Siamese Network： Architecture and Applications in Computer Vision

Tech Report

Dec 30, 2014

Hengliang Luo

Outline

- Metric Learning
- Siamese Architecture
- Siamese Network: Applications in computer vision
- Triplet Network
- Conclusion

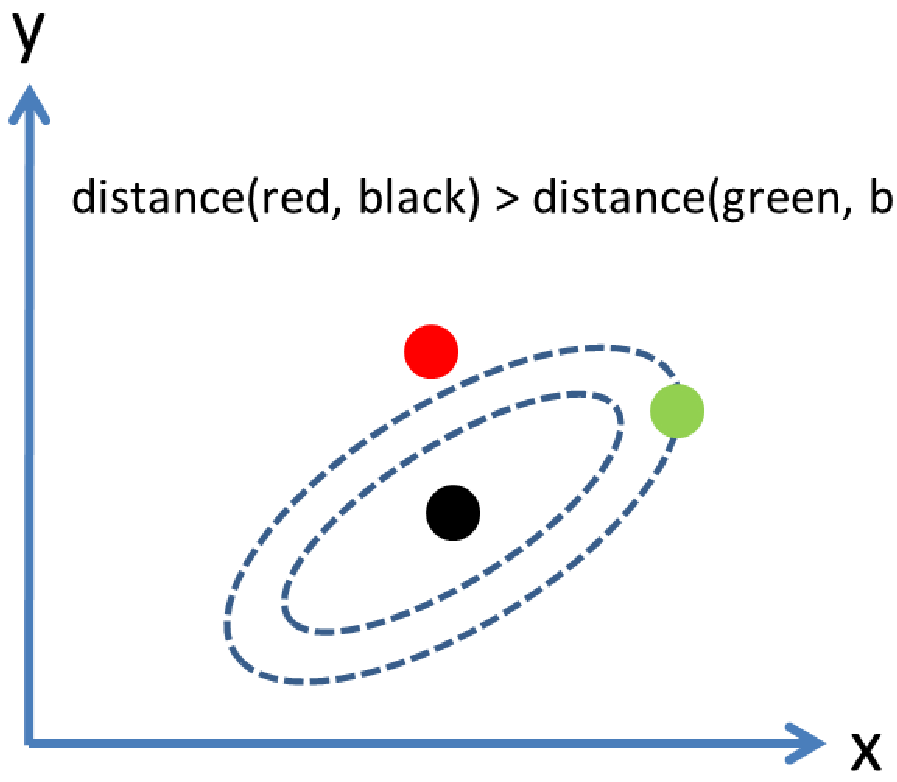
Siamese

- Someone or something from Thailand:
  - The Thai language, The Thai people
- Siamese, an informal term for conjoined or fused:
  - Siamese twins, conjoined twins
  - Siamesing (engineering), the practice, whose name is derived from siamese twins, of combining two devices (such as cylinder ports or cooling jackets) together into a closely coupled pair, so as to save space between them.

http://en.wikipedia.org/wiki/Siamese

Metric Learning

- Euclidean distance *vs* Mahalanobis distance



http://horicky.blogspot.jp/2012/08/measuring-similarity-and-distance.html

Metric Learning

Mahalanobis Distance Metric Learning

- *Euclidean distance*
- *Mahalanobis distance* $d(\vec{x},\vec{y}) = \sqrt{(\vec{x}-\vec{y})^T S^{-1}(\vec{x}-\vec{y})}.$
- *Mahalanobis Distance Metric Learning*

$$d(x,y) = d_A(x,y) = ||x-y||_A = \sqrt{(x-y)^T A(x-y)}$$

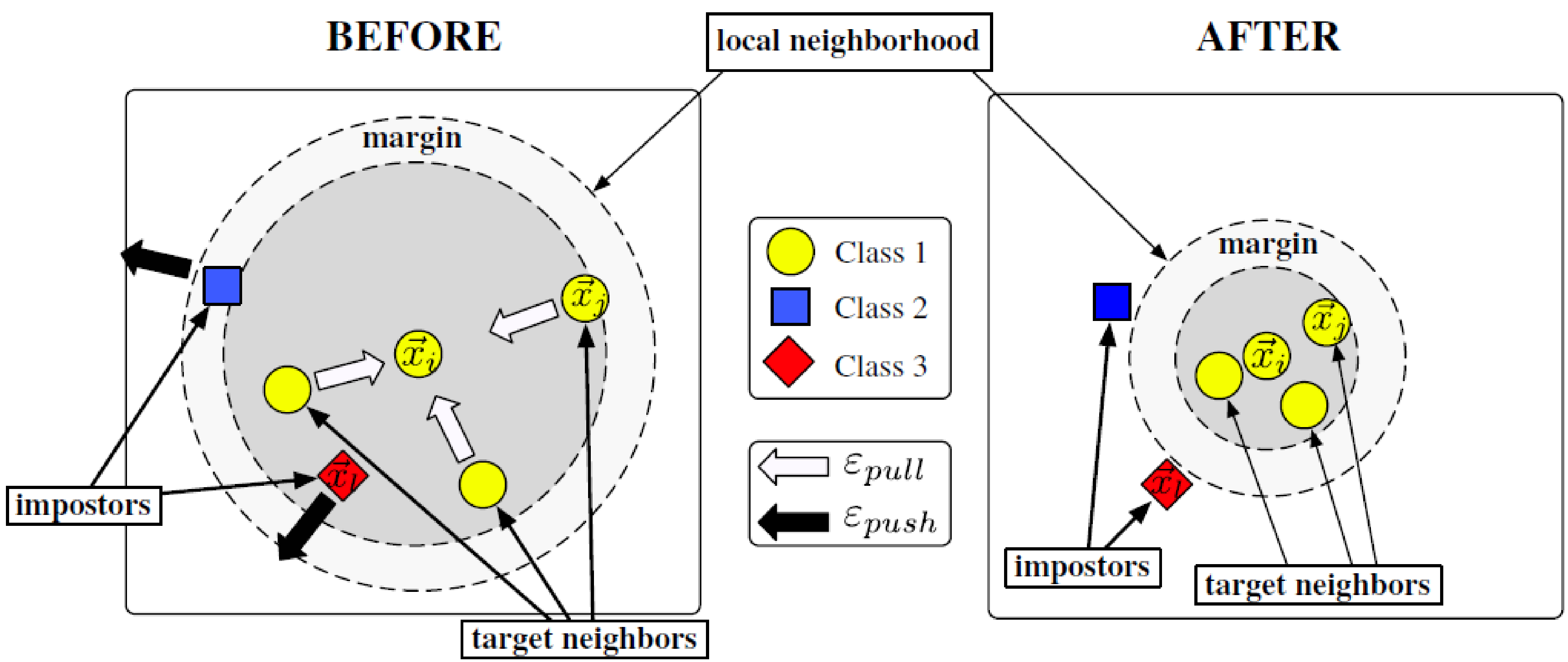
$$\begin{array}{ll} \min\limits_{A} & \sum_{(x_i,x_j)\in\mathcal{S}} ||x_i - x_j||_A^2 \\ \text{s.t.} & \sum_{(x_i,x_j)\in\mathcal{D}} ||x_i - x_j||_A \geq 1 \\ & A \succeq 0 \end{array}$$

Xing E P, Jordan M I, Russell S, et al. Distance metric learning with application to clustering with side-information[C], NIPS2002: 505-512.

Metric Learning

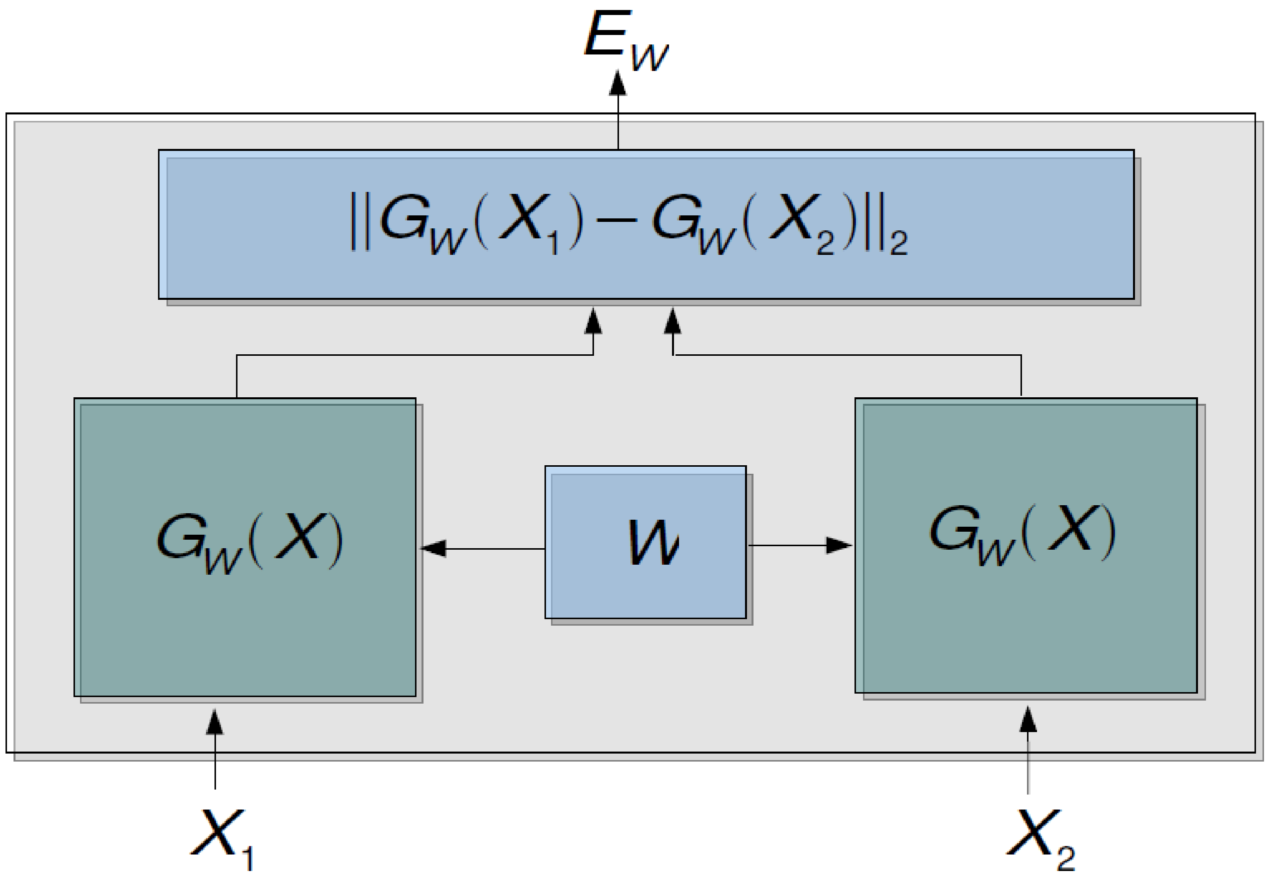
Large-Margin Nearest Neighbors(LMNN)

$$\min_{A \succeq 0} \sum_{(i,j)\in\mathcal{S}} d_A(\boldsymbol{x}_i, \boldsymbol{x}_j) + \lambda \sum_{(i,j,k)\in\mathcal{R}} [1 + d_A(\boldsymbol{x}_i, \boldsymbol{x}_j) - d_A(\boldsymbol{x}_i, \boldsymbol{x}_k)]_+$$



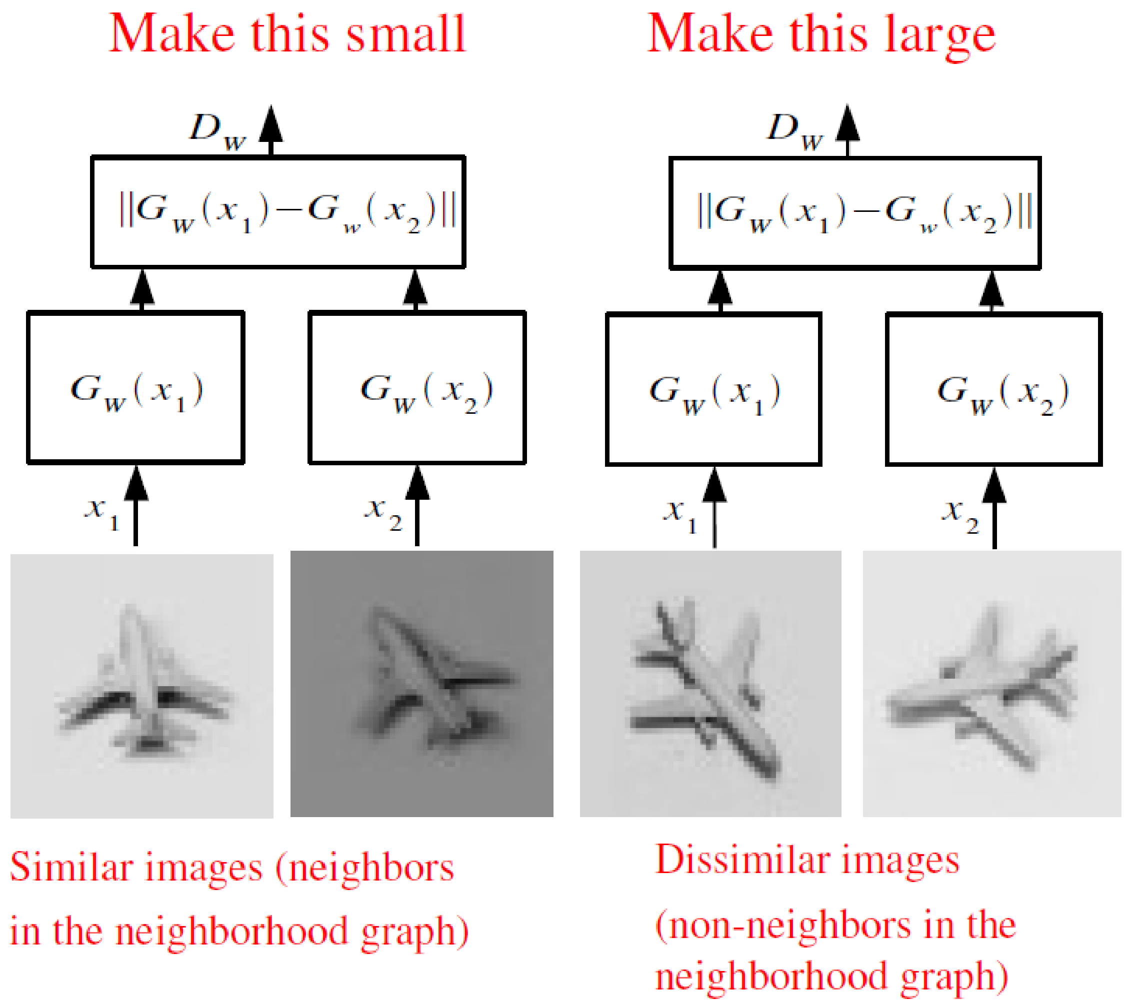
Brian Kulis, Metric Learning: A Survey. *web.cse.ohio-state.edu/~kulis/pubs/ftml_**metric_learning**.pdf*

Siamese Architecture



Learning Hierarchies of *Invariant Features*. Yann LeCun.
helper.ipam.ucla.edu/publications/gss2012/gss2012_10739.pdf

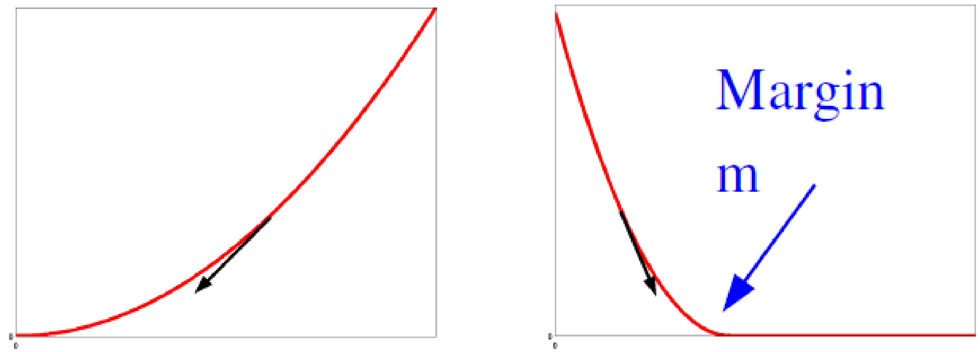
Siamese Architecture and loss function



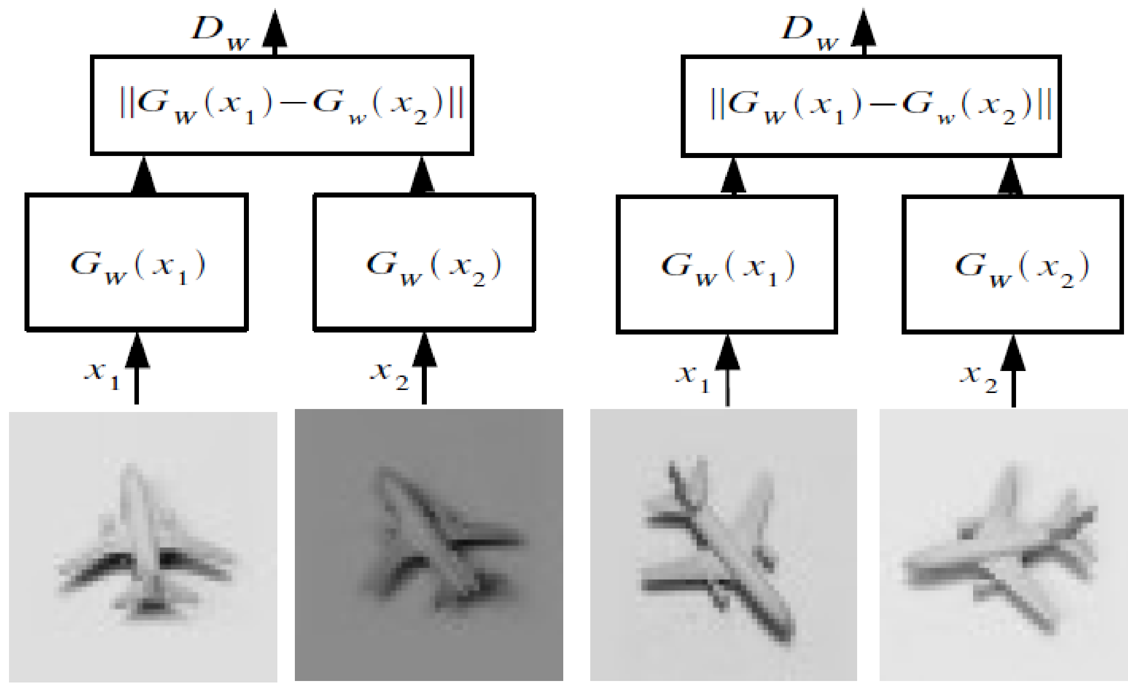
Learning Hierarchies of *Invariant Features*. Yann LeCun.
helper.ipam.ucla.edu/publications/gss2012/gss2012_10739.pdf

Loss function

$$L_{similar} = \frac{1}{2} D_w^2 \qquad L_{dissimilar} = \frac{1}{2} \{ max(0, m - D_W) \}^2$$



Hinge Loss

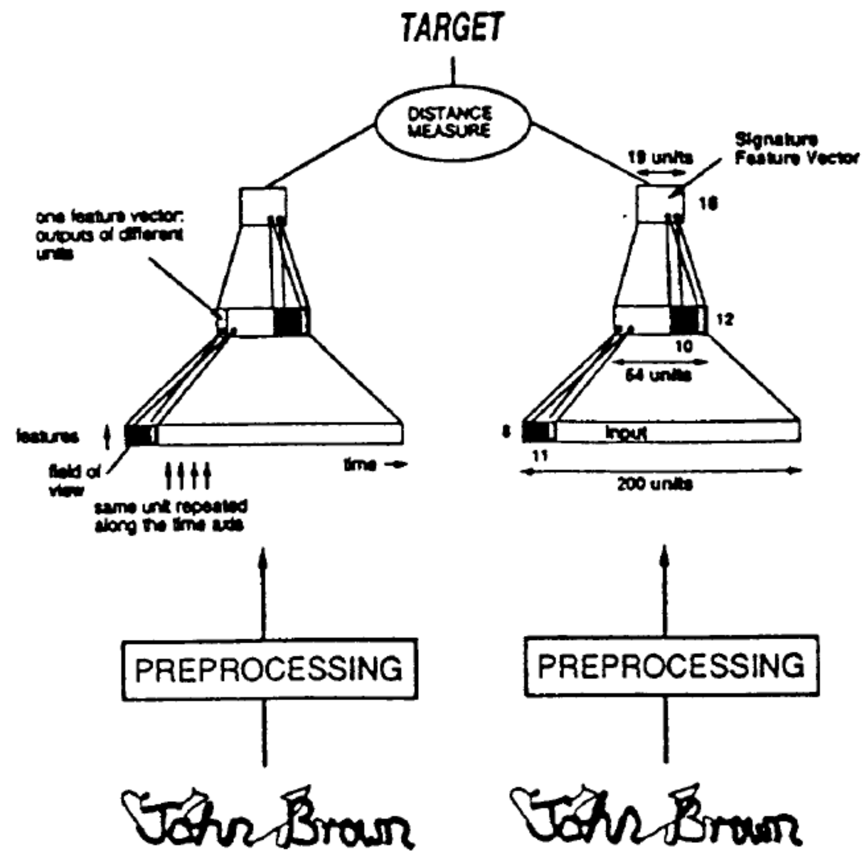


Learning Hierarchies of *Invariant Features*. Yann LeCun.
helper.ipam.ucla.edu/publications/gss2012/gss2012_10739.pdf

Siamese Network

Application in Signature Verification

- The input is 8(feature) x 200(time) units.
- The cosine distance was used, (1 for genuine pairs, -1 for forgery pairs)

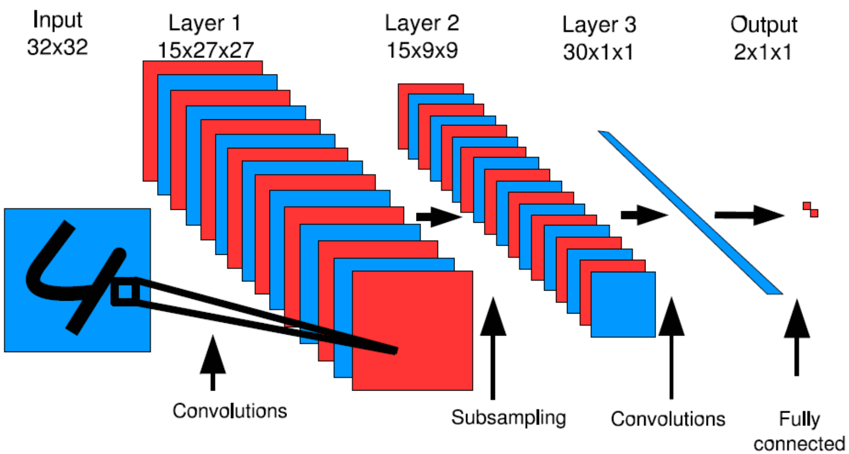


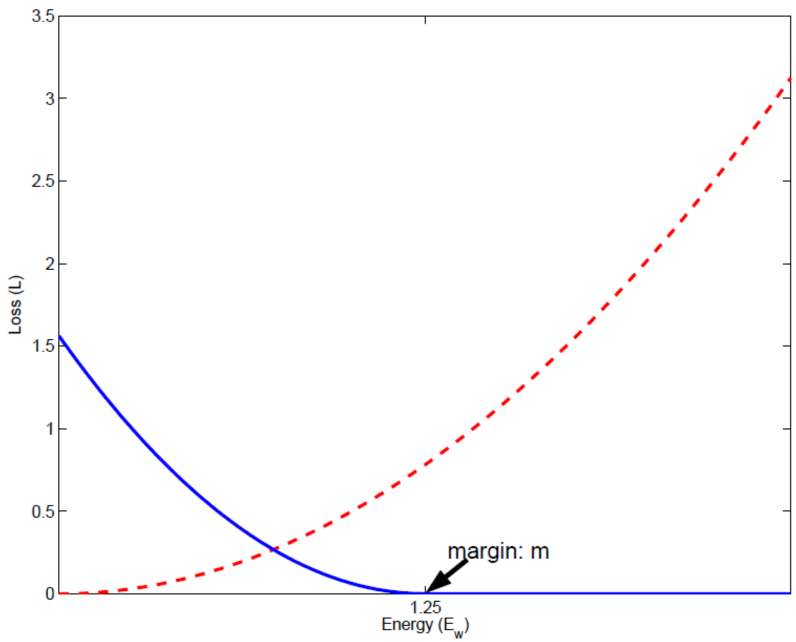
Bromley J, Guyon I, Lecun Y, et al. Signature Verification using a" Siamese" Time Delay Neural Network, NIPS Proc. 1994.

Siamese Network
Application in Dimensionality reduction

The exact loss function is

$$L(W, Y, \vec{X}_1, \vec{X}_2) =$$
$$(1-Y)\frac{1}{2}(D_W)^2 + (Y)\frac{1}{2}\{max(0, m - D_W)\}^2$$

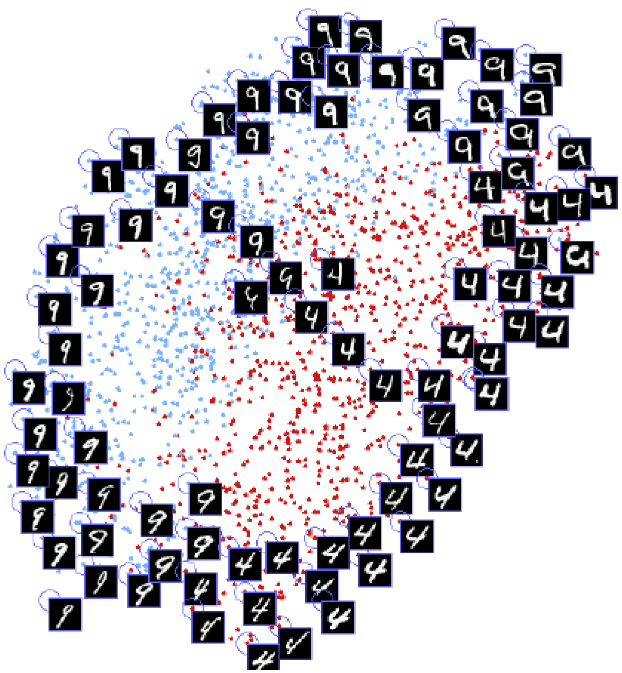




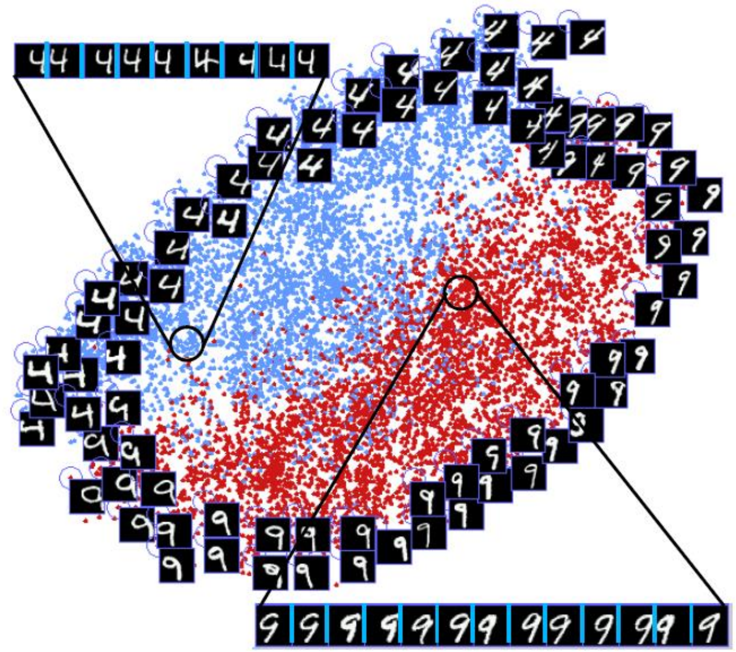
Hadsell R, Chopra S, LeCun Y. Dimensionality reduction by learning an invariant mapping, CVPR 2006

Siamese Network

Application in Dimensionality reduction



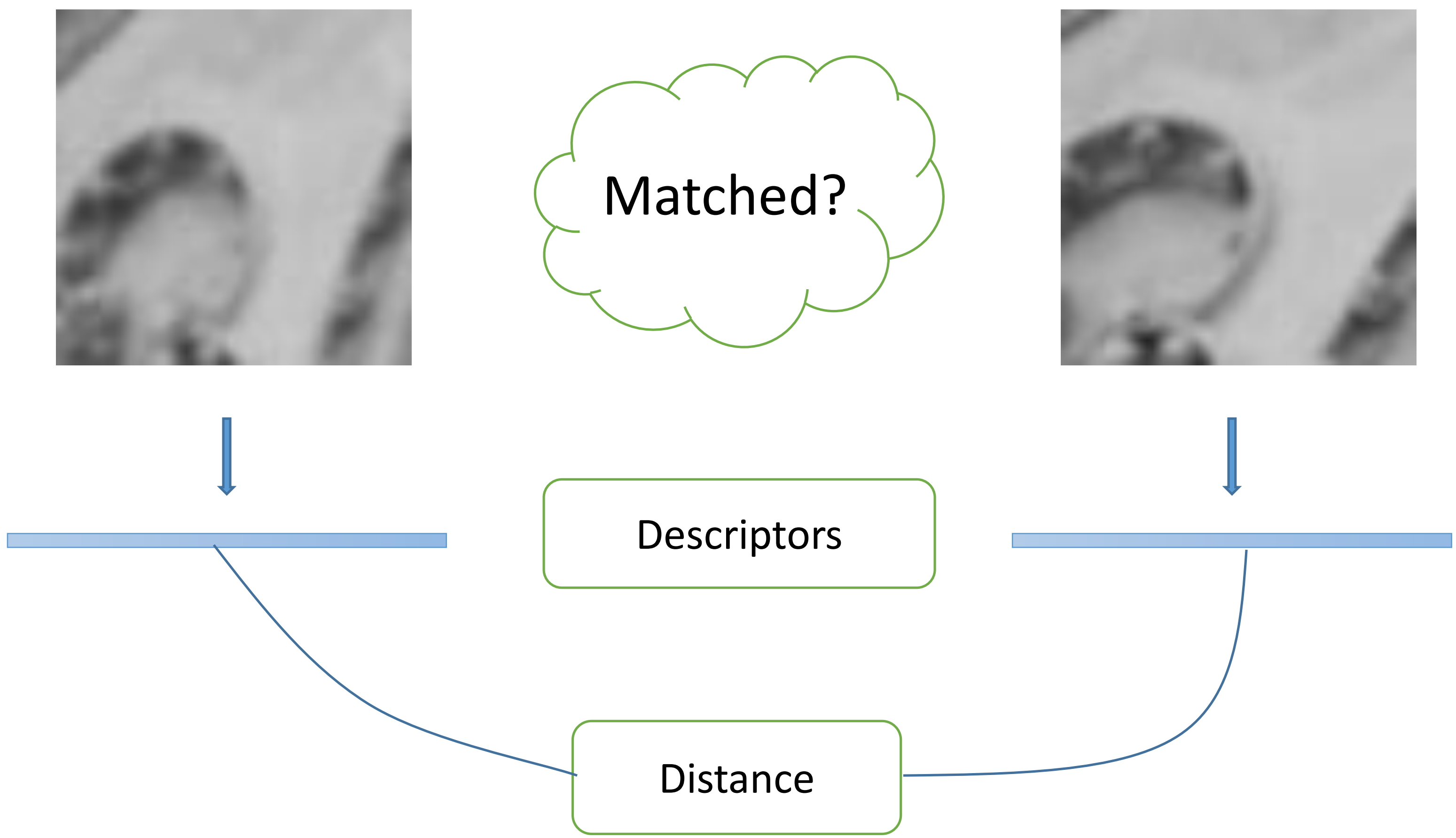
LearnedMapping of MNIST samples



Learning a Shift Invariant Mapping of MNIST samples

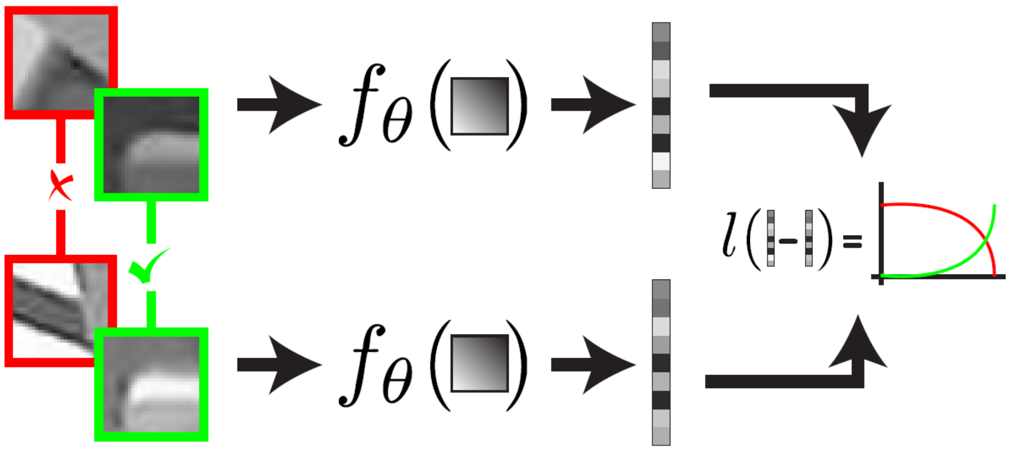
Hadsell R, Chopra S, LeCun Y. Dimensionality reduction by learning an invariant mapping, CVPR 2006

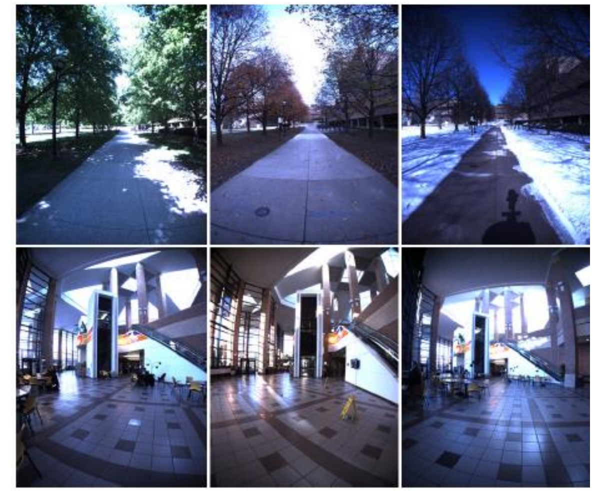
Image Descriptors



Siamese Network

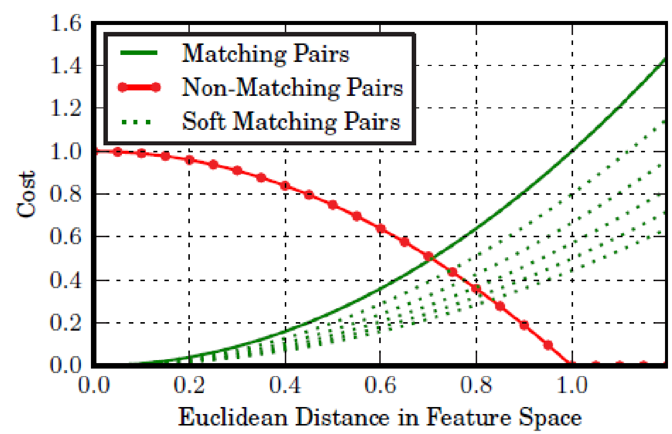
Application in Learning Image Descriptors (I)





Using the contrastive cost function

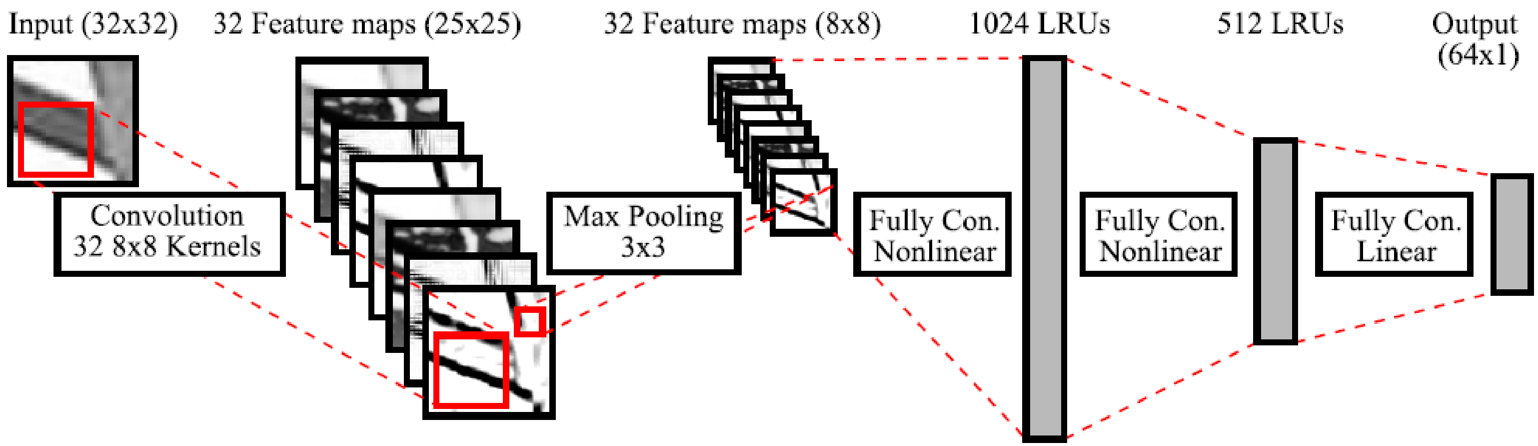
$$l_\theta(\mathbf{y}_i, \mathbf{y}_j) = \begin{cases} s_{ij} d_{ij}^2, & \text{if matching} \\ \max\left(1.0 - d_{ij}^2,\ 0\right), & \text{if non-matching} \end{cases}$$



Nicholas Carlevaris-Bianco and Ryan M. Eustice, Learning visual feature descriptors for dynamic lighting conditions. In Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, 2014

Siamese Network

Application in Learning Image Descriptors (I)



CNN Model

Nicholas Carlevaris-Bianco and Ryan M. Eustice, Learning visual feature descriptors for dynamic lighting conditions. In Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, 2014

Siamese Network

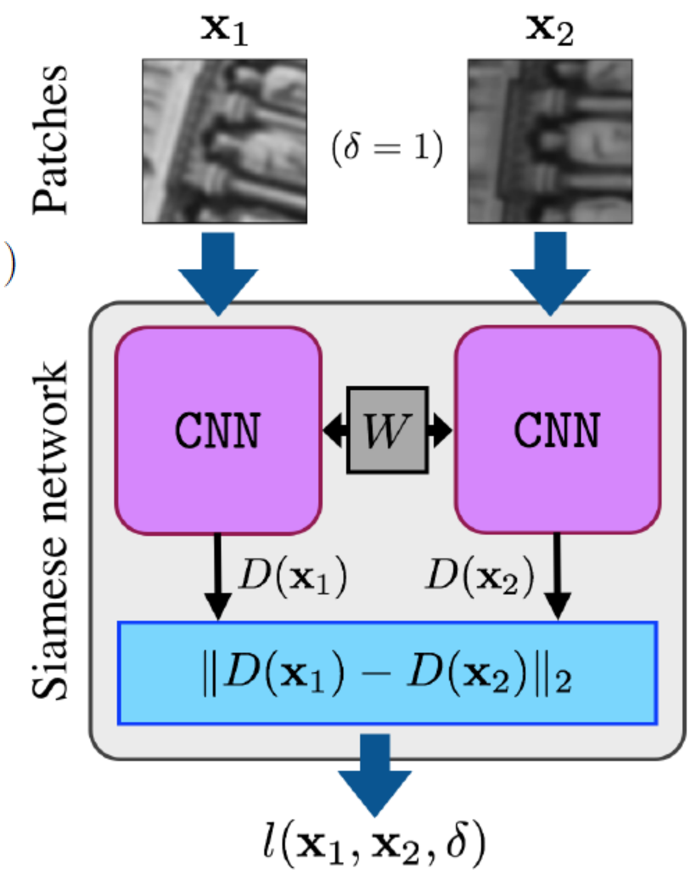
Application in Learning Image Descriptors (II)

$$d_D(\mathbf{x}_1, \mathbf{x}_2) = \|D(\mathbf{x}_1) - D(\mathbf{x}_2)\|_2$$

$$l(\mathbf{x}_1, \mathbf{x}_2, \delta) = \delta \cdot l_P(d_D(\mathbf{x}_1, \mathbf{x}_2)) + (1-\delta) \cdot l_N(d_D(\mathbf{x}_1, \mathbf{x}_2))$$

$$l_P(d_D(\mathbf{x}_1, \mathbf{x}_2)) = d_D(\mathbf{x}_1, \mathbf{x}_2)$$

$$l_N(d_D(\mathbf{x}_1, \mathbf{x}_2)) = \max(0, m - d_D(\mathbf{x}_1, \mathbf{x}_2))$$

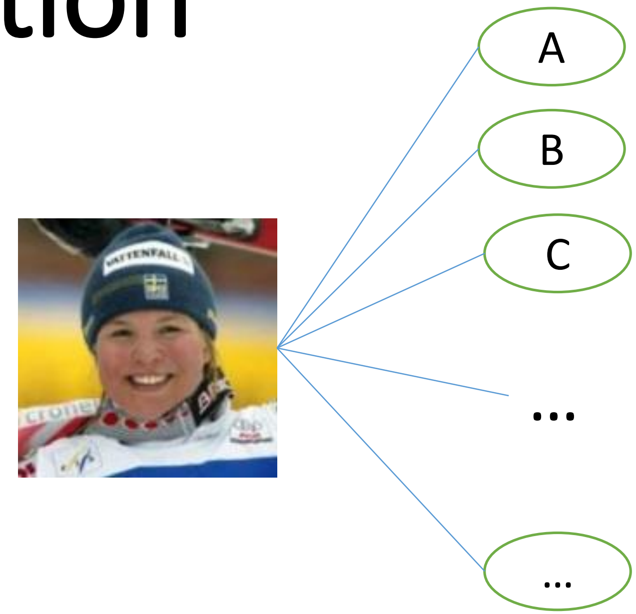


Fracking Deep Convolutional Image Descriptors, Under review as a conference paper at ICLR 2015, http://arxiv.org/abs/1412.6537
Convolutional Neural Networks learn compact local image descriptors, http://arxiv.org/abs/1304.7948

Face recognition

1. Face identification

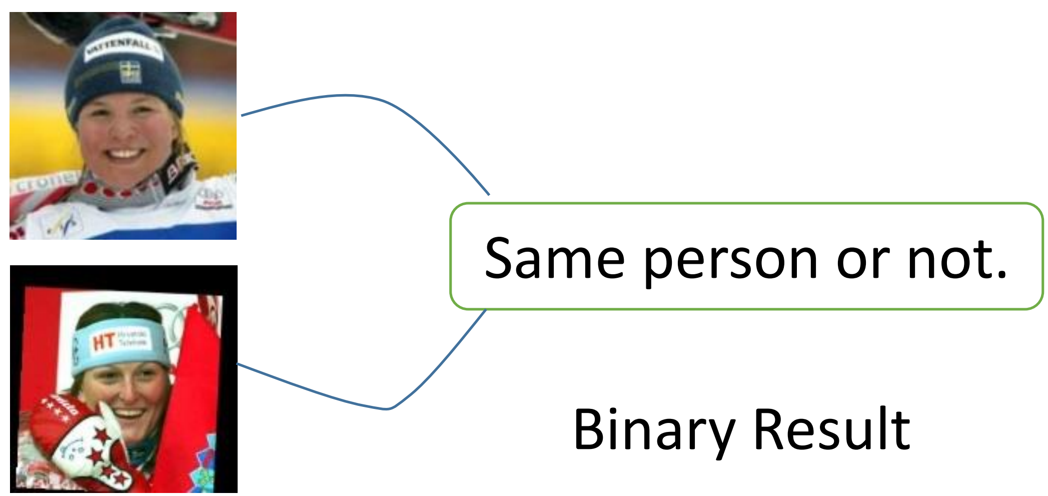
Who?



Multiclass classification

2. Face verification

Same?



Binary Result

http://vis-www.cs.umass.edu/lfw/person/Janica_Kostelic.html

Siamese Network
Application in face verification(Ⅰ)

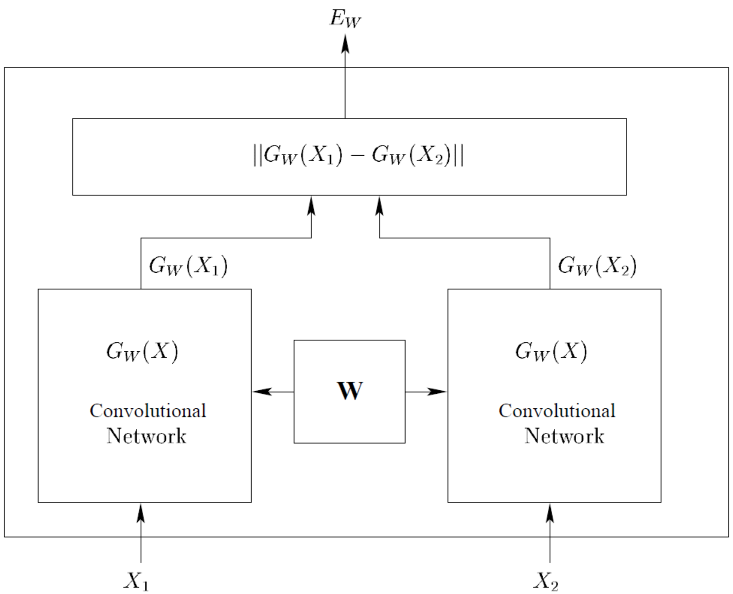
Let X_1 and X_2 be a pair of images shown to our learning machine. Let Y be a binary label of the pair, $Y = 0$ if the images X_1 and X_2 belong to the same person (a "genuine pair") and $Y = 1$ otherwise (an "impostor pair").

We assume that the loss function depends on the input and the parameters only indirectly through the energy. Our loss function is of the form:

$$\mathcal{L}(W) = \sum_{i=1}^{P} L(W, (Y, X_1, X_2)^i)$$

$$\begin{aligned} L(W, (Y, X_1, X_2)^i) &= (1 - Y) L_G \left(E_W(X_1, X_2)^i\right) \\ &+ Y L_I \left(E_W(X_1, X_2)^i\right) \\ &= (1 - Y)\frac{2}{Q}(E_W)^2 + (Y) 2Q\, e^{-\frac{2.77}{Q} E_W} \end{aligned}$$

$$E_W = \|G_W(X_1) - G_W(X_2)\|$$

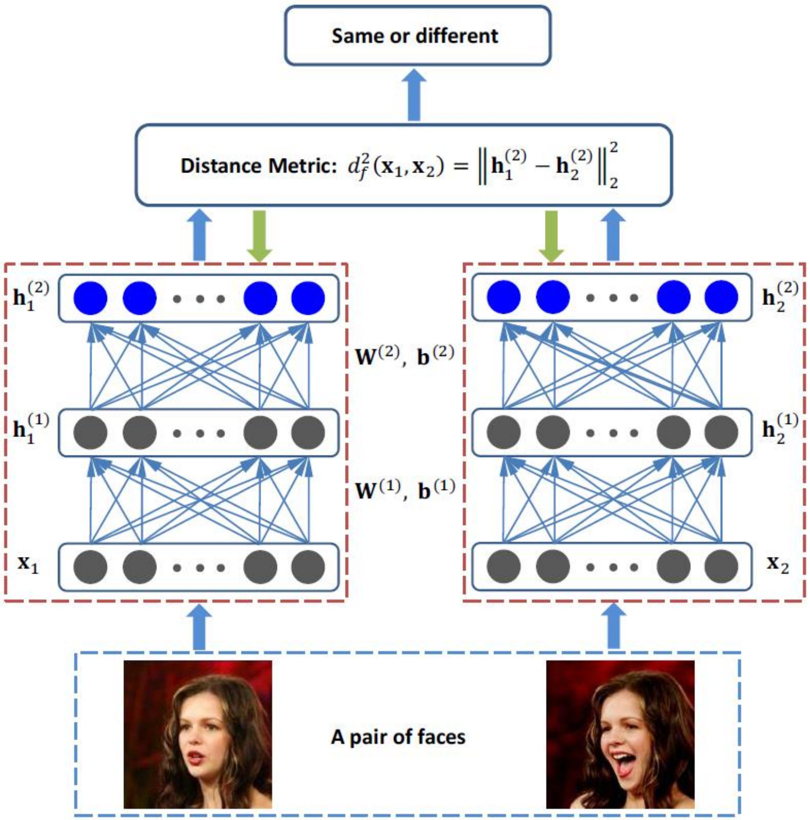


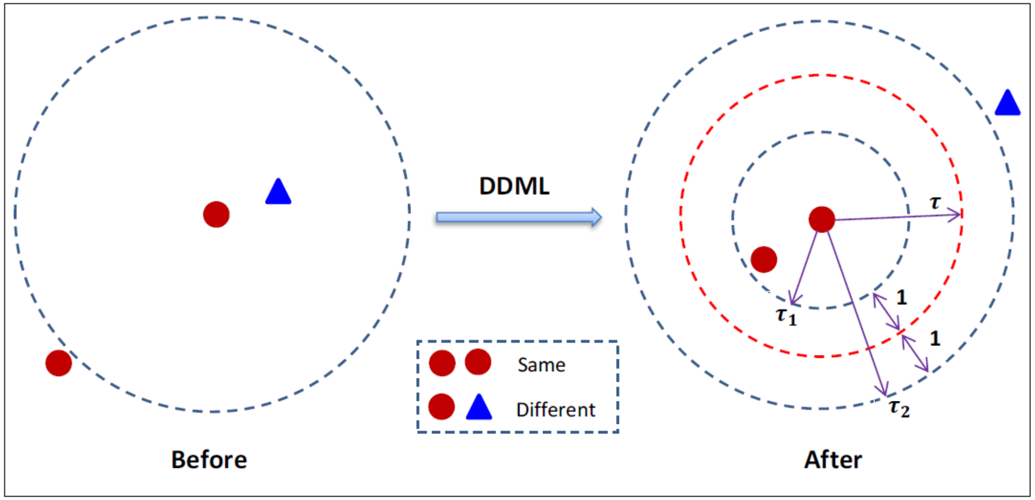
Chopra S, Hadsell R, LeCun Y. Learning a similarity metric discriminatively, with application to face verification, CVPR 2005

Siamese Network

Application in face verification(II)

LFW:90.68%





Intuitive illustration of the proposed DDML method

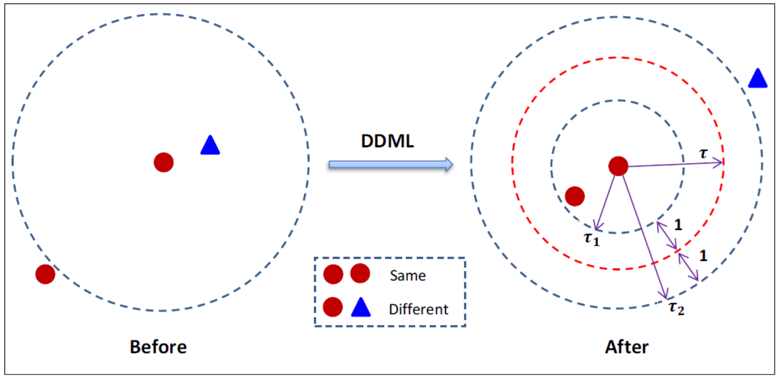
Junlin Hu, etc. Discriminative Deep Metric Learning for Face Verification in theWild, CVPR 2014

Siamese Network

Application in face verification(Ⅱ)

$$d_f^2(x_i, x_j) < \tau - 1, l_{ij} = 1$$
$$d_f^2(x_i, x_j) > \tau + 1, l_{ij} = -1$$

$$\ell_{ij}\left(\tau - d_f^2(\mathbf{x}_i, \mathbf{x}_j)\right) > 1$$

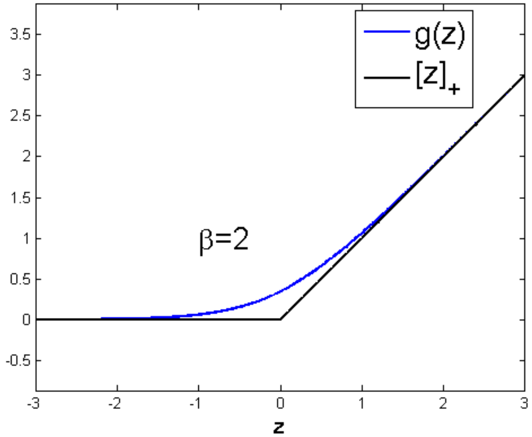


Intuitive illustration of the proposed DDML method

DDML as the following optimization problem:

$$\begin{aligned} \arg\min_f J &= J_1 + J_2 \\ &= \frac{1}{2}\sum_{i,j} g\Big(1 - \ell_{ij}\left(\tau - d_f^2(\mathbf{x}_i, \mathbf{x}_j)\right)\Big) \\ &+ \frac{\lambda}{2}\sum_{m=1}^{M}\left(\|\mathbf{W}^{(m)}\|_F^2 + \|\mathbf{b}^{(m)}\|_2^2\right) \end{aligned}$$

where $g(z) = \frac{1}{\beta}\log\left(1 + \exp(\beta z)\right)$ is the generalized logistic loss function [25], which is a smoothed approximation of the hinge loss function $[z]_+ = \max(z, 0)$

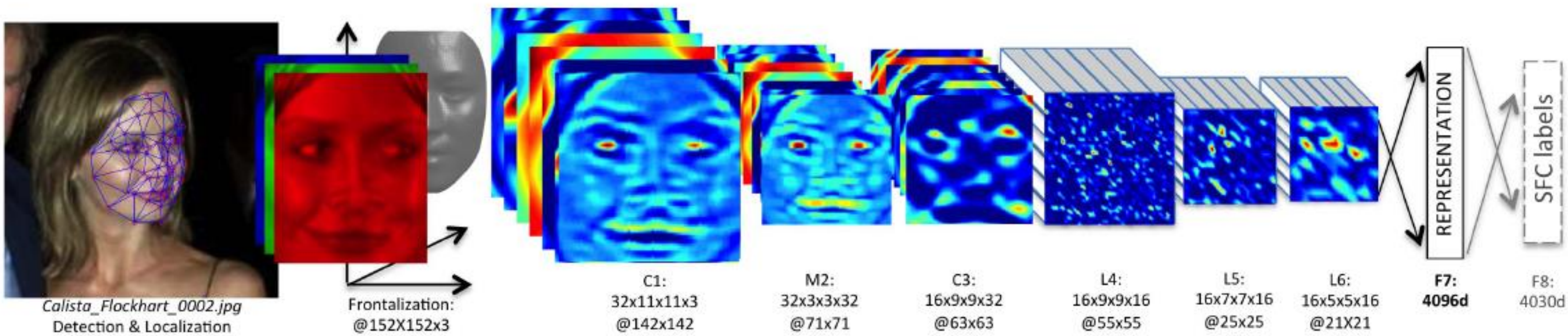


Junlin Hu, etc. Discriminative Deep Metric Learning for Face Verification in theWild, CVPR 2014

Classification Network

Application in face verification(IV)

LFW:97.35%



Verification Metric:

1)Cosine similarity

2) Weighted χ2 distance $\chi^2(f_1, f_2) = \sum_i w_i (f_1[i] - f_2[i])^2 / (f_1[i] + f_2[i])$

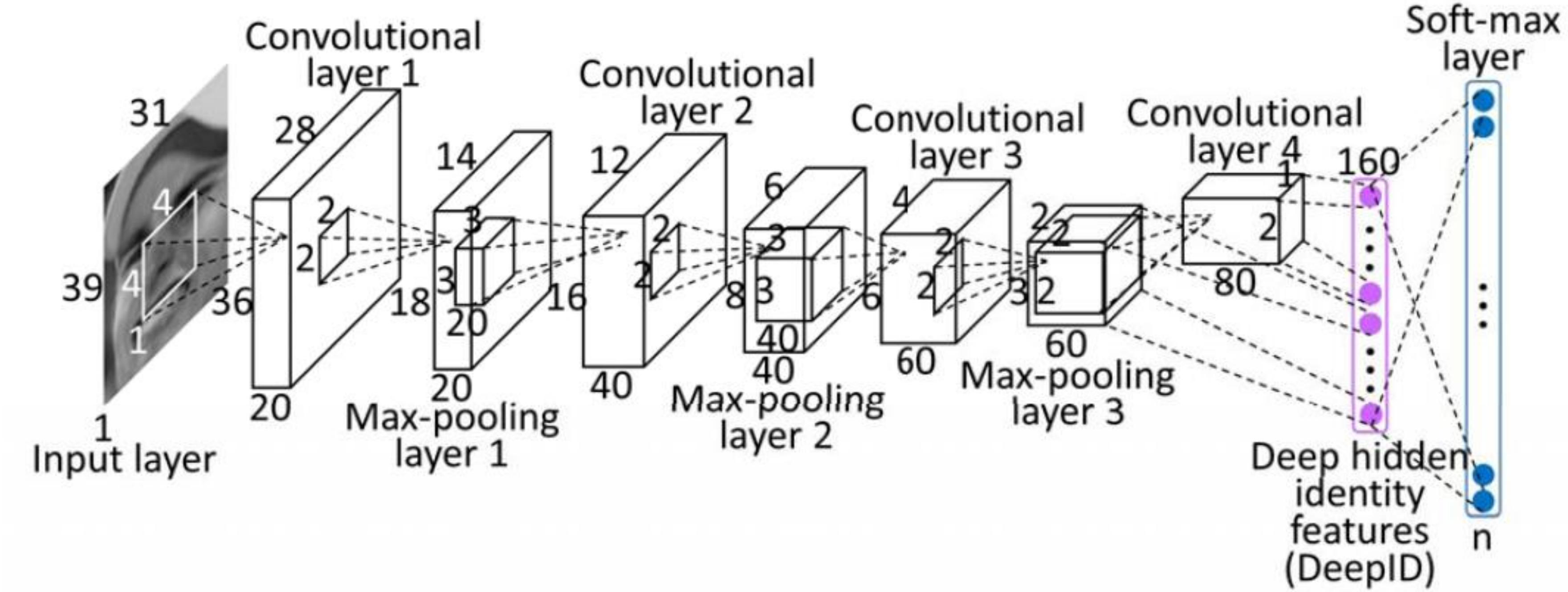
3) Siamese network $d(f_1, f_2) = \sum_i \alpha_i |f_1[i] - f_2[i]|$

Yaniv Taigman, etc. DeepFace: Closing the Gap to Human-Level Performance in Face Verification, CVPR 2014

Classification Network

Application in face verification(Ⅲ)

LFW:97.45%



Face Verification: Joint Bayesian

Yi Sun, etc. Deep Learning Face Representation from Predicting 10,000 Classes, CVPR 2014

Classification & Siamese Network

Application in face verification(V)

LFW:99.15%

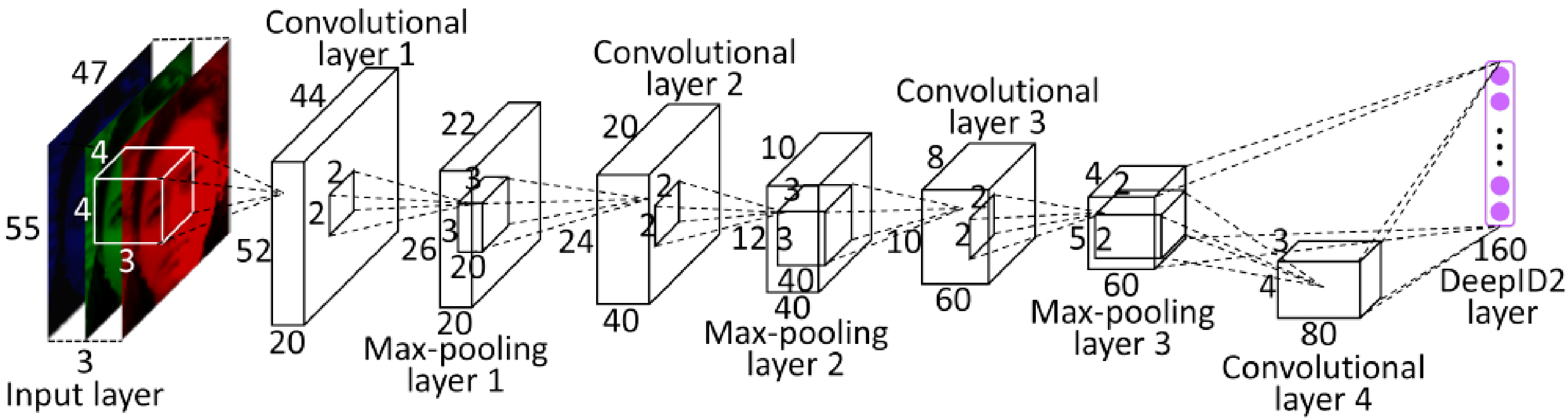


Figure 1: The ConvNet structure for DeepID2 extraction.

Deep Learning Face Representation by Joint Identification-Verification

Yi Sun, etc. Deep Learning Face Representation by Joint Identification-Verification. NIPS 2014

Classification & Siamese Network

Application in face verification(V)

1. identification loss(cross-entropy)

$$\mathrm{Ident}(f, t, \theta_{id}) = -\sum_{i=1}^{n} -p_i \log \hat{p}_i = -\log \hat{p}_t$$

2. verification loss (contrastive)

$$\mathrm{Verif}(f_i, f_j, y_{ij}, \theta_{ve}) = \begin{cases} \frac{1}{2}\|f_i - f_j\|_2^2 & \text{if } y_{ij} = 1 \\ \frac{1}{2}\max\left(0, m - \|f_i - f_j\|_2\right)^2 & \text{if } y_{ij} = -1 \end{cases}$$

3. verification loss (cosine)

$$\mathrm{Verif}(f_i, f_j, y_{ij}, \theta_{ve}) = \frac{1}{2}\left(y_{ij} - \sigma(wd + b)\right)^2$$

where $d = \frac{f_i \cdot f_j}{\|f_i\|_2 \|f_j\|_2}$ is the cosine similarity

Yi Sun, etc. Deep Learning Face Representation by Joint Identification-Verification. NIPS 2014

Classification & Siamese Network

Application in face verification(V)

Table 1: The DeepID2 learning algorithm.

input: training set $\chi = \{(x_i, l_i)\}$, initialized parameters θ_c, θ_{id}, and θ_{ve}, hyperparameter λ, learning rate $\eta(t)$, $t \leftarrow 0$

while not converge **do**

$t \leftarrow t+1$ sample two training samples (x_i, l_i) and (x_j, l_j) from χ

$f_i = \mathrm{Conv}(x_i, \theta_c)$ and $f_j = \mathrm{Conv}(x_j, \theta_c)$

$\nabla\theta_{id} = \frac{\partial \mathrm{Ident}(f_i, l_i, \theta_{id})}{\partial \theta_{id}} + \frac{\partial \mathrm{Ident}(f_j, l_j, \theta_{id})}{\partial \theta_{id}}$

$\nabla\theta_{ve} = \lambda \cdot \frac{\partial \mathrm{Verif}(f_i, f_j, y_{ij}, \theta_{ve})}{\partial \theta_{ve}}$, where $y_{ij} = 1$ if $l_i = l_j$, and $y_{ij} = -1$ otherwise.

$\nabla f_i = \frac{\partial \mathrm{Ident}(f_i, l_i, \theta_{id})}{\partial f_i} + \lambda \cdot \frac{\partial \mathrm{Verif}(f_i, f_j, y_{ij}, \theta_{ve})}{\partial f_i}$

$\nabla f_j = \frac{\partial \mathrm{Ident}(f_j, l_j, \theta_{id})}{\partial f_j} + \lambda \cdot \frac{\partial \mathrm{Verif}(f_i, f_j, y_{ij}, \theta_{ve})}{\partial f_j}$

$\nabla\theta_c = \nabla f_i \cdot \frac{\partial \mathrm{Conv}(x_i, \theta_c)}{\partial \theta_c} + \nabla f_j \cdot \frac{\partial \mathrm{Conv}(x_j, \theta_c)}{\partial \theta_c}$

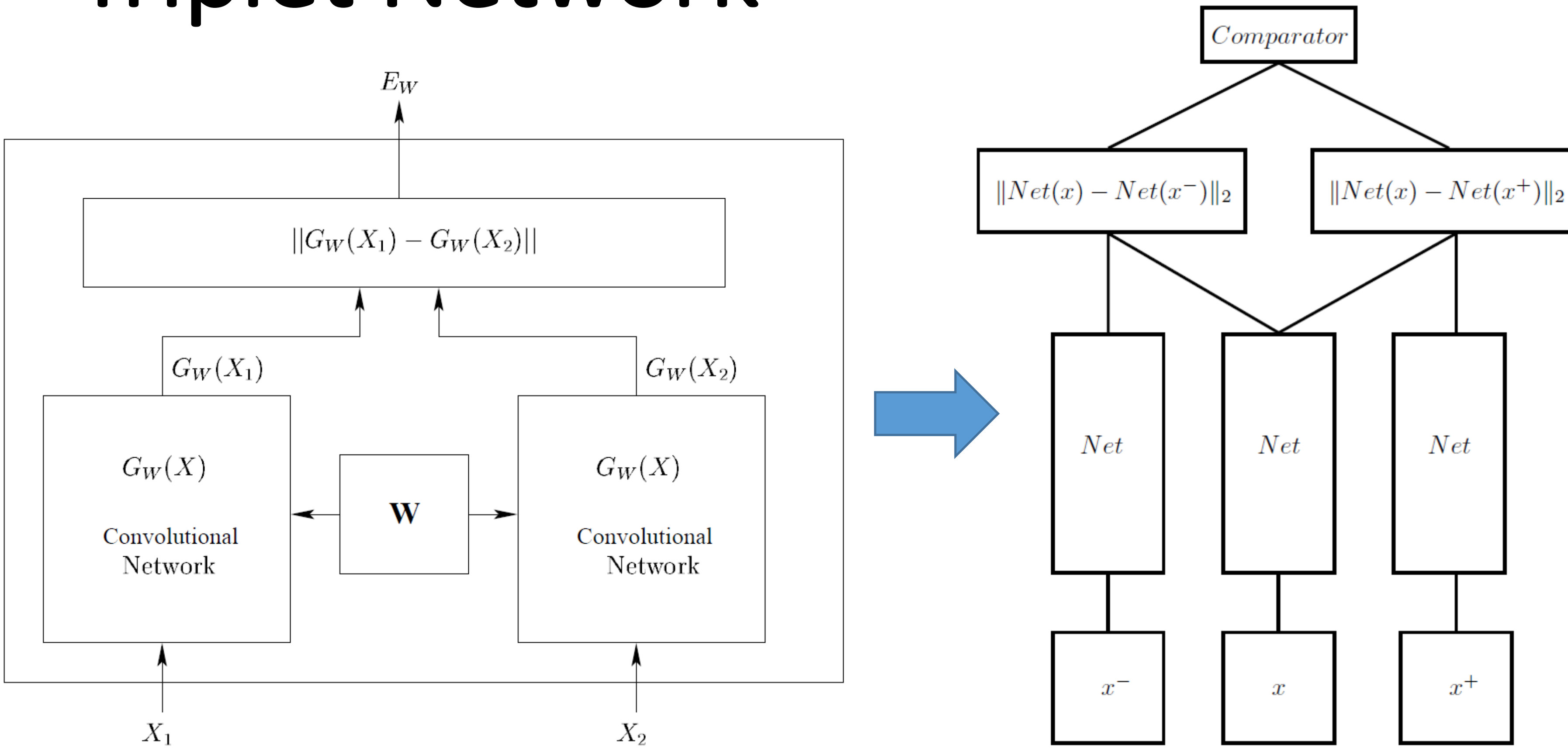
update $\theta_{id} = \theta_{id} - \eta(t) \cdot \theta_{id}$, $\theta_{ve} = \theta_{ve} - \eta(t) \cdot \theta_{ve}$, and $\theta_c = \theta_c - \eta(t) \cdot \theta_c$.

end while

output θ_c

Yi Sun, etc. Deep Learning Face Representation by Joint Identification-Verification. NIPS 2014

Triplet Network



From Siamese to Triplet Network

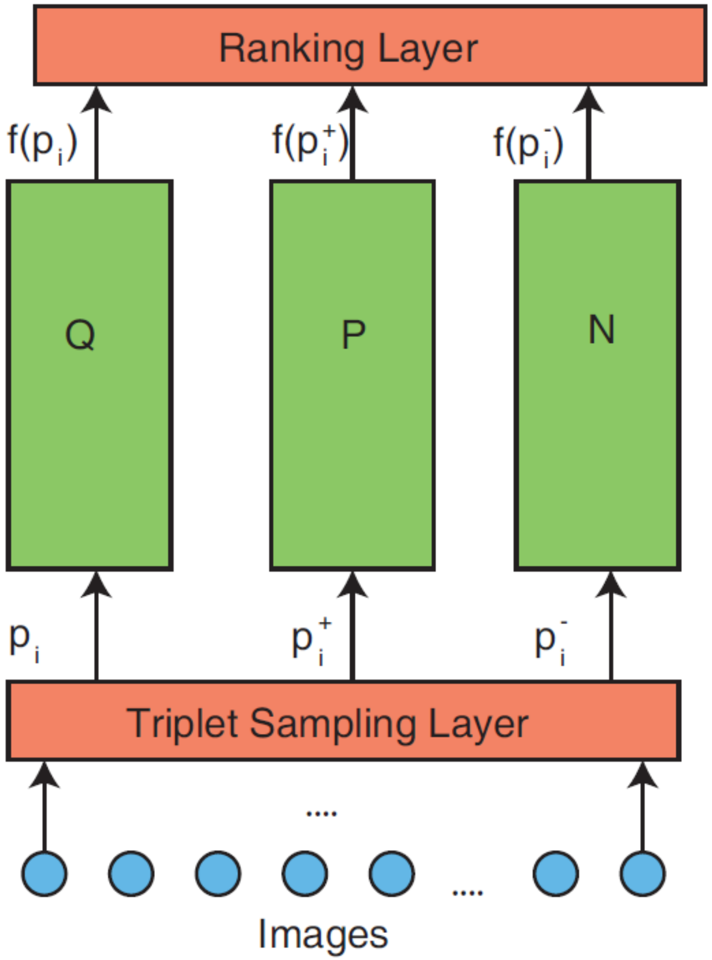
Elad Hoffer, etc. DEEP METRIC LEARNING USING TRIPLET NETWORK. Under review as a conference paper at ICLR 2015
http://arxiv.org/abs/1412.6622

Triplet Network
Application in Image ranking

Sample images from the triplet dataset

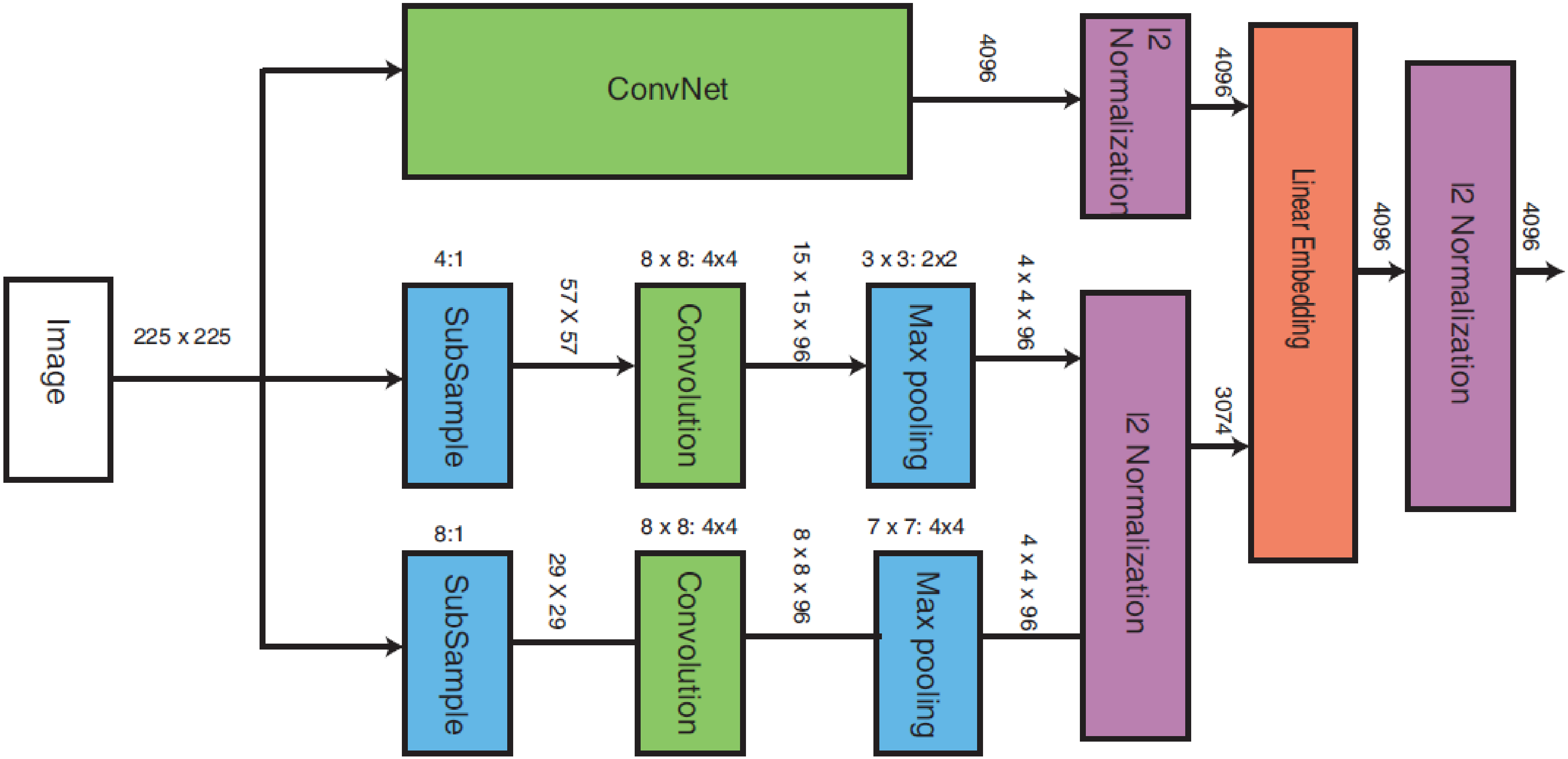
Jiang Wang, etc. Learning Fine-grained Image Similarity with Deep Ranking. CVPR 2014

Triplet Network
Application in Image ranking



Jiang Wang, etc. Learning Fine-grained Image Similarity with Deep Ranking. CVPR 2014

Triplet Network
Application in Image ranking



Jiang Wang, etc. Learning Fine-grained Image Similarity with Deep Ranking. CVPR 2014

Triplet Network

Application in Image ranking

Distance $$D(f(P), f(Q)) = \|f(P) - f(Q)\|_2^2$$

$$D(f(p_i), f(p_i^+)) < D(f(p_i), f(p_i^-)),$$
$$\forall p_i, p_i^+, p_i^- \text{ such that } r(p_i, p_i^+) > r(p_i, p_i^-)$$

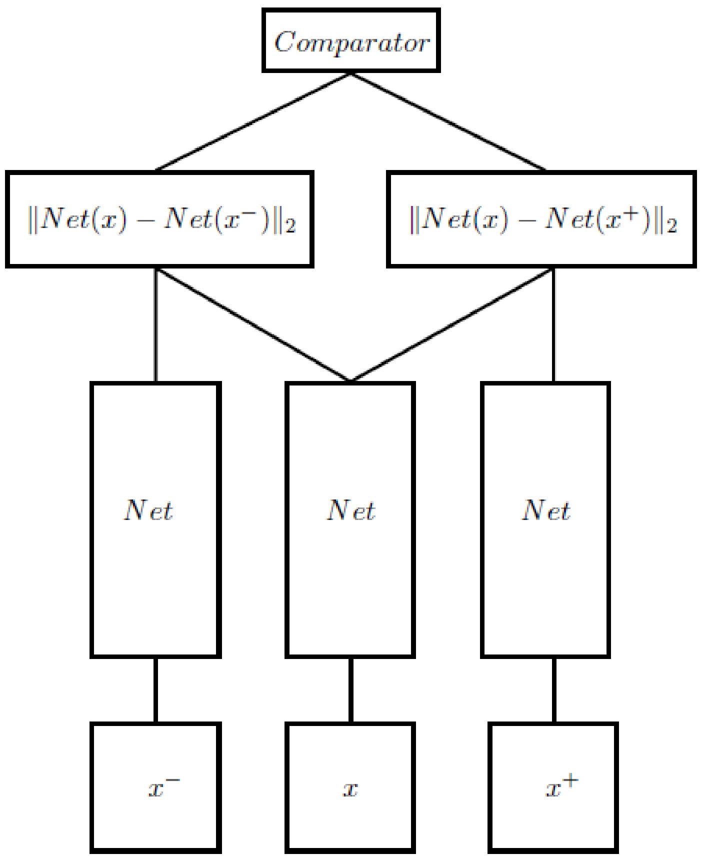
Hinge Loss $$l(p_i, p_i^+, p_i^-) =$$
$$\max\{0, g + D(f(p_i), f(p_i^+)) - D(f(p_i), f(p_i^-))\}$$

Jiang Wang, etc. Learning Fine-grained Image Similarity with Deep Ranking. CVPR 2014

Triplet Network

Application in deep metric learning

$$TripletNet(x, x^-, x^+) = \begin{bmatrix} \|Net(x) - Net(x^-)\|_2 \\ \|Net(x) - Net(x^+)\|_2 \end{bmatrix} \in \mathbb{R}^2_+$$

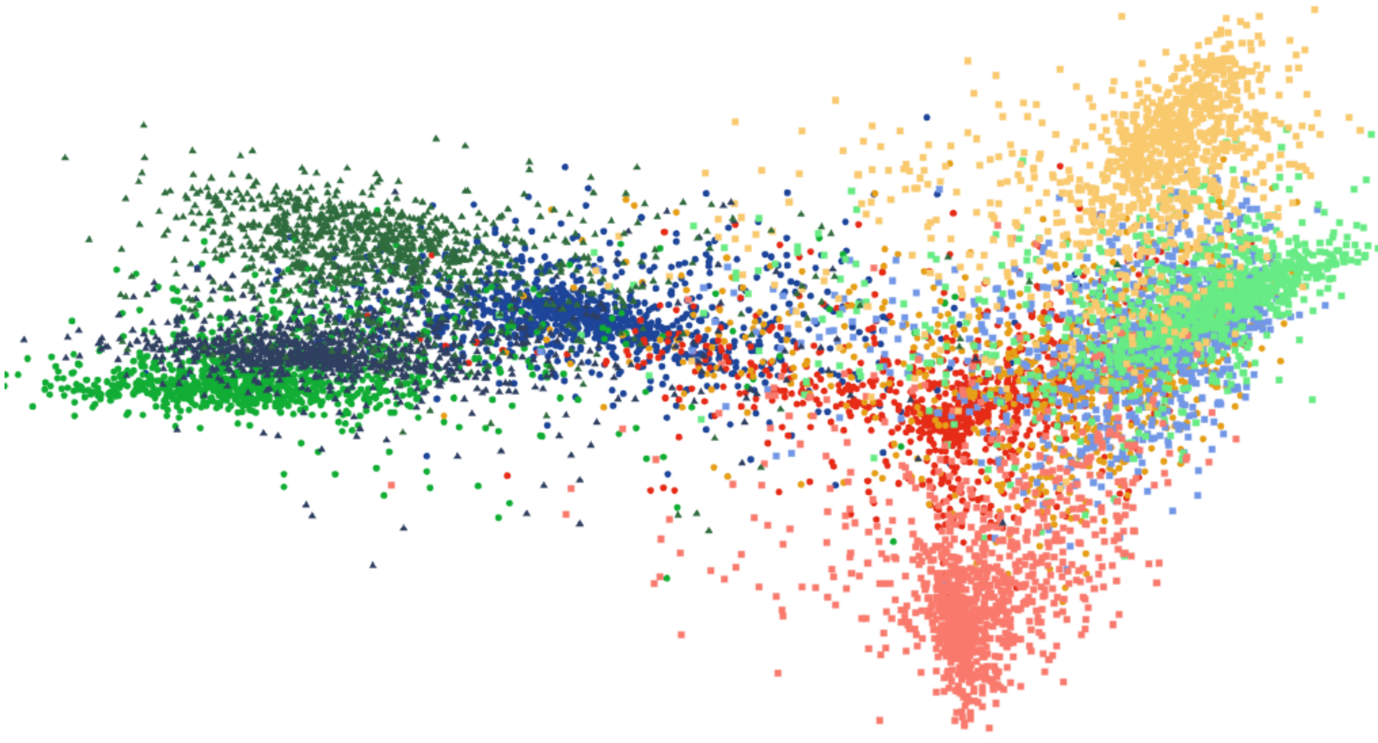


SoftMax function is applied on both outputs

Elad Hoffer, etc. DEEP METRIC LEARNING USING TRIPLET NETWORK. Under review as a conference paper at ICLR 2015
http://arxiv.org/abs/1412.6622

Triplet Network

Application in deep metric learning



2D VISUALIZATION OF FEATURES of CIFAR10

Elad Hoffer, etc. DEEP METRIC LEARNING USING TRIPLET NETWORK. Under review as a conference paper at ICLR 2015
http://arxiv.org/abs/1412.6622

Conclusion

- The loss function in Siamese Network is very important.
- Mixed Network Architecture can improve the performance.
- Caffe implementation of Siamese Network:
  http://caffe.berkeleyvision.org/gathered/examples/siamese.html

Thank you!