International Journal of Machine Learning and Cybernetics
https://doi.org/10.1007/s13042-021-01336-2

ORIGINAL ARTICLE

Self-paced hierarchical metric learning (SPHML)

Mohammed Al-taezi[1] · Pengfei Zhu[1] · Qinghua Hu[1] · Yu Wang[1] · Abdulrahman Al-badwi[2]

Received: 30 March 2020 / Accepted: 15 April 2021

Abstract
Metric learning aims to learn a distance to measure the difference between two samples, and it plays an important role in pattern recognition tasks. Most of the existing metric learning methods rely on pairs of samples. However, the importance of sample pairs varies greatly because of possible noise and the difference between samples and the decision boundaries. In this paper, we propose a robust hierarchical metric learning (SPHML) framework based on self-paced learning, which can help gain knowledge about the weights of sample pairs and utilize them in an easy or hard manner. Hierarchical nonlinear functions are learned by back-propagation to map sample pairs into a more discriminative feature space. Experimentally, our method achieves very competitive performance when compared with state-of-the-art methods.

Keywords Hierarchical metric learning methods · Deep neural networks · Self-paced learning · Face verification

1 Introduction

Metric learning aims to understand the similarity or distance metric to measure the difference between two samples. In recent years, it has attracted much attention in the fields of pattern recognition and computer vision. According to the availability of label information, metric learning can be categorized into supervised, unsupervised, semi-supervised, and weakly supervised methods. Generally, the performance of metric learning models is highly dependent on the extracted features. Distance metrics can be learned separately or jointly with feature learning. Compared with traditional metric learning, which uses handcrafted features or deep features as the input, deep metric learning methods can learn a discriminative and representative feature embedding directly in an-end-to end manner.

Most metric learning models consist of three components: loss function, sample pairs, and optimization algorithms [20, 29]. The most common loss function includes triplet loss, margin loss [39], N-pair loss [36, 40], margin-based loss, ranked list loss [9], etc. The sampling strategies are specially designed to generate sample pairs, such as naive sampling, semi-hard sampling, and hierarchical sampling. The optimization variable of metric learning can be a projection matrix or a positive definite matrix in Mahalanobis distance.

Although massive metric learning algorithms are designed, the challenges in the field are still far from resolved. Firstly, when sample pairs are generated from the training data, the importance of different sample pairs are usually ignored and not well exploited in the loss function of metric learning models. Secondly, most existing metric learning models adopt stochastic gradient decent as the optimization tool with a mini-batch strategy [14]. Recent researches in object detection and image classification show that learning from easy to hard progressively can bring about significant improvement in the model performance. Thirdly, the noise in the data inevitably affects the stability of metric learning models. The robust models should be developed to improve the robustness and generalization ability of metric learning algorithms. In classification, the weight of noise is thought to be effective and enhances the generalization, but such methods only work [20] in the classification problem. The noise removal techniques and data purification methods are important points in machine learning. STFRD+PMML [6] uses Whitened PCA (WPCA) to reduce the dimension of TF features and suppress the noise in the leading eigenvectors. In our SPHML model, we used feature descriptors SIFT and LBP on LFW and TYF dataset and then applied WPCA to project features into a 500-dimensional feature vector and

✉ Mohammed Al-taezi
mohd_altazzi@hotmail.com

[1] Intelligence and Computing School, Tianjin University, Tianjin, China

[2] Computer Science Collage, Central South University, Changsha, China

Published online: 10 June 2021

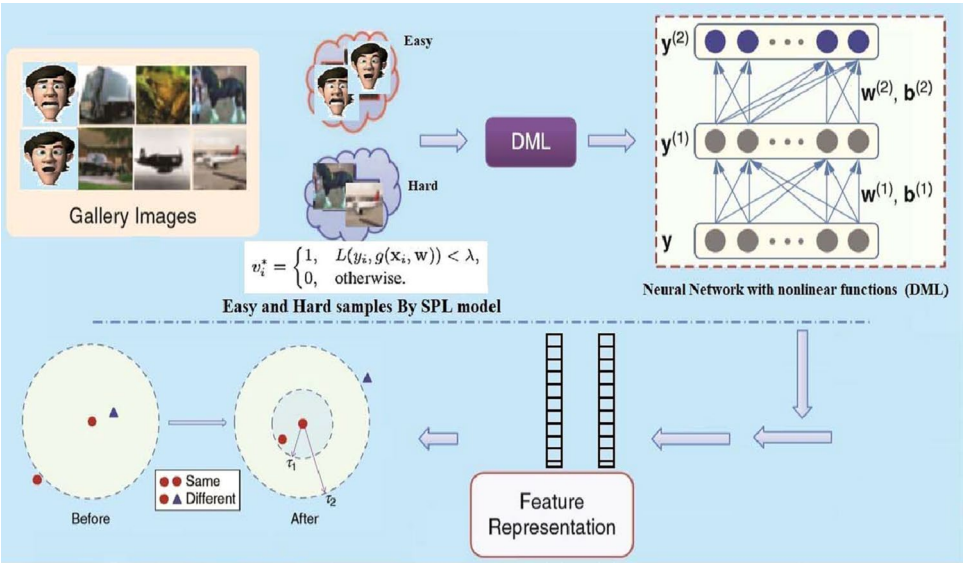

Fig. 1 The flowchart of the SPHML model learns the weights by optimizing a hierarchical metric learning loss function with an SPL regularizer and exploits these sample pairs from easy to hard. As the loss induced by noisy sample pairs is very large, the latent weight will become zero and therefore the impact of noise can be alleviated. Then, a deep neural network is adopted to learn a set of hierarchical nonlinear transformations, using back-propagation to map sample pairs into other subspace, where each positive sample pair is less than a smaller threshold τ_1 and that of each negative pair is higher than a larger threshold τ_2, so that discriminative information is exploited for the robustness accuracy in unconstrained environmental images

remove the redundancy, which can be alleviated with noise data. Besides, to avoid overfitting of the learned multi-layer neural networks, we add salt and pepper noise to the images. As the self-paced regularizer can penalize the samples with larger loss, which could be caused by noise in the data. The proposed method should be robust to the noise from the perspective of the loss functions. In this paper, we focus on the design of metric learning models and, thus, only compare it with the state-of-the-art metric learning algorithms.

We addressed two issues in Page 2, i.e., the importance of sample pairs and noise in the data. In this paper, we propose a novel self-paced hierarchical metric learning (SPHML) model, as shown in Fig. 1. A self-paced learning SPL strategy is inspired by the curriculum learning. The aspect of SPL is embedded into the loss function of metric learning by introducing latent weights of sample pairs, where the importance of sample pairs varies greatly because of possible noise and the difference between samples and decision boundaries. SPHML learns the weights dynamically by optimizing a metric learning loss function with a self-paced regularizer and exploits these sample pairs from easy to hard. As the loss induced by noisy sample pairs is very large, the latent weight will become zero and therefore the impact of noise can be alleviated.

SPHML utilizes a deep neural network to learn a set of hierarchical nonlinear transformations, using back-propagation to map image pairs into other subspace, where each positive image pair is less than a smaller threshold. Moreover, it helps each negative pair higher than a larger threshold. Finally, it helps solve the scalability problem in unconstrained environmental images so that discriminative information is exploited for robustness in accuracy. Experimental results on LFW and YTF datasets show the effectiveness of the proposed model.

The rest of this paper is organized as follows: Sect. 2 reviews self-paced learning and deep metric learning. Section 3 shows the algorithms of DML and our model SPHML. Thereafter, we discuss the significant implementation points of SPHML and the results of the experiments in Sect. 4. The conclusion and future work are presented in Sect. 5.

2 Related work

2.1 Metric learning

Machine learning (ML) consists of supervised and unsupervised learning in which the metric learning method is significant, and this can evaluate machine learning algorithms. The main objective of metric learning methods is to learn a distance metric to measure the difference between two samples so that the sample's hard pairs are as far as possible, and the distance between the sample's easy pairs is reduced from threshold. Most of the metric learning algorithms have been explored in past articles, and their main attention is to solve the problem of verification face in the wild [3, 6, 8, 37, 50].

Most of them are not strong enough to obtain the nonlinear transformation wherever face images usually lie, as these methods use linear functions to project face representations to a new feature space. To solve this problem, some methods use the kernel trick, which is commonly used to first map face representations to feature space with high dimension and then learn the distance metric in the high dimensional feature space [12, 21, 47]. But these methods cannot capture the nonlinear functions to solve the problem of scalability.

2.2 Self-paced learning (SPL)

Since 2009, the curriculum learning CL [2] provided a new learning model, where the model starts learning gradually from easy to hard samples. The main idea is to find a ranking function that gives priority to samples model in training time. The curriculum is extracted by heuristics predetermined for the problem, where the curriculum (ranking function) is divided into sentences. The heuristic is that the number of solutions increase exponentially depending on the length of the sentence, and the short sentence is easy and must be learned first. Sometimes, in the heuristical curriculum, there is an inconsistency between the dynamic and fixed curriculum models: in the latter, the curriculum learning (CL) is fixed and cannot be modified depending on the feedback about the learner. To remove the drawback of CL, a new model called self-paced learning (SPL) [25] is suggested, which combines the curriculum (regularization) design with the learning objective function. This method has two advantages: first, it combines and optimizes the learning objective model with the curriculum, and, consequently, both the curriculum and the learned objective model are consistent under the same optimization problem and second, the regularization term in model is independent of loss functions of specific problems.

The main goal of self paced learning is to reduce the influence of the noise and highly corrupted data. The learning robustness relies on a sample selection to distinguish the reliable samples from the confusing ones. One key issue that both CL and SPL have is avoiding bad local minima and providing high generalization result [22, 41, 44]. The second key issue of SPL is to get better weighting strategy by the minimizer loss functions.

For self-paced learning, the weights of sample pairs are automatically learned and updated in the training process. Thus, the system tends to learn with easy sample pairs first and then hard pairs, which is similar to AdaBoost. Hence, there should be no special bias to certain datasets. There are multiple variations of this SPL learning regime, such as self-paced reranking [18], self-paced multiple instance learning [49], and self-paced learning with diversity [17]. The effectiveness of this SPL paradigm is visible in various machine learning and computer vision tasks.

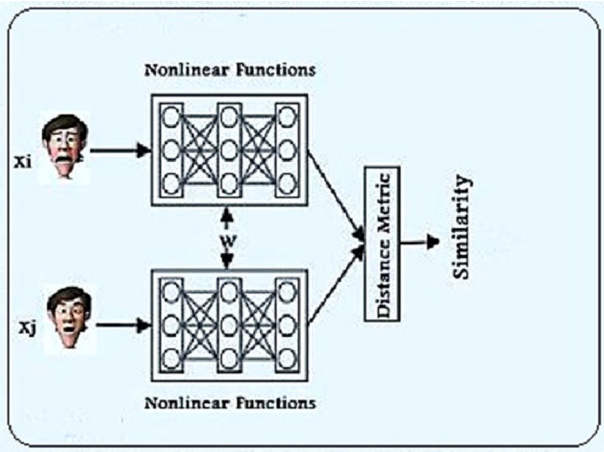

Fig. 2 Flowchart of the DML model

2.3 Deep metric learning (DML) model

In recent years, deep neural networks have attracted increasing attention, especially relating to artificial intelligence, machine learning, and computer vision, and a lot of deep learning algorithms have been stated in many articles [10, 28, 35, 38]. Generally, most of the methods are based on feature engineering, such as autoencoder [28, 48], Convolution neural networks and CNN deep belief network [3, 11], face verification, and human action recognition [30, 42]. The main purpose of deep learning is to learn a set of hierarchical nonlinear functions to map sample pairs into one feature space where the nonlinear transformations mapping is captured.

SPHML proposed model uses the deep neural networks to learn the nonlinear functions, which can use a back propagation to train the network, as shown in Fig. 2. Our method uses both SPL and DML methods to strengthen the accuracy in unconstrained environmental images algorithm, which outperforms the other methods of metric learning.

3 Proposed approach

In this section, we first review the self-paced learning SPL method and DML model and then present new model self-paced hierarchical metric learning (SPHML).

3.1 Self-paced learning (SPL) method

Let $L(y_i, g(x_i, w))$ indicate the loss function that measures the cost between the ground truth label y_i and the calculated label $g(x_i, w)$; w indicates to the model parameter inside the decision function g. In SPL, the aim is to combine and

learn the model parameter$w$ and the variable of latent weight $v = [v_1, \ldots, v_n]^T$ minimization.

$$\min_{w,\mathbf{v}\in[1,0]^n} \mathbb{E}(w, v, \lambda) = \sum_{i=1}^{n} v_i L(y_i, g(x_i, w)) + f(\mathbf{v}, \lambda) \quad s.t. \mathbf{v} \in \Psi \tag{1}$$

$f(v;\lambda)$ self-paced function determines a learning scheme. Suppose that $l = [l_1, \ldots, l_n]^T$ indicates loss and$v = [v_1, \ldots, v_n]^T$ represents latent weight variables reflecting the samples' importance; λ controls the learning pace in training, and Ψ is a feasible region that encodes the prior knowledge. A curriculum is described as follows :

Definition 1 (*Total ranking curriculum*) Given a training set $D = \{(x_i, y_i)\}_{i=1}^n$, where x_i refers to ith observed data sample and y_i its corresponding label,γ is a ranking function, where$\gamma(x_i) < \gamma(x_j)$ represents thatx_i must be learned earlier than x_j in training time.$\gamma(x_i) = \gamma(x_j)$ shows the order for the two samples without preferred learning.

Definition 2 (*Curriculum CL region*) Let $D = \{(x_i, y_i)\}_{i=1}^n$ indicate the training set, $\gamma(.)$ is predetermined curriculum on training samples, $v = [v_1, \ldots, v_n]^T$ is weight variables, and Ψ is a curriculum region of γ as follow:

2.1Ψ is a non empty convex set.

2.2 For any sample pair (x_i, x_j), if $\gamma(x_i) > \gamma(x_j)$, it holds that $\int_\Psi v_i dv > \int_\Psi v_j dv$, where $\int_\Psi v_i dv$ calculates expectation of v_i respect to Ψ, also if $\gamma(x_i) = \gamma(xj)$, then $\int_\Psi v_i dv = \int_\Psi v_j dv$.

The Definition 2.1 confirms the calculating of the constraints, and Definition 2.2 indicates that large expected value must be for the samples that are learned early.

Definition 3 (*SPL function*) For a learning scheme of SPL, suppose that $l = [l_1, \ldots, l_n]^T$ represents to loss, $v = [v_1, \ldots, v_n]^T$ is a vector of latent weight variable for each sample, λ controls the learning pace in training, and Ψ is a feasible region that encodes that prior knowledge. $f(v;\lambda)$ is called a self-paced function, if

3.1 $f(v;\lambda)$ is convex with respect to $v \in [0, 1]^n$ that ensures the model can learn good solutions within the curriculum region. In addition, the $\| v \| = \sum_{i=1}^n v_i$ term corresponds to a binary scheme, where one can only get binary values [0, 1].

3.2 $v^*(\lambda;l)$ monotonically decreases with respect to loss l_i, by utilizing the partial gradient to zero and fixed all variables except for v_i ,l_i to get optimal solution of v, it shows v_i decreasing according l_i that $\lim_{l_i \to 0} v_i^* = 1$ indicates the model gradually chooses easy samples with small noise, where $\lim_{l_i \to \infty} v_i^* = 0$, the model chooses hard samples with large noise.

3.3 $\| v \|_1 = \sum_{i=1}^n v_i$ increases with respect to λ, and it holds that $\forall i \in [1, n], \lim_{\lambda \to 0} v_i^* = 0, \lim_{\lambda \to \infty} v_i^* = 1$ when the model is young, it selects few samples, and when the model "age" becomes more, it should add more hard samples to train as "mature" model.

where $v^* = \arg\min_{v\in[0,1]^n} \sum v_i l_i + f(v;\lambda)$, and also refer to $v^* = [v_1^*, \ldots, v_n^*]$.

The SPL method gives some freedom to adapt the curriculum to learning paces that can use diffident f self-paced functions in regularization term depending on the type of the problem. There are many learning schemes we can use one of them such as *binary scheme* or hard scheme, see Eq. (9), where the weight variable can take binary value, *linear scheme* the weight samples are real value, *mixture scheme*, which combines both hard and soft schemes, can afford small errors until a fixed point, *logarithmic scheme* penalizes the loss weight by logarithm function [5, 25, 27, 31].

3.1.1 Selecting input samples from easy to hard

To select all samples in the dataset from (easy) samples with low noise to (hard) samples with much noise, we can use the following calculation:

$$v_i^* = \begin{cases} 1, & \mathrm{L}\,(y_i, g(x_i, w)) < \lambda' \\ 0, & \mathrm{o}\,\mathit{therwise}. \end{cases}$$

An alternative search method can be used to achieve this: first, update v with a fixed w, a sample has weight loss less than the threshold value λ' is selected as an easy sample and set $(v_i^* = 1)$. Otherwise, $(v_i^* = 0)$ the model can not select any hard sample for training. Second, update w with a fixed v, when a parameter λ' is small, only easy samples with small losses will be selected in training time, then when λ' increasing, more hard samples with large losses will be added to the training model as "mature" model gradually. So, the classifier is trained only on the "easy" samples. The parameter λ' can control the pace of the model "age" to earn new samples [1, 25].

The SPHML is inspired from SPL to learn the weights by optimizing a metric learning loss function with a self-paced regularizer and exploits these sample pairs from easy to hard. The improvement of the performance mainly comes from two aspects, i.e., utilizing sample pairs from easy to hard and robustness to noisy samples, which are both modeled by self-paced learning.

3.2 Deep metric learning (DML) model

Most of the distance metric learning methods [20, 23] use linear transformations employed in projects inputting samples into high dimensional feature space to learn the distance between two samples [26]. So, the kernel trick solves

limitation and employs the nonlinear transformations for face images in unconstrained environments that have varying noise levels such as illuminations, lighting, background, and different poses when face images are captured in the wild [13]. However, the kernel trick suffers from scalability problem in input samples. By contrast, hierarchical deep metric learning can learn nonlinear transformations hierarchically so that it can solve the linear transformations and scalability problems at the same time.

Algorithm 1 Deep Metric Learning DML

Input the sample $X = (x_i, x_j)$, pairwise label l_{ij} threshold τ, number of network layers $N+1$, iterative number I_t, learning rate μ, regularization parameter λ and convergence error ξ.
Output: The parameters of the model: $(w^{(n)}, a^{(n)})_{(n=1)}^{N}$

Initialize $(w^{(n)}, a^{(n)})_{(n=1)}^{N}$
for $t = 1 : l_t$ **do**
  Make back propagation
    Select sample pairs randomly $X = (x_i, x_j)$ from X
      Find $h_i^{(0)} = x_i$ and $h_j^{(0)} = x_j$ by Equations(2)(3)and (4).
  for $n = 1 : N$ **do**
    Make forward propagation to get $h_i^{(n)}$and $h_j^{(n)}$
  end for
  for $n = N : 1$ **do**
    Computing gradient by back propagation by Equations (12) and (13)
  end for
  for $n = 1 : N$ **do**
    Updating $w^{(n)}$ and $a^{(n)}$ by Equations (14) and (15)
  end for
  Calculate T_t using (8).
  If $t > 1$ and $|T_1 - T_{(t-1)}| < \xi$, go to Return.
end for
Return: $(w^{(n)}, a^{(n)})_{(n=1)}^{N}$

Algorithm 1 shows a set of DML to calculate the representations of image pairs by multiple layers of nonlinear functions. Assume there are N + 1 layers in neural network and $p^{(q)}$ units in the *n th* layer, where n = 1, 2, 3,..., N as shown in Fig. 3. For a given face sample $v \in R^d$, the first layer output is calculated by:

$$h^{(1)} = s(w^{(1)}\mathbf{x} + a^{(1)}) \in R^{p(1)} \tag{2}$$

where $w^{(1)} \in R^{p(1)*d}$ is a map matrix to be learned in the first layer, $a^{(1)} \in R^{p(1)}$ is a bias vector, and $s : R \longmapsto R$ is a nonlinear activation function as *sigmoid* or *tanh*. First layer output $h^{(1)}$ used as input vector for the second layer as vector-valued function:

$$h^{(2)} = s(w^{(2)}h^{(1)} + a^{(2)}) \in R^{p(2)} \tag{3}$$

where $w^{(2)} \in R^{p(2)*p(1)}$ is a map matrix, $a^{(2)} \in R^{p(2)}$ is a bias vector for second layer, and s is a nonlinear activation function for second layer, repeatedly. The output for the *nth* layer is:

$$h^{(n)} = s(w^{(n)}h^{(n-1)} + a^{(n)}) \in R^{p(n)} \tag{4}$$

The output top level can be calculated as:

$$f(\mathbf{x}) = h^{(N)} = s(w^{(N)}h^{(N-1)} + a^{(N)}) \in R^{p(N)} \tag{5}$$

where $f(\mathbf{x})$ is activation nonlinear function which can get $w^{(n)}$ and $a^{(n)}$ in n =1 ,2 ,..., N. The distance between pair of face samples x_i and x_j by Euclidean distances of most top level layers, is as follows :

$$d_f^2(x_i, x_j) = \| f(x_i) - f(x_j) \|_2^2 \tag{6}$$

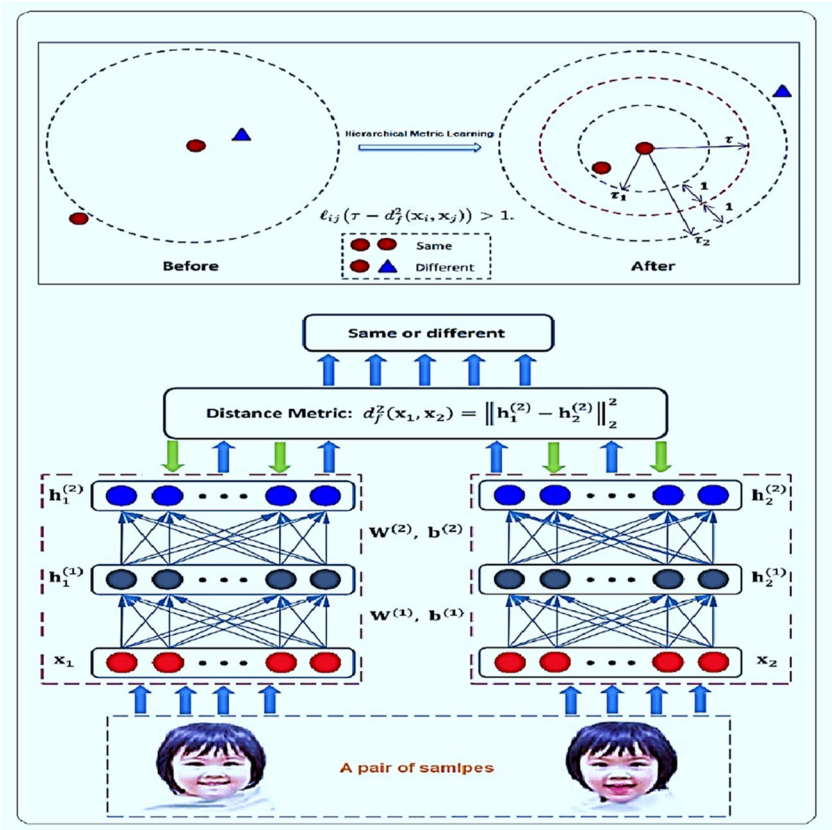

Fig. 3 Architecture hierarchical networks of DML model

The model can exploit discriminative information at the highest level for face representation. There is a margin between each hard and easy sample pairs in the feature space as shown in Fig. 3; when SPHML method is applied, the distance of an easy sample pair is less than a smaller threshold τ_1, and a hard sample pair is higher than a large threshold τ_2 of the most top level of SPHML model. To reduce the parameter in the experiment, we can use one threshold τ ($\tau > 1$) rather than τ_1, τ_2. So, the margin between $d_f^2(x_i, x_j)$ and τ is larger than 1 as follows:

$$l_{ij}(\tau - d_f^2(x_i, x_j) > 1 \tag{7}$$

where $\tau_1 = \tau - 1$ and $\tau_2 = \tau + 1$. It is more useful to exploit discriminative information for face pairs that can add more robustness to accuracy. SPHML is inspired from DML, a deep neural network which learns a set of hierarchical nonlinear transformations using back-propagation to map image pairs into other subspace, where each positive pair is less than a smaller threshold τ_1 and that of each negative pair is higher than a larger threshold τ_2 as in Fig. 3. Moreover, it solves scalability problem. This paper combines two models, the SPL with hierarchical deep metric learning, to optimize new robustness model to attain more accuracy for unconstrained environmental images.

3.3 Architecture self-paced hierarchical metric learning SPHML model

We have optimized a new formula for SPHML model as shown in Algorithm 2; our main objective function Eq. (8) has T_1, T_2 and $f(v, \lambda)$, where T_1 represents logistic loss function with latent variable v_{ij} and T_2 defines regularization term then $f(v, \lambda)$ is SPL function, respectively. The logistic loss function $g(z) = \frac{1}{\beta}\log(1 + \exp(\beta z))$, then $\|D\|_F$ defines the Frobenius norm for matrix D, and β is a parameter for sharpness as follows :

$$\begin{aligned} \arg\min_f \; T &= T_1 + T_2 + f(v, \lambda) \\ &= 1/2 \sum_{i,j} v_{ij}\, g(1 - l_{ij}(\tau - d_f^2(\mathbf{x}_i, \mathbf{x}_j))) \\ &\quad + \lambda/2 \sum_{n=1}^{N} (\| w^{(n)} \|_F^2 + \| a^{(n)} \|_2^2) \\ &\quad - \lambda \sum_{i,j=1}^{n} v_{ij} \end{aligned} \tag{8}$$

The objective function penalizes the image pairs, which do not match the desired constraints that adopt non linear function tan$h$ to find distance $d_f^2(x_i, x_j)$ between sample pair, where the distance is smaller than τ that means the sample

pair (x_i, x_j) belong to same class ($l_{ij} = 1$); otherwise the sample pair (x_i, x_j) belong to different class ($l_{ij} = -1$),where pairwise label l_{ij} refers to dissimilarity and similarity between sample pairs input(x_i, x_j) as in Fig. 1. While $\sum_{i,j} v_{ij}\, g(1 - l_{ij}(\tau - d_f^2(\mathbf{x}_i, \mathbf{x}_j)))$ is the generalized logistic loss function in our model, the loss induced by noisy sample pairs is very large, and the latent weight v_{ij} become zero and therefore the impact of noise can be alleviated otherwise $v_{ij} = 1$. This indicates the model gradually selects easy samples (with small loss) and hard samples (with large loss) in each iteration. The loss function is a smooth approximation of the hinge loss $[z]_+$,β as the parameter sharpness.

Where $T2$ defines regularization term, $\|w\|_F$ represents the Frobenius norm, and λ represents regularization parameter, respectively. $f(v, \lambda)$ defines SPL function method which measures the learning scheme, where $v = [v_1, \ldots, v_n]$ shows the latent weight variables belonging to the samples' importance. We employ a binary scheme for SPL function, which only yields binary weight variables, and then stochastic sub gradient descent scheme to get values for (w^n, a^n).

$$f(v, \lambda) = -\lambda \|v\|_1 = -\lambda \sum_{i,j=1}^{n} v_{ij} \tag{9}$$

Algorithm 2 Self-paced Hierarchical Metric Learning SPHML.

Input: Input dataset LFW or YTF, predetermined curriculum , a set: $X = (x_i, x_j, l_{ij})$ threshold 1 τ, number of network layers $N + 1$, iterative number I_t, learning rate μ, regularization parameter λ, and convergence error ξ.

Output: The parameters of the model: $(w^{(n)}, a^{(n)})_{(n=1)}^{N}$

Initialize $(w^{(n)}, a^{(n)})_{(n=1)}^{N}$
for $t = 1 : l_t$ **do**
  Randomly a set $h_i^{(0)} = x_i$ and $h_j^{(0)} = x_j$
  Selecting easy samples, logistic loss function g(z)
  If $g(z) > \upsilon$ then $\upsilon = 0$;
  $g(z) = g(z) * \upsilon$
  else
  $\upsilon = 1$;
  $g(z) = g(z) * \upsilon$;
  $end \;\; if$
  for $n = 1 : N$ **do**
    Make forward propagation to get $h_i^{(n)}$and $h_j^{(n)}$
  end for
  for $n = N : 1$ **do**
    Getting gradient by back propagation by Equations (12) and (13)
  end for
  for $n = 1 : N$ **do**
    Updating $w^{(n)}$ and $a^{(n)}$ by Equations (14) and (15)
  end for
  Calculate T_t using (8).
  If $t > 1$ and $|T_1 - T_{(t-1)}| < \xi$, go to Return.
end for
Return: $(w^{(n)}, a^{(n)})_{(n=1)}^{N}$

where f is defined as binary scheme for SPL function that controls the learning model and indicates that image samples to be learned earlier must have larger predict values, we can set $\gamma : X \to 1, 2, \ldots, n$ as a ranking function, where $\gamma(x_i) < \gamma(x_j)$ represents that(x_i) must be learned earlier than(x_j) in training time. $\gamma(x_i) = \gamma(x_j)$ refers to the order of the two samples without preferred learning. The symbolλ indicates the learning rate of model "age" when the model is young, it selects few samples (easy samples) which monotonically increases as the model becomes more mature it selects more samples (hard samples) in each iteration in training.

There are many activation functions used inside nodes of neural networks as *sigmoid* or *tanh*. We use the *tanh* as activation function because it has good result in our model. The formula of the tanh function and its derivative is given below:

$$t(z) = \tan h(z) = \frac{e^z - e^{-z}}{e^z + e^{-z}} \tag{10}$$

$$t'(z) = \tan h'(z) = 1 - \tan h^2(z) \tag{11}$$

The sub-gradient descent of the objective function T to get the parameters $w^{(n)}$ and $a^{(n)}$ are as follows:

$$\frac{\partial T}{\partial w^{(n)}} = \sum_{i,j} v_{i,j} \Big(\triangle_{i,j}^{(n)} \mathbf{h}_i^{(n-1)^T} + \triangle_{j,i}^{(n)} \mathbf{h}_j^{(n-1)^T} \Big) + \lambda \mathbf{w}^{(n)} \tag{12}$$

$$\frac{\partial T}{\partial \mathbf{a}^{(n)}} = \sum_{i,j} v_{i,j} \Big(\triangle_{i,j}^{(n)} + \triangle_{j,i}^{(n)} \Big) + \lambda \mathbf{a}^{(n)} \tag{13}$$

Algorithm 2 shows SPHML solving our objective function Eq. (8), a set inputs of a pre-determined curriculum in training. In step 1, we start to initialize values w and a ,$(1 \le n \le N)$, and in step 2, we optimally use back prorogation to find the gradient descent in neural network. In step 3, sample input pairs are randomly selected, and step 4 learns the easy samples and then increases to the most complex samples in training, and then utilizes forward propagation to compute gradient descent for objective function T for both w and a as shown in Eqs. (12) and (13), where layers n = 1, 2, 3,...,$N-1$ and $h_i^{(0)} = x_i$ and $h_j^{(0)} = x_j$. By step 17 in Eqs. (14) and (15), we update gradient descent to get both w and a until convergence ξ.

$$\mathbf{w}^{(n)} = \mathbf{w}^{(n)} - \mu \frac{\partial T}{\partial w^{(n)}} \tag{14}$$

$$\mathbf{a}^{(n)} = \mathbf{a}^{(n)} - \mu \frac{\partial T}{\partial a^{(n)}} \tag{15}$$

It is useful for SPHML to learn that the significance of sample pairs differ widely because of possible noise and the difference between samples and decision boundaries. Also, the local bad minima can be avoided because SPL method learns latent variable models by an iterative process that can select easy input samples (less noise) and updates the training parameters at each iteration. At each iteration, the number of samples is selected by a weight that is gradually increased such that later iterations has more hard samples (more noise), the SPL applies not to specific input samples in training but to all input samples. And nonlinear transformation functions can be utilized by hierarchical metric learning to project image pairs progressively that map image pairs into one feature space, which demonstrates that the negative image pair is more than a larger threshold τ_2 and the distance of positive image pair is less than a smaller threshold τ_1. The weights of sample pairs and utilizing them to improve a distance metric or similarity measure between two different samples. The exploited discriminative information in the SPL and deep neural network can add more robustness to accuracy in our model.

4 Experiments

4.1 Experiment settings on dataset

We develop a framework SPHML to learn similarity or distance metrics for unconstrained environmental images. The objective similarity metric learning to learn a proper distance or similarity measure to compare pairs of samples that need to label each sample to measure the distance between two samples images. The power of discriminative for similarity metrics provides the solution for the robustness accuracy in the computer vision such as the recognition field. The other key issue that SPHML utilizes to solve the noise caused by restricted images that are taken under unconstrained conditions and show large significant noise variations in pose, background, expression, and lighting. In addition, one challenge is to preserve the similarity metric robustness between the noise in samples and the large variations in unconstrained environmental.

Firstly, LFW and YTF are two benchmark datasets for face verification, which is very suitable for verifying the performance of metric learning models. Secondly, the state-of-the-art metric learning algorithms, e.g., DDML [13] and AHISD [4] use the two face datasets to conduct experiments. We follow the experiment setting of the existing metric learning models, which could be easy and fair for comparison. Of course, the proposed model is not limited to face

recognition and can easily extended to other types of tasks. The following framework explains the details of effects and results of our proposed model.

4.1.1 The LFW dataset

The LFW[1] dataset [15] contains more than 13000 images for the faces that belong to 5749 subjects obtained from the Internet with different expressions, as per pose, resolution, illumination, light, and age. There are two types of supervised learning training for the LFW dataset: (1) the unrestricted face images and (2) the restricted face images. In our model, we use the restricted face image type, which has pairwise label information that can help the model to learn. The standard protocol evaluation dataset "View 2" contains 3000 similar pairs and 3000 dissimilar pairs [15]. The LFW dataset is divided into ten blocks, and each block contains 300 similar (easy) face pairs and 300 dissimilar (hard) face pairs. We used the "funneled" version of the LFW dataset and then utilized some descriptor tools as SIFT (SIFT) provided by Refs. [8, 19]. This was then concatenated into 3456 dimensional feature vector. We utilized PCA to remove the redundancy and applied the square root of each feature to enhance the work of SPHML model when the various feature descriptors are joined as status in Refs. [24, 33, 34, 46].

4.1.2 The YTF dataset

The YTF[2] dataset [45] contains 3425 videos that belong to 1595 different people, obtained from the YouTube website and the mean is 181.3 frames for each video clip. There are a lot of changes in expression, illumination, pose, and light in each video clip. In our paper, we used standard protocol [45] in our model for unconstrained face verification, which contains 5000 video clip pairs. These are divided into ten blocks, and each block has 250 similar pairs and 250 dissimilar pairs. For the YTF dataset, we utilized the provided feature descriptor LBP [45] for all face images aligned by the discovered facial significance. Finally, we utilized PCA to project each mean vector into a 500-dimensional feature and remove the redundancy.

For SPHML, we ran a deep neural network that consists of three layers ($N = 2$) by a set of the following parameters the threshold 1 $\tau = 3$, learning rate $\mu = 3.10^{-4}$, regularization parameter $\lambda = 3.10^{-3}$, and threshold 2 $\lambda' = 2.06^{-9}$. For the validation SPHML model, there are two measures, including the mean classification accuracy with standard error and the receiving operating characteristic (ROC) curve from the ten-fold cross validation to validate our method.

[1] http://vis-www.cs.umass.edu/lfw/results.html.

[2] https://www.cs.tau.ac.il/~wolf/ytfaces/.

To the accuracy (*Acc*) of each fold, assume the take mean average (*MeanAcc*) for the 10-fold as follow :

$$MeanAcc = \frac{\sum_{i=1}^{n=10} Acc}{n} \tag{16}$$

$$Acc = \frac{TP + TN}{P + N} \tag{17}$$

where TP is the true positives and TN is the true negatives.
n = Total folds number
P = True positive + False negative.
N=True negative + False positive.

4.2 Comparison with state-of-the-art methods on LFW dataset

Table 1 shows that SPHML model outperforms other methods on the LFW dataset with restricted images setting. The result can be classified into two classes:

(1) Method-based metric learning such as PCCA [32], STFRD + PMML [50], CSML + SVM [34], and DDML [13].
(2) Method-based descriptors such as pose adaptive filter (PAF) [7] and fisher vector faces [37].

Figure 4 shows ROC curves between our model SPHML and state-of-the-art methods on the LFW dataset with the number of descriptors NoD. Although other methods have more than one descriptor, the performance of model SPHML with one descriptor SIFT is 95.43 ± 2.10, which indicates the superiority of our model's accuracy on unconstrained environmental images. This is an improvement over the current method, DDML, which has six descriptors by 4.75% increase of the mean verification rate. This proves the high performance of the SPL function with hierarchical deep metric learning model.

4.3 Comparison with deep learning methods on LFW dataset

Almost all the results generated from the supervised methods are more reliable and accurate as compared to the results generated from the unsupervised method. One of the reasons that explains why the supervised method generates reliable and accurate results is that the input data is well known and labeled, which means that the machine will only analyze the hidden patterns. This is unlike the unsupervised method of learning where the machine has to define and label the input data before determining the hidden patterns and functions. Table 2 displays the result of model SPHML compared

to two main proposed deep learning methods, which are commonly:

(1) Unsupervised deep learning methods such as CDBN [16]; we can see here that SPHML outperforms the different deep learning methods. The reason is that CDBN is an unsupervised deep learning method, while our model is supervised learning.
(2) Supervised deep learning methods such as DNLML-ISA [2] and DDML [13], SPHML model outperforms the other deep learning supervised method. The reason is that SPHML selects importance samples from easy to hard, relying on the proposed weighted loss function and then utilizes hierarchical nonliner functions on image pairs. We conducted AUC that represents the degree or measure of separability. The higher score our AUC = 0.96 on the LFW dataset indicates that our model can effectively distinguish between distinct classes. In addition, the significance result for evaluation indexes as Recall = 0.75 and F1 = 0.80 that indicates the distinguished information between images can be exploited, which makes our model more robust to accuracy with the other state-of-art models. Altogether, it is evident that SPHML can be a useful framework in unconstrained environments.

4.4 Comparison with state-of-the-art methods on YTF dataset

Table 3 shows that our SPHML model outperforms the other methods in terms of the mean verification rate with restricted images on YTF dataset with the number of descriptors NoD. The results of different methods, such as MBGS (LBP) [47], STFRD + PMML, MBGS + SVM (LBP) [6], VSOF + OSS [30], DDML [13], PHL + SILD (LBP) [26], and APEM (fusion) [5], reveal that the performance of SPHML with one descriptor LBP under the restricted image protocol is 85.26 ± 0.26, which is an improvement from the current method as DDML, which has more than one descriptor by 2.92% increase of the mean verification rate.

Figure 5 shows ROC function curves between our model, SPHML, and other models on the YTF dataset; performance of SPHML model with LBP descriptor is 85.26 ± 0.24, which indicates the robustness of our model's accuracy, which is an improvement from the current method (DDML). We utilized AUC, which indicates the measure of separability. Our score AUC = 0.80 on the YTF dataset demonstrates that SPHML can effectively discriminative between classes. In addition, the significant result under the evaluation indexes Recall = 0.633 and F1 = 0.70 shows that SPHML can achieve highly competitive performance with the other video face verification methods that indicates the discriminative information between video faces can be exploited which makes SPHML model more robust to accuracy.

Table 1 Comparisons of the mean rate and standard error % verification with the state-of-the-art results with the number of descriptors used in each method NoD on the LFW dataset under the restricted images setting

Methods	Number of descriptors NoD	Accuracy
PCCA	1	83.80 ± 0.40
FISHER VECTOR FACE	1	87.47 ± 1.49
PAF	1	87.77 ± 0.51
DDML (SIFT)	1	87.83 ± 0.93
CSML + SVM	6	88.00 ± 0.37
STFRD + PMML	8	89.35 ± 0.50
DDML (combined)	6	90.68 ± 1.41
SPHML (SIFT)	1	**95.43 ± 2.10**

Bold value represents significant of mean rate and standard error (%) with differences between SPHML model and other models

4.5 Comparison with conventional video-based methods on the YTF dataset

Table 4 compares SPHML model with common video face verification methods on the YTF dataset. Affine hull based image set distance (AHISD) [4], convex hull based image set distance (CHISD) [4], sparse approximated nearest points (SANP) [28], manifold distance (MMD) [43], and discriminative deep metric learning (DDML) [13], discriminant-analysis of canonical correlations (DCC) [23]. The importance samples from easy to hard, relying on the proposed weighted loss function by SPL function and utilizes hierarchical metric learning can add more robust for accuracy. It can be seen that SPHML model outperforms with other methods, in respect of the mean verification rate and standard error on video face verification.

4.6 Effect of the activation function

We examined the result of the activation functions in SPHML model, the effect of tan*h* function with the other two common activation functions, viz., *non-saturating sigmoid* and *sigmoid*, where the descriptor SIFT feature is used on the LFW and LBP descriptor used on the YTF dataset. The tan*h* activation function produced good results among all of them in both datasets.

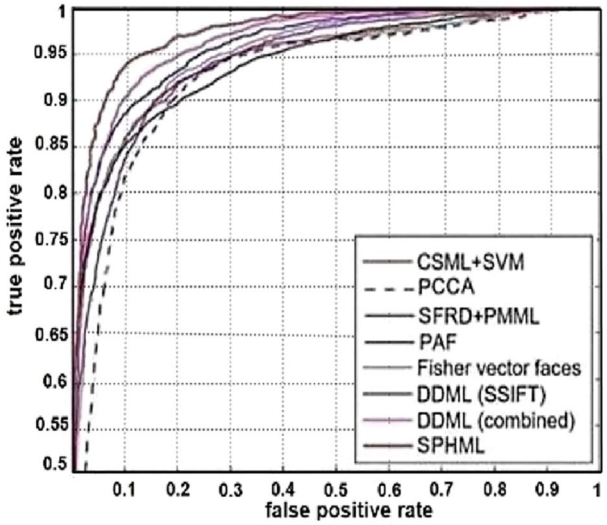

Fig. 4 ROC curve result of comparison with other methods on the LFW dataset

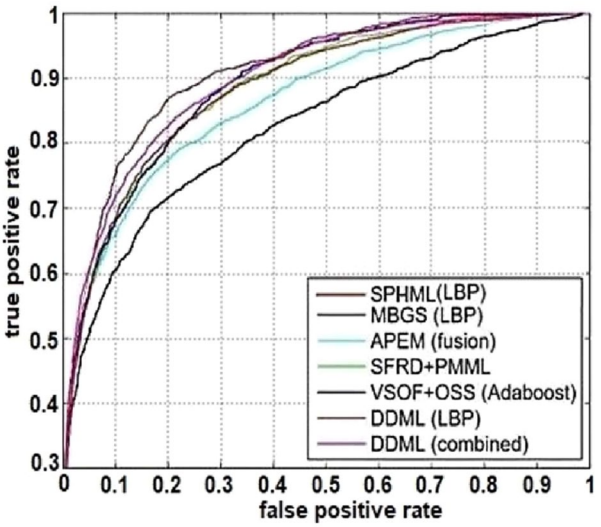

Fig. 5 ROC curve comparison of results of other methods on YTF dataset

Table 2 Result of comparison with deep learning methods on the LFW dataset

Deep learning methods	Based on	Accuracy
CDBN	Unsupervised	86.88 ± 0.62
CDBN + Hand-crafted	Unsupervised	87.77 ± 0.62
DNLML-ISA(SSIFT)	Supervised	86.17 ± 0.40
DNLML-ISA	Supervised	88.50 ± 0.40
DDML (SSIFT)	Supervised	87.83 ± 0.93
DDML (combined)	Supervised	90.68 ± 1.41
SPHML (SIFT)	Supervised	$\mathbf{95.43 \pm 2.10}$

Bold value represents significant of mean rate and standard error (%) with differences between SPHML model and other models

Table 3 Comparisons of the mean rate and standard error (%) verification with other restricted images on the YTF dataset with the number of descriptors NoD

Methods	Number of descriptors NoD	Accuracy
MBGS (LBP)	1	76.40 ± 1.80
APEM (LBP)	1	77.44 ± 1.46
STFRD+PMML	6	79.48 ± 2.52
VSOF+OSS	1	79.70 ± 1.80
DDML (LBP)	1	81.26 ± 1.63
DDML (combined)	6	82.34 ± 1.47
SPHML (LBP)	1	$\mathbf{85.26 \pm 0.24}$

Bold value represents significant of mean rate and standard error (%) with differences between SPHML model and other models

Table 4 Comparisons of the mean rate and standard error (%) verification with the state-of-the-art results for video-based face verification methods on the YTF dataset

Methods	Accuracy
SANP	63.74 ± 1.69
MMD	64.96 ± 1.00
CHISD	66.24 ± 1.70
AHISD	66.50 ± 2.03
DCC	70.84 ± 1.57
DML (LBP)	81.26 ± 1.63
DDML (LBP)	82.34 ± 1.47
SPHML (LBP)	$\mathbf{85.26 \pm 0.24}$

Bold value represents significant of mean rate and standard error (%) with differences between SPHML model and other models

5 Conclusion

In this paper, we proposed a novel model for robust hierarchical metric learning called SPHML. Our model is inspired by human education that can train the samples from easy to more complex in a graded manner, depending on weight noise for each sample, then exploited the hierarchical non-linear functions in a deep learning. The SPHML model thus accomplishes a very competitive verification performance, vis-à-vis the other methods. The robustness of accuracy on the labeled face in wild (LFW) dataset is (95.43 ± 2.10) and (85.26 ± 0.24) on YouTube Face (YTF) datasets.

In future work, we plan to extend our proposal method to fill the semantic gap between metric learning and other

visual deep applications such as image verification and classification Moreover, we plan to compare the analysis of our model SPHML with purification techniques and removal noise methods.

Acknowledgements This work was supported by the National Natural Science Foundation of China under Grants 61876127 and 61732011, Natural Science Foundation of Tianjin Under Grants 17JCZDJC30800, Key Scientific and Technological Support Projects of Tianjin Key R&D Program 18YFZCGX00390 and 18YFZCGX00680.

References

1. Basu S, Christensen J (2013) Teaching classification boundaries to humans. In: Twenty-seventh AAAI conference on artificial intelligence
2. Bengio Y, Louradour J, Collobert R, Weston J (2009) Curriculum learning. In: Proceedings of the 26th annual international conference on machine learning, ICML 2009, Montreal, Quebec, Canada, June 14–18, 2009
3. Cai X, Wang C, Xiao B, Xue C, Ji Z (2012) Deep nonlinear metric learning with independent subspace analysis for face verification. In: ACM international conference on multimedia
4. Cevikalp H, Triggs B (2010) Face recognition based on image sets. In: Computer vision & pattern recognition
5. Chatzis S (2014) Dynamic Bayesian probabilistic matrix factorization. In: Twenty-eighth AAAI conference on artificial intelligence
6. Davis JV, Kulis B, Jain P, Sra S, Dhillon IS (2007) Information-theoretic metric learning. In: ICML 07: international conference on machine learning
7. Dong Y, Zhen L, Li SZ (2013) Towards pose robust face recognition. In: Computer vision & pattern recognition
8. Guillaumin M, Verbeek JJ, Schmid C (2009) Is that you? Metric learning approaches for face identification. In: IEEE international conference on computer vision
9. Guo H, Zhu K, Tang M, Wang J (2019) Two-level attention network with multi-grain ranking loss for vehicle re-identification. IEEE Trans Image Process 28(9):4328–4338
10. Hinton GE, Salakhutdinov RR (2006) Reducing the dimensionality of data with neural networks. Science 313(5786):504–507
11. Hinton GE, Osindero S, Teh YW (2014) A fast learning algorithm for deep belief nets. Neural Comput 18(7):1527–1554
12. Hormozi H, Hormozi E, Nohooji HR (2012) The classification of the applicable machine learning methods in robot manipulators. Int J Mach Learn Comput 2(5):560
13. Hu J, Lu J, Tan YP (2014) Discriminative deep metric learning for face verification in the wild. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 1875–1882
14. Huai M, Miao C, Li Y, Suo Q, Su L, Zhang A (2018) Metric learning from probabilistic labels. In: Proceedings of the 24th ACM SIGKDD international conference on knowledge discovery & data mining, pp 1541–1550
15. Huang GB, Mattar M, Berg T, Learned-Miller E (2008) Labeled faces in the wild: a database for studying face recognition in unconstrained environments. In: Workshop on faces in 'real-life' images: detection, alignment, and recognition, Marseille, France, October 2008. Erik Learned-Miller and Andras Ferencz and Frédéric Jurie. https://hal.inria.fr/inria-00321923
16. Jiang L, Meng D, Yu S-I, Lan Z, Shan S, Hauptmann A (2014a) Self-paced learning with diversity. In: Ghahramani Z, Welling M, Cortes C, Lawrence ND, Weinberger KQ (eds) Advances in neural information processing systems, vol 27, pp 2078–2086. Curran Associates, Inc. http://papers.nips.cc/paper/5648-self-paced-learning-with-diversity.pdf
17. Jiang L, Meng D, Yu S-I, Lan Z, Shan S, Hauptmann A (2014b) Self-paced learning with diversity. In: Advances in neural information processing systems, pp 2078–2086
18. Jiang L, Meng D, Zhao Q, Shan S, Hauptmann AG (2015) Self-paced curriculum learning. In: Twenty-ninth AAAI conference on artificial intelligence
19. KalatehJari EH, Hosseini MM, Gharahbagh AA (2012) Image registration based on a novel enhanced scale invariant geometrical feature. Int J Mach Learn Comput 2(5):667
20. Kaya M, Bilge HS (2019) Deep metric learning: a survey. Symmetry 11(9):1066
21. Keyvanpour M, Izadpanah N, Karbasforoushan H (2012) Classification and evaluation of high-dimensional image indexing structures. Int J Mach Learn Comput 2(3):252
22. Khan F, Mutlu B, Zhu J (2011) How do humans teach: on curriculum learning and teaching dimension. In: Advances in neural information processing systems, pp 1449–1457
23. Kim T-K, Kittler J, Cipolla R (2007) Discriminative learning and recognition of image set classes using canonical correlations. IEEE Trans Pattern Anal Mach Intell 29(6):1005–1018
24. Kulis B et al (2012) Metric learning: a survey. Found Trends Mach Learn 5(4):287–364
25. Kumar MP, Packer B, Koller D (2010) Self-paced learning for latent variable models. In: Advances in neural information processing systems 23: 24th annual conference on neural information processing systems, Proceedings of a meeting held 6–9 December 2010. Vancouver, British Columbia, Canada, p 2010
26. Kwok JT, Tsang IW (2003) Learning with idealized kernels. In: Proceedings of the 20th international conference on machine learning ICML-03, pp 400–407
27. Law MT, Thome N, Cord M (2017) Learning a distance metric from relative comparisons between quadruplets of images. Int J Comput Vis 121(1):65–94
28. Le QV, Zou WY, Yeung SY, Ng AY (2011) Learning hierarchical invariant spatio-temporal features for action recognition with independent subspace analysis. In: Computer vision & pattern recognition
29. Lu J, Hu J, Zhou J (2017) Deep metric learning for visual understanding: an overview of recent advances. IEEE Signal Process Mag 34(6):76–84
30. Marc, Huang FJ, Boureau Y, Lecun Y (2007) Unsupervised learning of invariant feature hierarchies with applications to object recognition. In: IEEE conference on computer vision & pattern recognition
31. Meng D, Xu Z, Zhang L, Zhao J (2013) A cyclic weighted median method for l1 low-rank matrix factorization with missing entries. In: Twenty-seventh AAAI conference on artificial intelligence
32. Mignon A (2012) PCCA: a new approach for distance learning from sparse pairwise constraints. In: Computer vision & pattern recognition
33. Méndez-Vázquez H, Martínez-Díaz Y, Chai Z (2013) Volume structured ordinal features with background similarity measure for video face recognition. In: International conference on biometrics
34. Nguyen HV, Bai Li (2010) Cosine similarity metric learning for face verification. In: Asian conference on computer vision
35. Nick W, Asamene K, Bullock G, Esterline A, Sundaresan M (2015) A study of machine learning techniques for detecting and classifying structural damage. Int J Mach Learn Comput 5(4):313
36. Song HO, Xiang Y, Jegelka S, Savarese S (2016) Deep metric learning via lifted structured feature embedding. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 4004–4012

37. Roth PM, Wohlhart P, Hirzer M, Kostinger M, Bischof H (2012) Large scale metric learning from equivalence constraints. In: IEEE conference on computer vision & pattern recognition
38. Salido JAA, Ruiz C Jr (2018) Using deep learning to detect melanoma in dermoscopy images. Int J Mach Learn Comput 8(1):61–68
39. Schroff F, Kalenichenko D, Philbin J (2015) Facenet: a unified embedding for face recognition and clustering. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 815–823
40. Sohn Kihyuk (2016) Improved deep metric learning with multi-class n-pair loss objective. Adv Neural Inf Process Syst 29:1857–1865
41. Tang Y, Yang Y-B, Gao Y (2012) Self-paced dictionary learning for image classification. In: Proceedings of the 20th ACM international conference on multimedia, pp 833–836
42. Taylor GW, Fergus R, Lecun Y, Bregler C (2010) Convolutional learning of spatio-temporal features. In: European conference on computer vision
43. Wang R, Shan S, Chen X, Wen G (2008) Manifold-manifold distance with application to face recognition based on image set. In: IEEE computer society conference on computer vision & pattern recognition
44. Weinshall D, Cohen G, Amir D (2018) Curriculum learning by transfer learning: theory and experiments with deep networks. arXiv preprint arXiv:1802.03796
45. Wolf L, Hassner T, Maoz I (2011) Face recognition in unconstrained videos with matched background similarity. In: CVPR 2011. IEEE, pp 529–534
46. Wolf L, Hassner T, Taigman Y (2009) Similarity scores based on background samples. In: Asian conference on computer vision
47. Yeung DY, Chang H (2007) A kernel approach for semisupervised metric learning. IEEE Trans Neural Netw 18(1):141–9
48. Yin X, Chen Q (2016) Deep metric learning autoencoder for nonlinear temporal alignment of human motion. In: 2016 IEEE international conference on robotics and automation (ICRA), pp 2160–2166
49. Zhang D, Meng D, Han J (2016) Co-saliency detection via a self-paced multiple-instance learning framework. IEEE Trans Pattern Anal Mach Intell 39(5):865–878
50. Zhen C, Wen L, Dong X, Shan S, Chen X (2013) Fusing robust face region descriptors via multiple metric learning for face recognition in the wild. In: Computer vision & pattern recognition