Similarity Learning

Leila Malihi

Headline

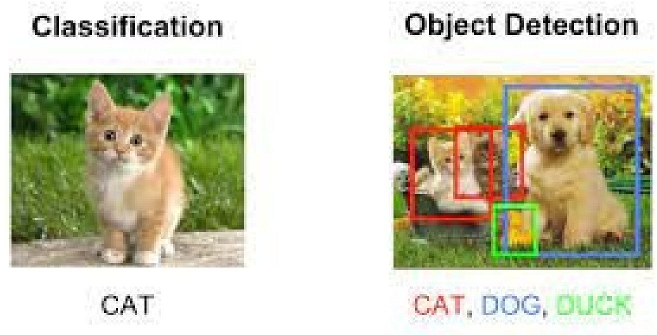
1-Why similarity learning

2- Siamese neural network

3- Triplet loss

4- Applications

Problem of Classification



Problem of Classification

A

Classification: person, face, male

B

Classification: person, face, male

Problem of Classification

- The question that a classification problem cannot answer is :

  Is this the same person?

A

B

- Comparison
- Ranking

Application of similarity learning

1- Unlocking cell phones with face

Application of similarity learning

1- Unlocking cell phones with face

Training Set

Application of similarity learning

1- Unlocking cell phones with face

A

yes

B

No

Application of similarity learning

2- Detect the students for exam

student1

student2

What is the problem?

- Retrain the model every time a new student register!!!!
- Can I actually train only one model and use it every year for the purpose of face recognition?

Similarity function

Low similarity score

High similarity score

Similarity function

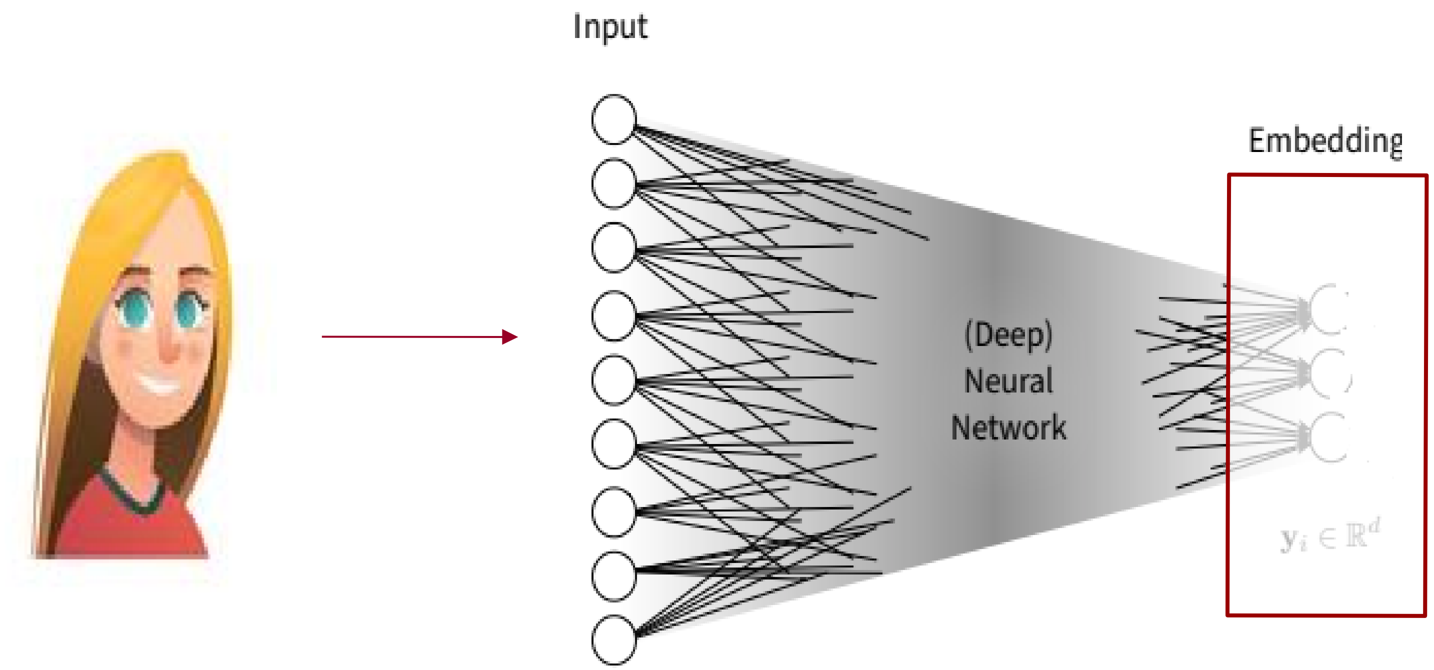
Distance(A,B) > Threshold → Different Person

Similarity function

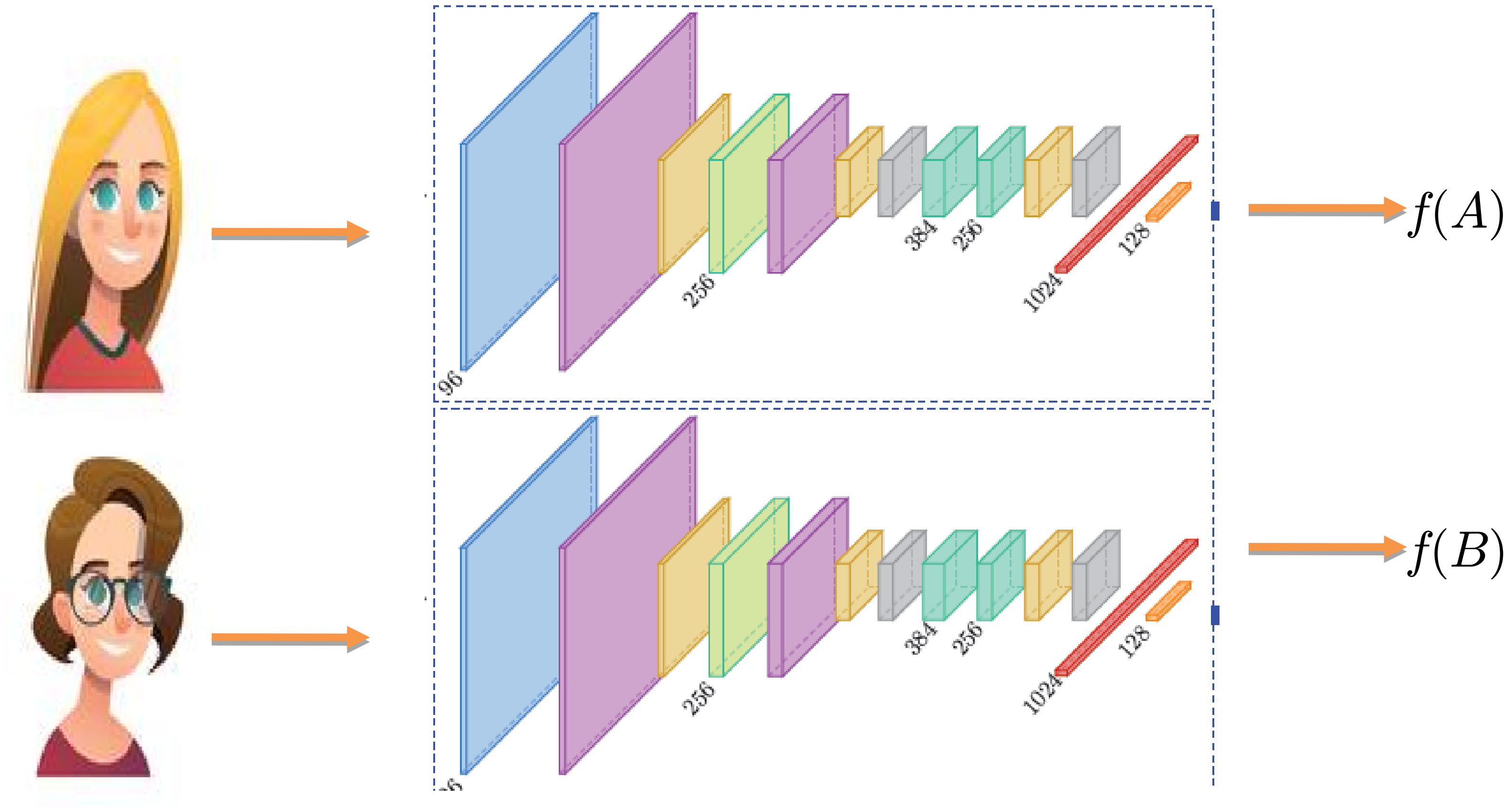
$$\text{Distance}(A,B) < \text{Threshold}$$

→ Same Person

Siamese Neural Networks



Siamese Neural Networks



Siamese Neural Networks

- The same network is used to obtain an encoding of the image A and to obtain an encoding of the image B
- Compare these two encodings

Siamese Neural Networks

- $d(A,B) = \|f(A) - f(B)\|$
- If A and B are same d(A,B) is small
- If A and B are different d(A,B) is large

Loss function for positive pair

- A & B are the same person

$$Loss = ||f(A) - f(B)||^2$$

Loss function for negative pair

- Use a Hinge loss
- A & B are different person

$$\max(0, m^2 - ||f(A) - f(B)||^2)$$

UNIVERSITÄT OSNABRÜCK

Contrastive loss:

$$y^*\|f(A)-f(B)\|^2+(1-y^*)max(0,m^2-\|f(A)-f(B)\|^2)$$

Positive pair

Negative pair

Train the Siamese networks

- Update the weights for each channel and then average them
- **Contrastive loss:**

  Bring the positive pairs together and negative pairs apart

Triplet loss

Anchor (A)

Positive (P)

Negative (N)

The goal: $$||f(A) - f(p)||^2 < ||f(A) - f(N)||^2$$

Learn Ranking with Triplet loss

$$||f(A) - f(p)||^2 < ||f(A) - f(N)||^2$$

$$||f(A) - f(p)||^2 - ||f(A) - f(N)||^2 < 0$$

$$||f(A) - f(p)||^2 - ||f(A) - f(N)||^2 + margin < 0$$

$$L(A,P,N) =$$

$$\max(||f(A) - f(p)||^2 - ||f(A) - f(N)||^2 + margin, 0)$$

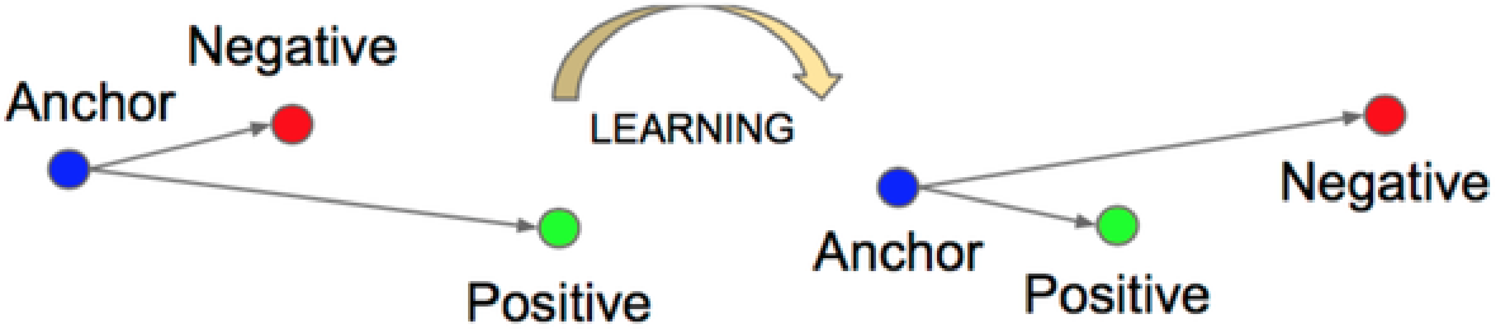
Hard cases

$$L(A,P,N)=$$

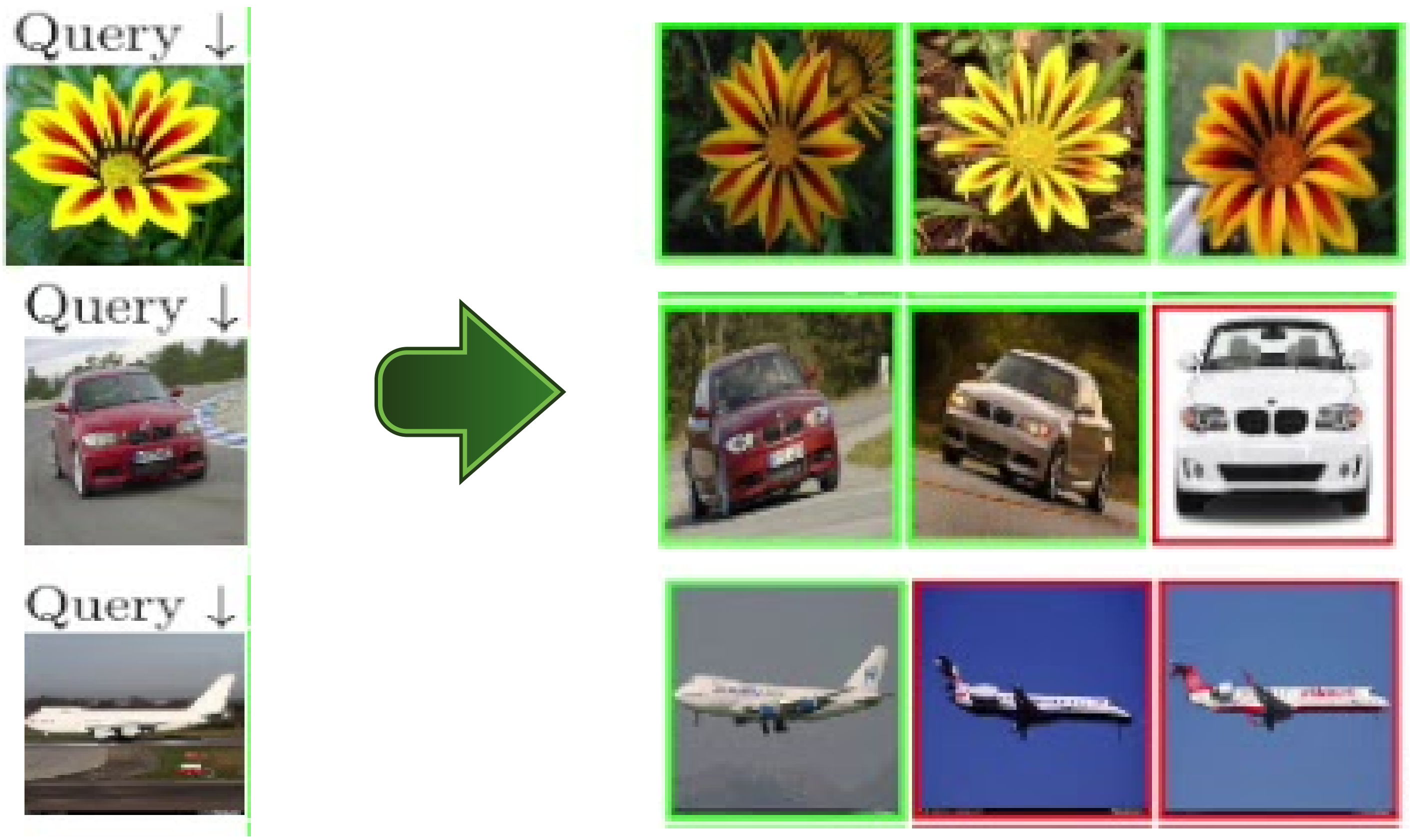
$$\max(||f(A)-f(p)||^2 - ||f(A)-f(N)||^2 + margin, 0)$$

distance(A, P) ~ distance(A,N)

Triplet loss



Nearest neighbor search



Challenges

- Random triplet loss does not work
- The number of possible triplets is huge, So the network should be trained in a long time

Improve Similarity learning

- **Improve the loss**
- **Sampling:**
  Choose the best triplet to train your neural network with
- **Ensembles:**
  Instead of making all decision with one neural network, use several networks and trained with a subset of triplets
- **Use a classification loss for similarity learning**

Articles

- Jian **Wang, et al.,** Deep Metric Learning with Angular Loss, 2017.

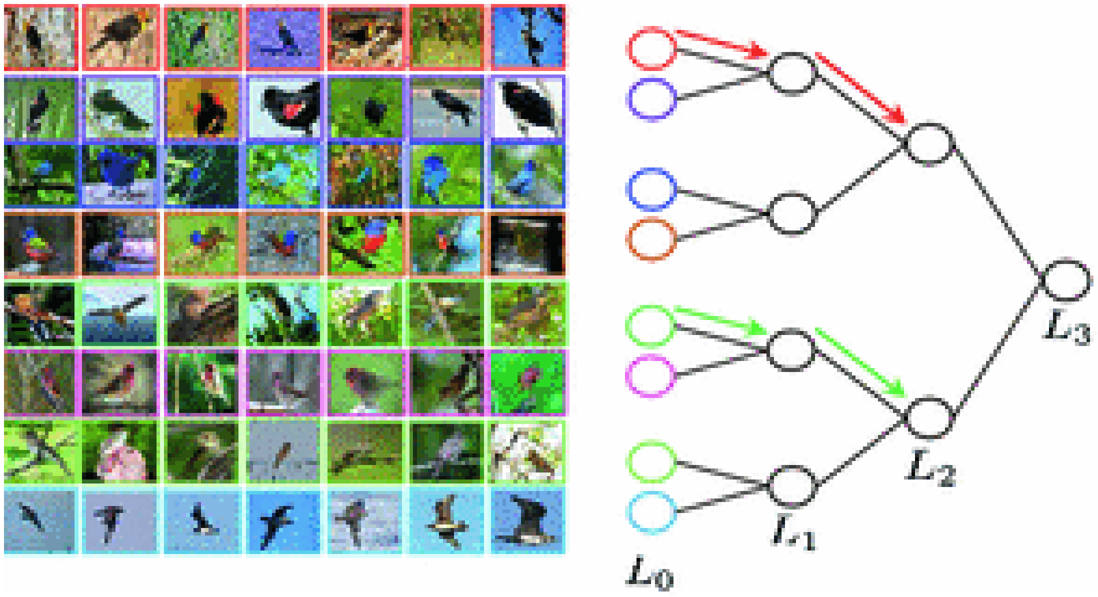
  (propose a novel angular loss, which takes angle relationship into account, for learning better similarity metric)

- Yu et al., Correcting the triplet selection bias for triplet loss, 2018.

  (propose a new variant of triplet loss, which tries to reduce the bias in triplet selection by adaptively correcting the distribution shift on the selected triplets)

Sampling Method: Hierarchical Triplet loss

- Leave of the tree= image classes



Weifeng Ge et al. Deep Metric Learning with Hierarchical Triplet Loss, ECCV 2018.

Tree creation

- *Create the tree: define a distance between classes*
- *If the distance is small, they merge in the next step*

$$D(p,q)= \frac{1}{n_p.n_q} \sum_{i\epsilon\, p, j\, \epsilon q} ||r_i - r_j||\ ^2$$

The number of sample for each class

Deep feature for image I and j

How to find the anchor?

- Select L' nodes at the 0 level, why?

  To preserve class diversity in the mini-batch

  - *Select M-1 classes at the 0 level for each L' nodes based on the distance in the feature space*

- Number of images in the mini-batch= T*M*L' images

Loss function

$$Loss = \frac{1}{Z_m} \sum_{T \in Tm} \left[\left| \left| x_a^z - x_p^z \right| \right| - \left| \left| x_a^z - x_n^z \right| \right| + \alpha_z \right.$$

All the triplets

Margin, it's going to adapt to the differences of the samples within the classes.

Sampling articles

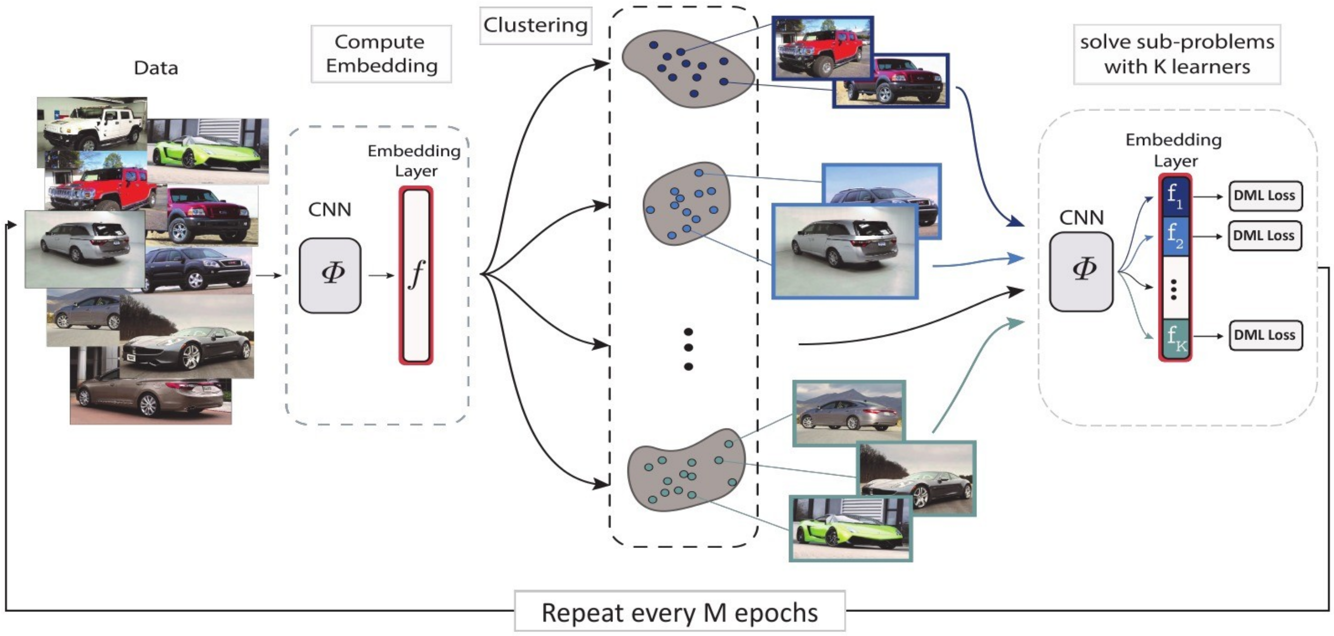
- Manmatha et al., sampling matter for deep metric learning, (ICC 2017) – original sampling method

(propose distance weighted sampling, which selects more informative and stable examples than traditional approaches and show that a simple margin based loss is sufficient to outperform all other loss functions.)

- Wang et al., Multi-similarity loss with general pair weighting for deep metric learning(CVPR 2019)

(A family of loss functions built on pair-based computation have been proposed in the literature which provide a myriad of solutions for deep metric learning. In this paper, they provide a general weighting framework for understanding recent pair-based loss functions)

Ensembles



Sanakoyeu et al. Divide and Conquer the Embedding Space for Metric Learning,CVPR 2019

Ensembles

- Cluster the embedding space in K clusters using K-means
- ***Divide:*** build k independent learners at the top of CNN
- K different sets of fully connected layers
- ***Conquer:*** use all the learners at the same time and fine tune our network with all the training set

Ensemble articles

- Manmatha et al., sampling matter for deep metric learning, (ICC 2017)
- Xu et al., Deep asymmetric metric learning via rich relationship mining, (CVPR 2019)
- Wang et al., Multi-similarity loss with general pair weighting for deep metric learning(CVPR 2019)

Classification loss articles

- Teh et al., ProxyNCA ++: Revisiting and Revitalizing Proxy neighborhood component analysis, arXiv 2020.
- Elezi et al., The group loss for deep metric learning, arXiv 2020.

(Propose Group Loss, a loss function based on a differentiable label-propagation method that enforces embedding similarity across all samples of a group)

- Qian et al., SoftTriple Loss: deep metric learning without triplet sampling, ICCV 2019.

(propose the SoftTriple loss to extend the SoftMax loss with multiple centers for each class)

How to choose a model?

Table 5. Accuracy on Cars196

	Concatenated (512-dim)			Separated (128-dim)		
	P@1	**RP**	**MAP@R**	**P@1**	**RP**	**MAP@R**
Pretrained	46.89	13.77	5.91	43.27	13.37	5.64
Contrastive	81.78 ± 0.43	35.11 ± 0.45	24.89 ± 0.50	69.80 ± 0.38	27.78 ± 0.34	17.24 ± 0.35
Triplet	79.13 ± 0.42	33.71 ± 0.45	23.02 ± 0.51	65.68 ± 0.58	26.67 ± 0.36	15.82 ± 0.36
NT-Xent	80.99 ± 0.54	34.96 ± 0.38	24.40 ± 0.41	68.16 ± 0.36	27.66 ± 0.23	16.78 ± 0.24
ProxyNCA	83.56 ± 0.27	35.62 ± 0.28	25.38 ± 0.31	73.46 ± 0.23	28.90 ± 0.22	18.29 ± 0.22
Margin	81.16 ± 0.50	34.82 ± 0.31	24.21 ± 0.34	68.24 ± 0.35	27.25 ± 0.19	16.40 ± 0.20
Margin / class	80.04 ± 0.61	33.78 ± 0.51	23.11 ± 0.55	67.54 ± 0.60	26.68 ± 0.40	15.88 ± 0.39
N. Softmax	83.16 ± 0.25	36.20 ± 0.26	26.00 ± 0.30	72.55 ± 0.18	29.35 ± 0.20	18.73 ± 0.20
CosFace	$\mathbf{85.52 \pm 0.24}$	37.32 ± 0.28	27.57 ± 0.30	$\mathbf{74.67 \pm 0.20}$	29.01 ± 0.11	18.80 ± 0.12
ArcFace	85.44 ± 0.28	37.02 ± 0.29	27.22 ± 0.30	72.10 ± 0.37	27.29 ± 0.17	17.11 ± 0.18
FastAP	78.45 ± 0.52	33.61 ± 0.54	23.14 ± 0.56	65.08 ± 0.36	26.59 ± 0.36	15.94 ± 0.34
SNR	82.02 ± 0.48	35.22 ± 0.43	25.03 ± 0.48	69.69 ± 0.46	27.55 ± 0.25	17.13 ± 0.26
MS	85.14 ± 0.29	$\mathbf{38.09 \pm 0.19}$	$\mathbf{28.07 \pm 0.22}$	73.77 ± 0.19	$\mathbf{29.92 \pm 0.16}$	$\mathbf{19.32 \pm 0.18}$
MS+Miner	83.67 ± 0.34	37.08 ± 0.31	27.01 ± 0.35	71.80 ± 0.22	29.44 ± 0.21	18.86 ± 0.20
SoftTriple	84.49 ± 0.26	37.03 ± 0.21	27.08 ± 0.21	73.69 ± 0.21	29.29 ± 0.16	18.89 ± 0.16

Musgrave et al. A Metric Learning Reality Check paper arXiv 2020

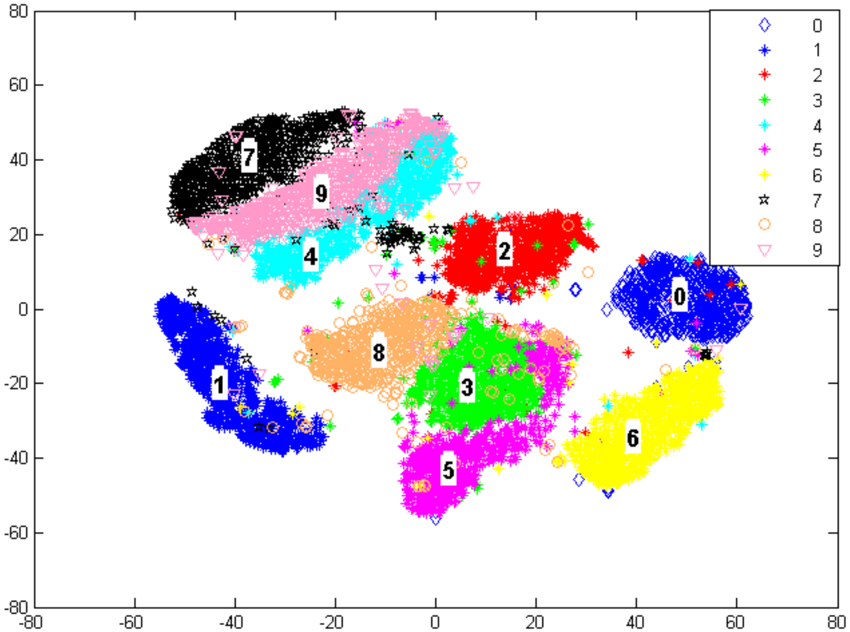
Points

- Use the simple baseline: contrastive loss , triplet loss, and classification loss

- Freezing batch-norm layers, using multiple centers per class
- Naive ensembles, copying your own network three times instead of one training with different triplets
- Two good out-of-the-box choices:One is proxy-NCA, the other one is soft-triplet loss

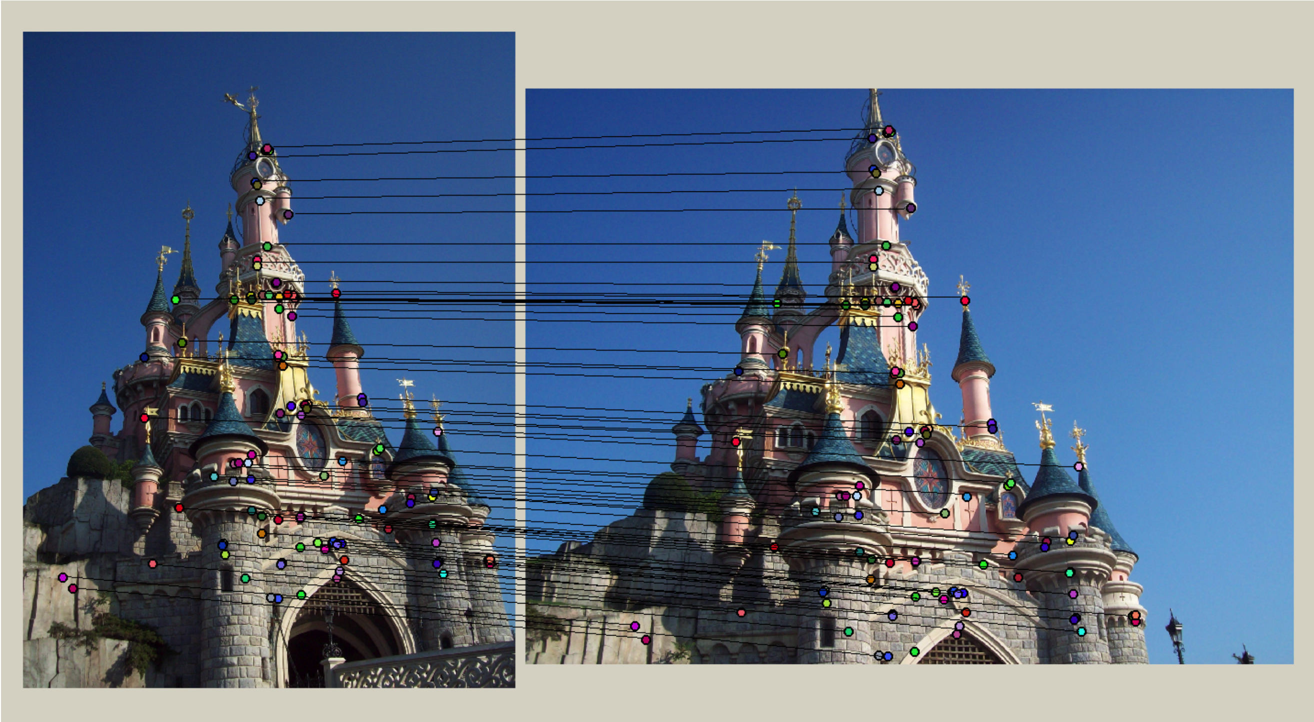
Applications

- Clustering on MNIST



Applications

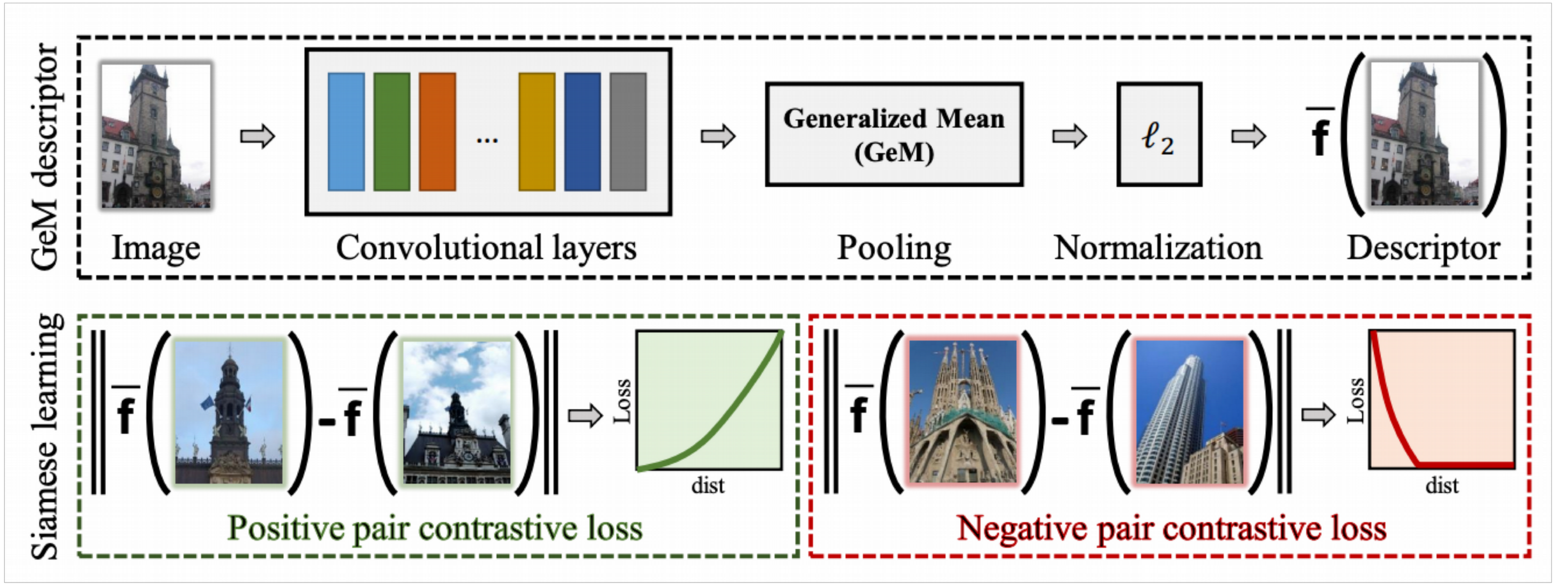
- Establishing image correspondences



Applications

- Establishing image correspondences

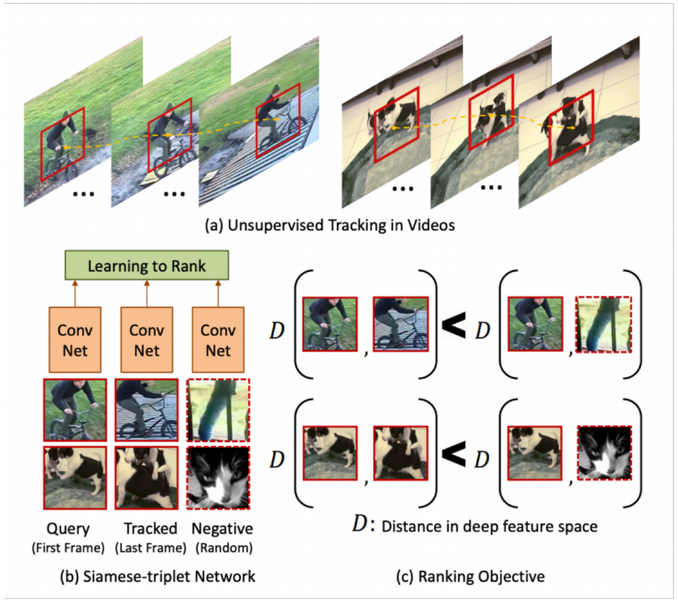
Image Retrieval



Radenovic et al. Fine-tuning CNN Image Retrieval with No Human Annotation“. TPAMI 2018

Application: Unsupervised learning

- Tracking provides the supervision
- Use those as positive samples
- Extract random patches as negative samples



Wang and Gupta. Unsupervised Learning of Visual Representations using Videos“. ICCV 2015

Question?