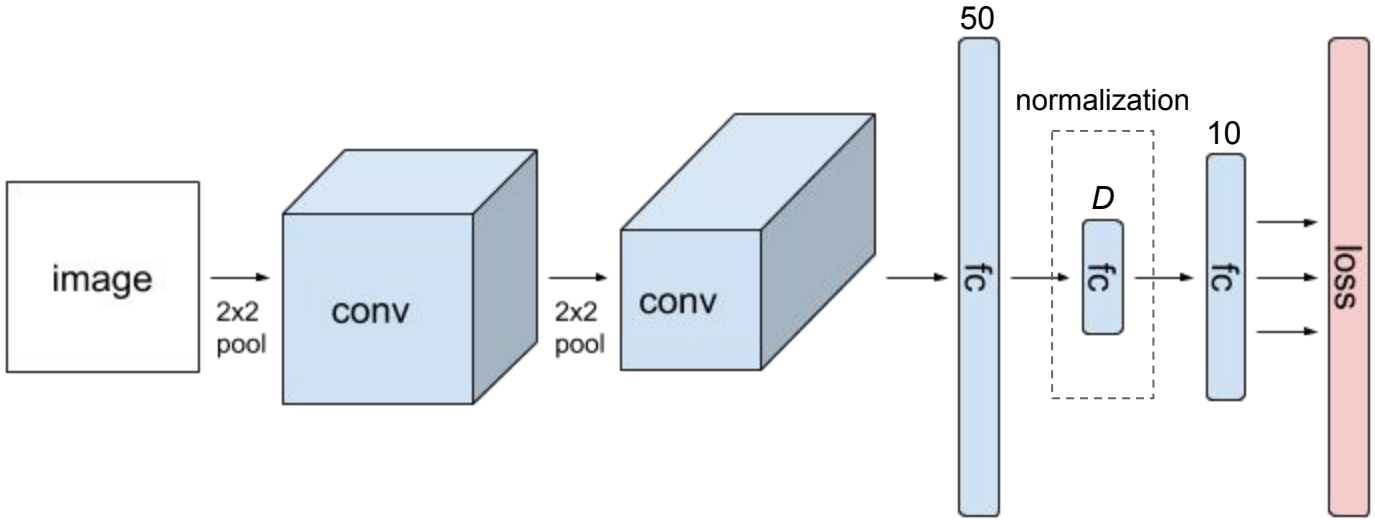
MNIST



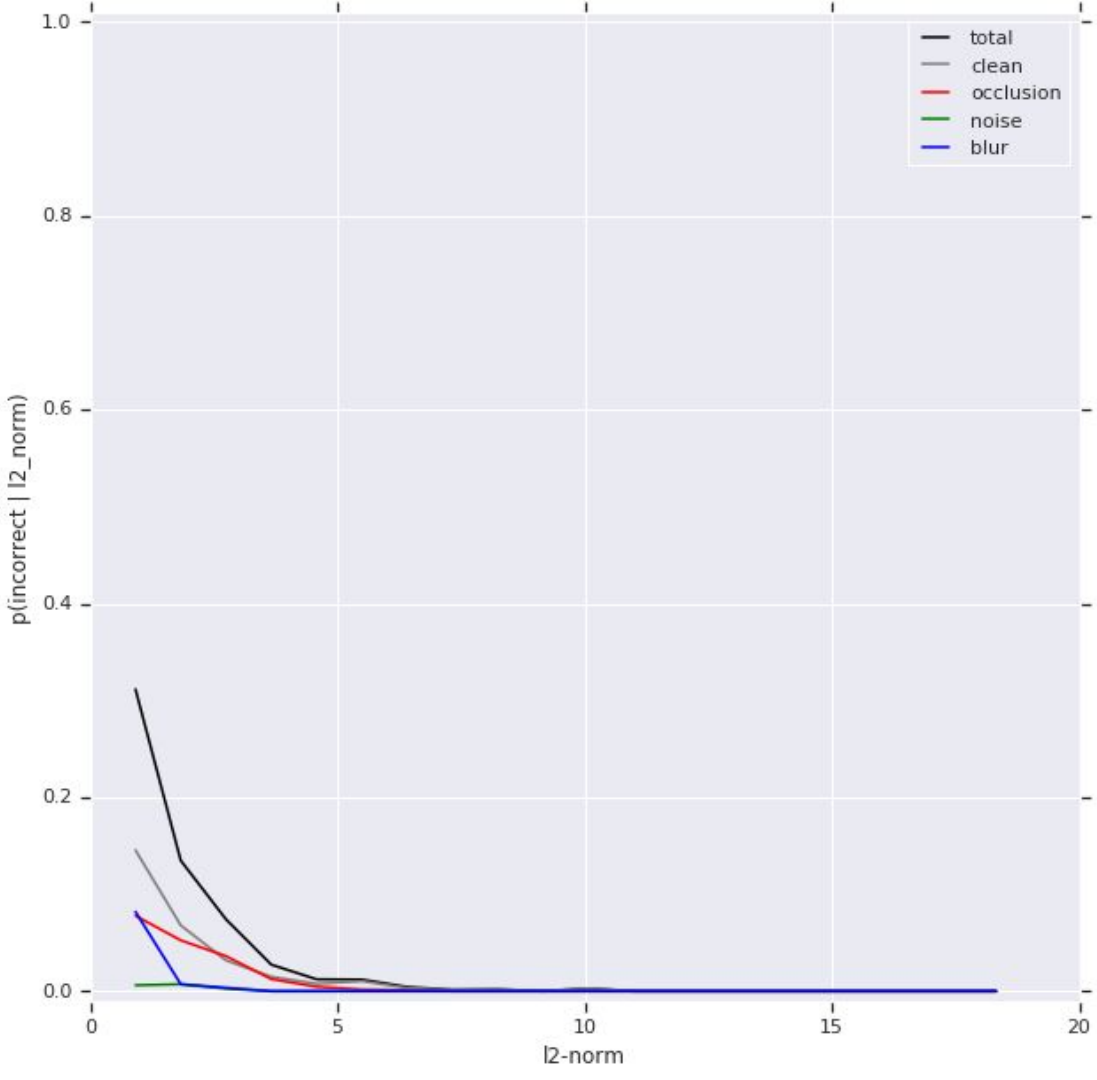
Fashion MNIST



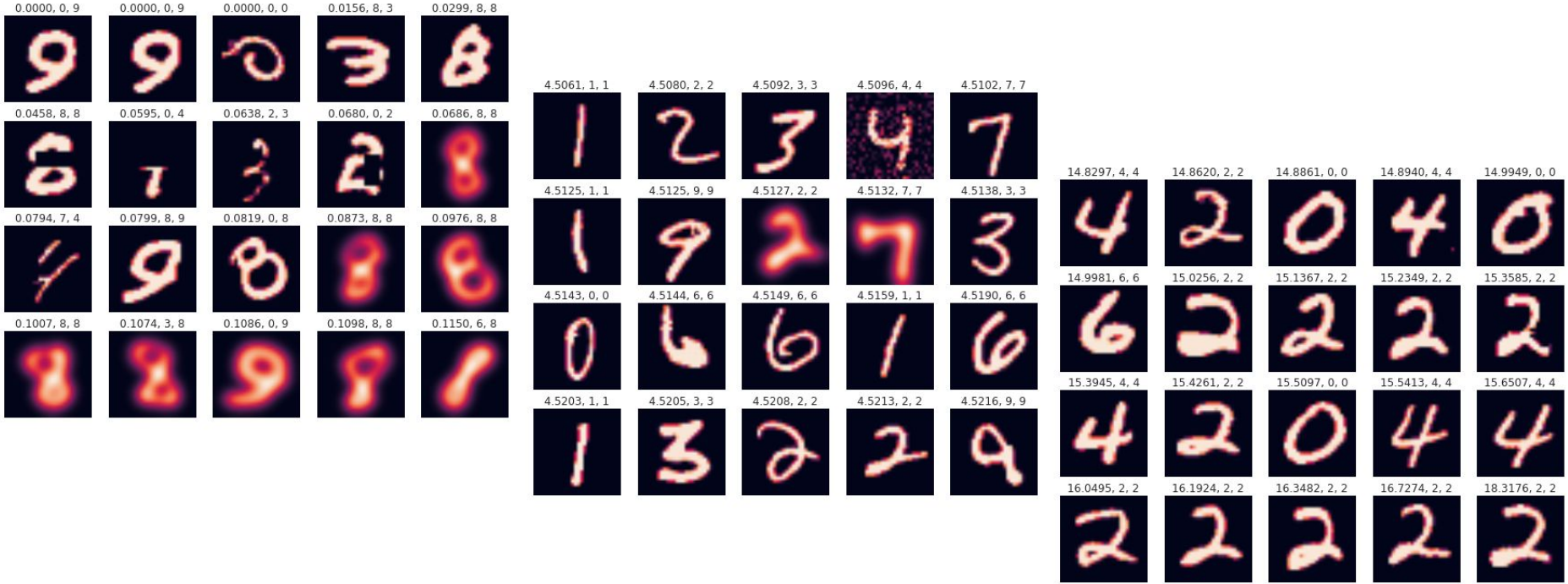
Architecture



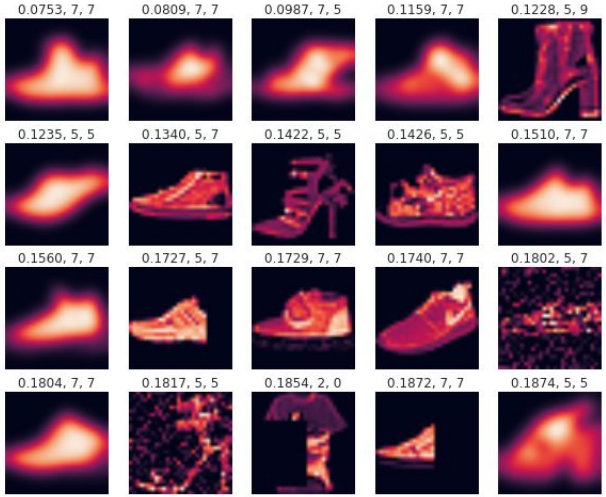
$$L_s = -\frac{1}{M}\sum_{i=1}^{M}\log\frac{\exp\left(W_{y_i}^T f(x_i)+b_{y_i}\right)}{\sum_{j=1}^{C}\exp\left(W_j^T f(x_i+b_j)\right)}$$


total
clean
occlusion
noise
blur
p(incorrect | l2_norm)
l2-norm
0.0
0.2
0.4
0.6
0.8
1.0
0
5
10
15
20

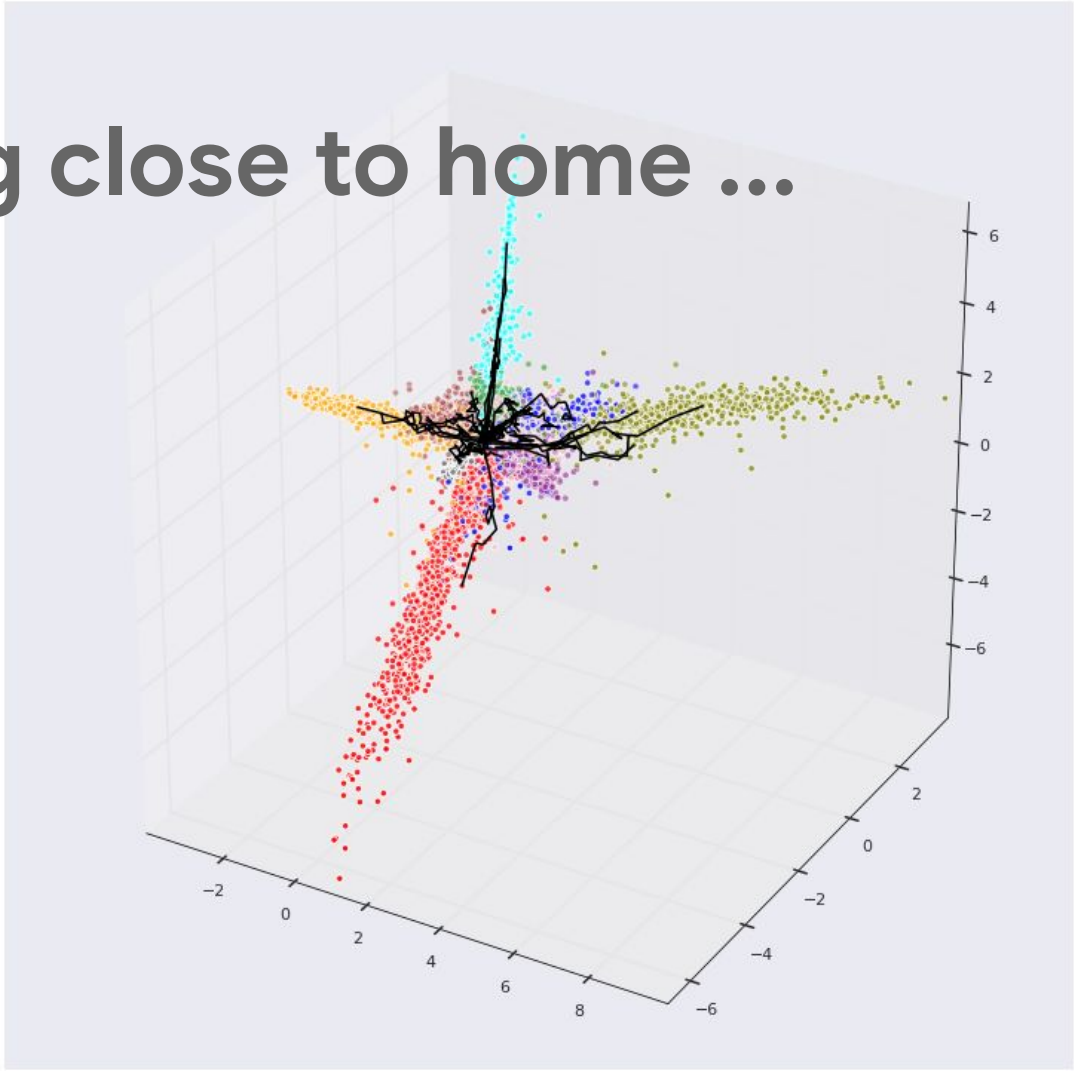
D=3, normalized, cross-entropy loss

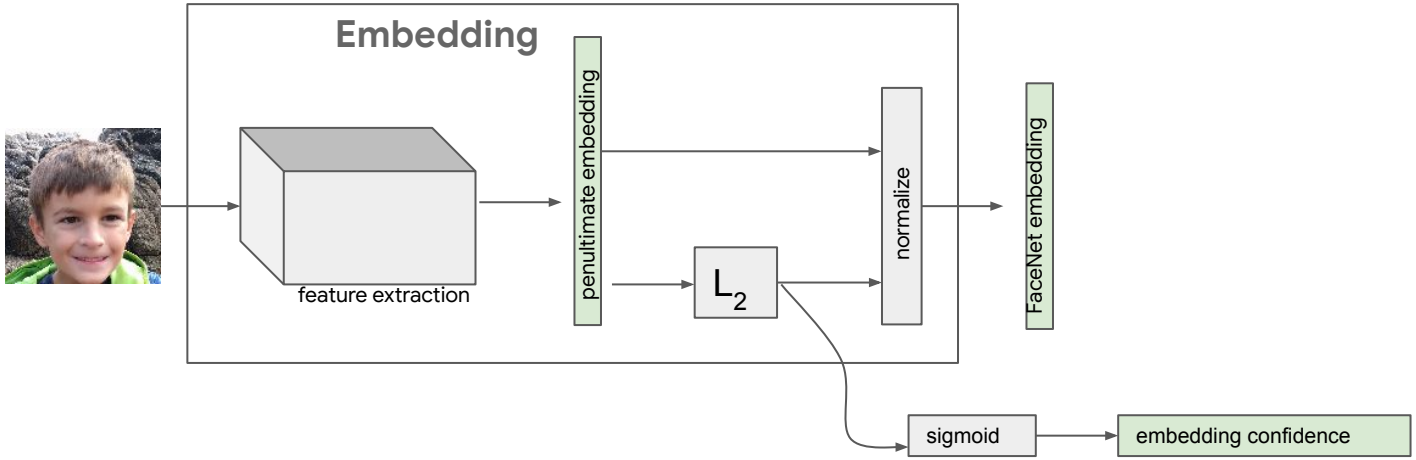


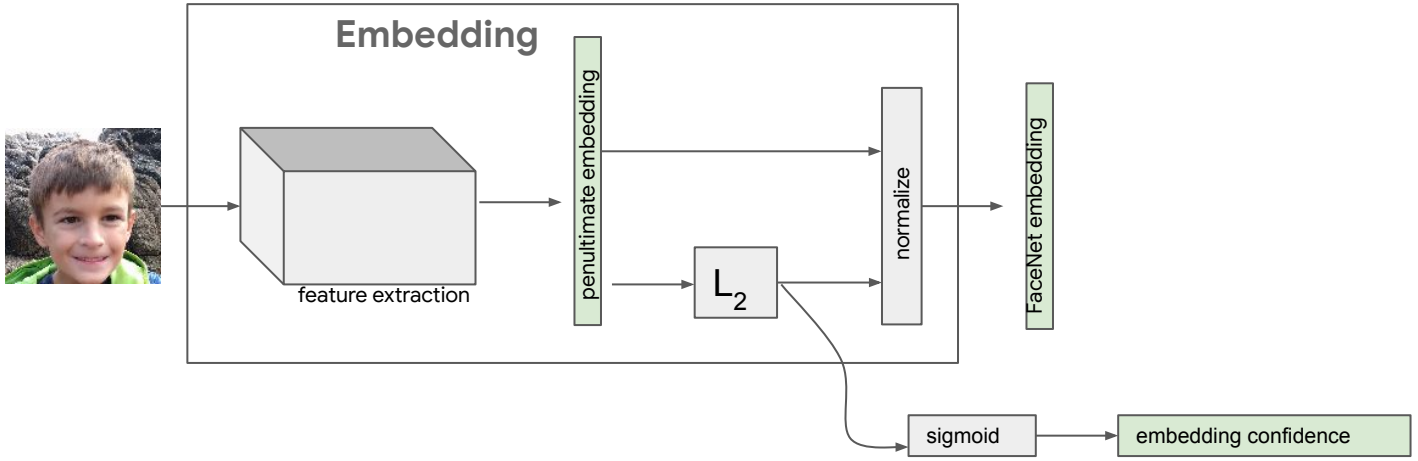
D=4, normalized, triplet loss

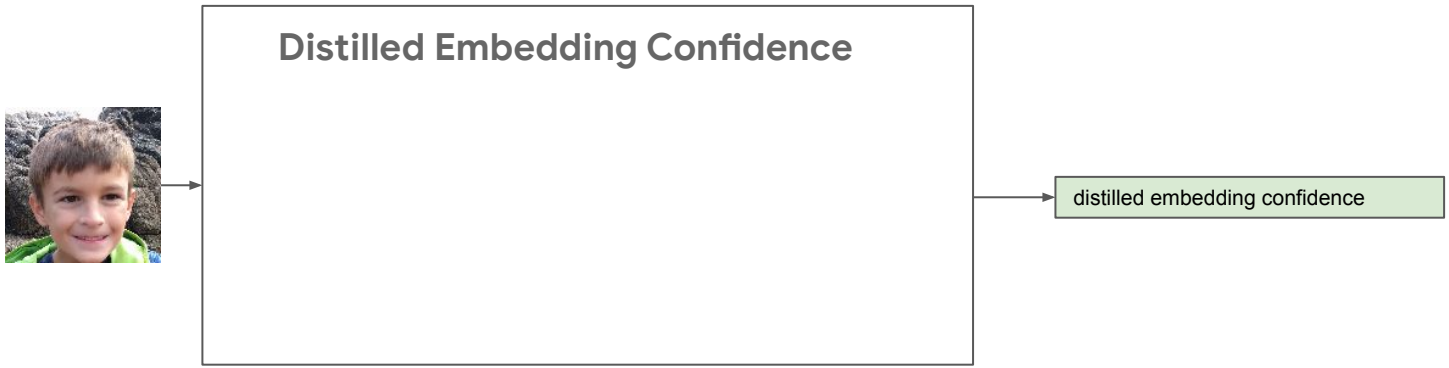


Staying close to home ...




Embedding
feature extraction
penultimate embedding
L_2
normalize
FaceNet embedding
sigmoid
embedding confidence


Embedding
feature extraction
penultimate embedding
L_2
normalize
FaceNet embedding
sigmoid
embedding confidence


Distilled Embedding Confidence
distilled embedding confidence

0.935 0.865 0.630 0.618 0.457 0.398 0.320 0.270 0.164 0.123 0.086

Recognizable

Not Recognizable

Introduction

Embeddings

Uncertainty Representations (2)

Discussion

Real-world applications

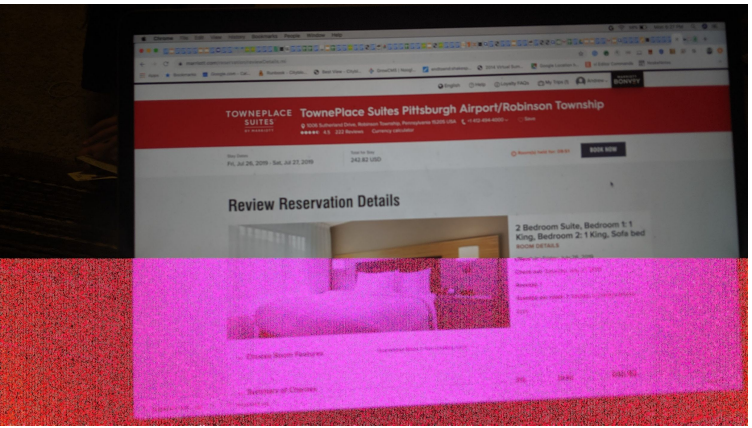

11:00
62°


9:56
Turn your face
Cancel

A boring image.

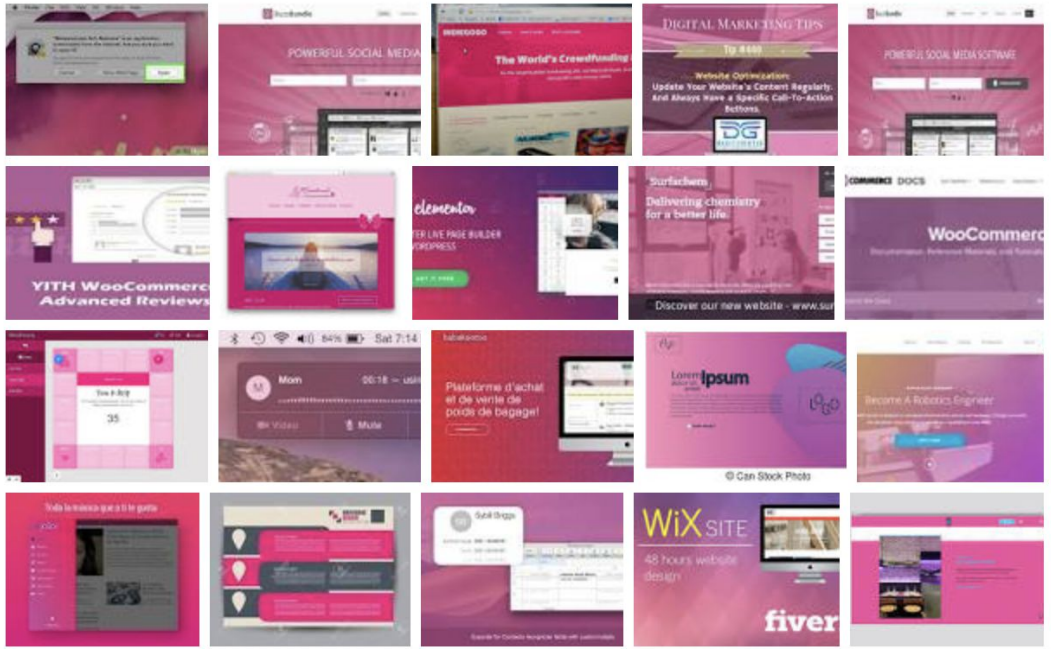
Visually similar images

Report images



A corrupted image.

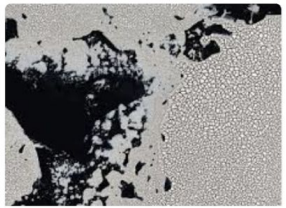
Visually similar images



Google Lens

Related results

SIMILAR IMAGES



Applications

Gallery Selection: Given a potentially large collection of images or a video. Select a subset of images/frames that will most reliably identify the individual.

Computational Efficiency: Prevent running the expensive models on images with high uncertainty.

Tracking: Use as the confidence score.

Offline Face Clustering: Confidence weighted cluster similarity instead of top-N or other heuristic based approaches.

Open Questions

Open Questions

How best to quantify and measure uncertainty?

Open Questions

Tell us when you don’t know.

But only when absolutely necessary.

Open Questions

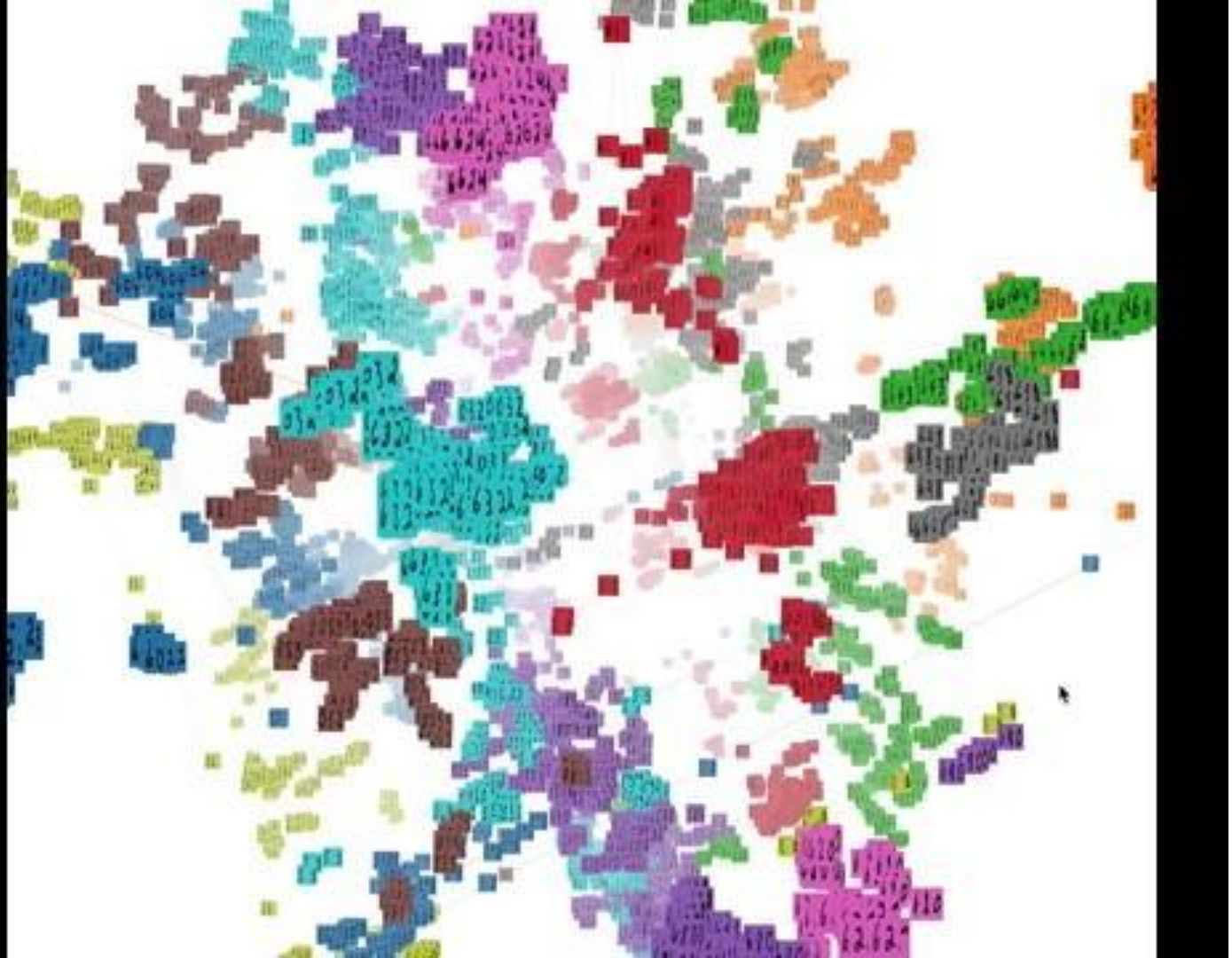
How best to compute, index, and match uncertain embeddings (at scale)?

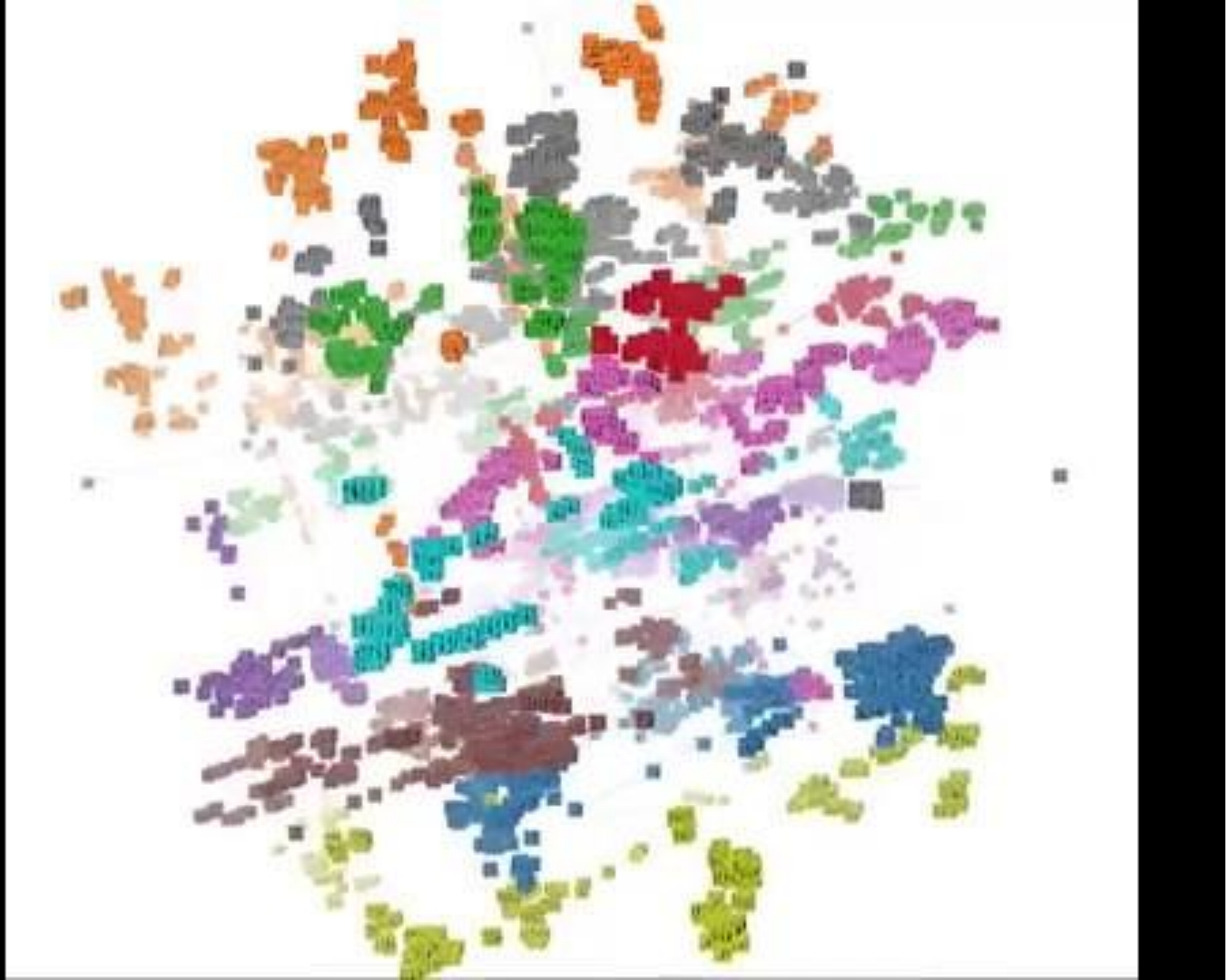
Open Questions

What network architectures encourage learning uncertainty?

Open Questions

Open world (epistemic) uncertainty.





Thank you