IET Image Processing

Research Article

IET Journals
The Institution of Engineering and Technology

Collaborative similarity metric learning for face recognition in the wild

ISSN 1751-9659
Received on 7th May 2019
Revised 27th January 2020
Accepted on 2nd March 2020
E-First on 22nd May 2020
doi: 10.1049/iet-ipr.2019.0510
www.ietdl.org

Batuhan Gundogdu[1] ✉, *Michael J. Bianco*[2]

[1]*Electrical Engineering Department, National Defense University, Naval Academy, Istanbul, 34940, Turkey*
[2]*Scripps Institution of Oceanography, University of California San Diego, La Jolla, CA, 92093-0238, USA*
✉ *E-mail: mbgundogdu@dho.edu.tr*

Abstract: Utilising different representations of face images is known to be helpful in face recognition. In this study, the authors propose two fusion techniques that make use of multiple face image features by collaboratively training a similarity metric learner, based on Siamese neural networks. This training procedure takes two (or possibly more) features of two face images and outputs a similarity score that depicts whether the faces belong to the same person or not. The authors investigate two approaches of collaborative similarity metric learning (CoSiM), both of which are based on training Siamese neural networks jointly, as a means of early fusion. The experiments are employed on hand-crafted features such as scale-invariant feature transform (SIFT) and variants of the local binary pattern (LBP), on the YouTube Faces and the Labeled Faces in the Wild data sets. The authors provide theoretical and empirical comparisons of the proposed models against the related methods in the literature. It is shown that the proposed technique improves on the verification accuracy, compared to single feature-based baselines. By only utilising simple features like SIFT and LBP, the proposed techniques are shown to yield comparable results to the state of the art techniques, which depend on deep convolutional architectures or higher level features.

1 Introduction

The astonishing growth in the amount of digital content volume, brought by the ever-increasing usage of social media, has motivated the recent challenge of working on face images in the wild. In the wild images possess an increased amount of variations due to lighting, facial expression, pose, age, scale, accessories, occlusions, background and misalignment, rather than just frontal passport pictures. Such a task is generally referred to as unconstrained face recognition or verification in the literature. In this paper, we propose a collaborative (or joint) distance metric learning (DML) methodology for the task of unconstrained face verification. In face verification, the systems are required to determine if two images belong to the same person or not. The growing use of face verification technology as the new means of biometric security check-in mobile devices increases the importance of this task.

Since the task of face verification is a pair matching problem, given a pair of face images, a measure of similarity (or dissimilarity) is evaluated to decide if they belong to the same person or not. The primary approach to this problem has concerned representation and DML or similarity metric learning (SML) methodologies to obtain a robust approach for unconstrained face images [1]. Many studies such as [2–6] investigated obtaining better representations and descriptors to facilitate face verification for given dissimilarity measures. Many other studies aimed at learning a better discriminative distance metric [7–10], while some investigated learning both distance metric and image representations together [11–13]. Most of the DML methods focus mainly on learning a linear/nonlinear function of image representations, so that the desired dissimilarity cost is minimised when two (or more) images are inputted to this function.

The most substantial work on the unconstrained face verification was initiated with the YouTube Face (YTF) [14] and Labeled Faces in the Wild (LFW) challenges [15]. Early work on the task-focused on descriptor-based methods to conduct the same versus different discrimination using known distance metrics or classifiers such as linear discriminant analysis or support vector machines (SVMs) [16, 17]. Distance metric learning/SML methodologies had recently been studied in the literature [18] and were brought to the unconstrained face verification task by Guillaumin *et al*. [7]. The primary objective in the DML methodology proposed in [7] is learning a Mahalanobis distance on the given representation space. Similarly, Nguyen and Bai [19] proposed the cosine SML methodology in which the cosine similarity cost is optimised per kinship, in the learned projection space. Cao [20] proposed optimising for a generalised similarity function, obtained by subtracting Euclidean distance measure from an inner product-based similarity measure calculated in two different sub-spaces to be learned. Zheng *et al*. [21] proposed the logistic SML methodology which optimises over the logistic loss function, calculated with respect to the cosine similarity and the kinship label of the images. Schroff *et al*. [22] proposed training deep convolutional neural networks that work directly on the raw images and maps them to Euclidean spaces in which the desired kinship discrimination is achieved. The general application depends on using a single or a concatenation of multiple hand-crafted feature representations of images as inputs. Some studies, however, employ learning these functions from raw images using deep convolutional neural networks [13].

Distance metrics learned on a single feature representation potentially fail to utilise the complementary information brought by different features. The two main approaches to overcome this conundrum can be stated as late and early fusion techniques. In late fusion, the fusion of information takes place at the decision level. Specifically, different similarity values are fused, each of which is obtained using a different feature representation [10]. On the other hand, in the early fusion approach, the different features are fused generally by concatenation, before the training. In this approach, several features are concatenated on the input side and a larger SML model is trained, in order to exploit the information content of various features. The comparison of early and late fusion approaches have been evaluated in several studies in the literature, such as semantic video analysis [23], emotion recognition [24], concept detection [25], multimedia event detection [26] and plant species classification [27]. The general conclusion of all of the work about the early versus late fusion approaches is that early fusion tends to perform better than late fusion. However, the naive concatenation-based early fusion approach generally suffers from

its inability to interpret the geometrical contribution of each feature in the discrimination of two images [28].

In order to approach this problem, this paper offers two methodologies that would learn a similarity metric that is jointly trained, using multiple feature representations of the input images. We name the approach proposed in this paper collaborative SML (CoSiM) since it learns a similarity metric between representations of two face images by the joint training of different features. This joint training is expected to bring a collaboration of different representations.

The literature survey provided in this section extends to Section 3, in which we provide a comparison of the methodology proposed in this paper and the related work in the literature. The detailed literature review is deferred to after Section 2, in which we introduce the methodology and the mathematical details of the proposed approach in this paper. With this set-up, we believe that the discussion of the (dis)similarities of the similar work with this paper and with each other are given in a more useful way. In Section 4, we provide information about the experiments on YTF and LFW data sets along with the experimental results. We also compare the performance of our approach with the state of the art techniques in the literature.

1.1 Contributions of the paper

In this paper, we propose a Siamese neural network-based SML methodology to facilitate joint learning of a similarity metric that exploits multiple feature representations of the input images. The key contributions of the paper are as follows:

• The sigma distance that was recently proposed for speech features in a keyword search task [29], is utilised for face images and compared with the other DML/SML techniques in the literature.
• Two COSiM methodologies are proposed and discussed both mathematically and empirically. For experiments, two image representations: (i) scale-invariant feature transform (SIFT) and (ii) local binary pattern (LBP) are combined and the SML networks are trained jointly to investigate as a means of early fusion. The proposed CoSiM networks are named as mass CoSiM and average CoSiM for the reasons to be provided along with the mathematical interpretations in Section 2.2.
• The proposed COSiM methodologies are tested on two of the most challenging data sets in the literature, the YTFs[14] and LFW [30] data sets, under the image restricted, label-free outside data and image restricted, no outside data paradigms. Results are compared with similar works in the literature and baselines. YTF data set is constructed by video snippets from YouTube and the LFW data set is specifically constructed by pictures captured from news articles on the web, to provide a benchmark for unconstrained face verification. Experiments conducted over ten-fold cross-validation show that the proposed early fusion provides significant improvements on the verification score. On the LFW data set, for which the results of some other works were also available, the proposed system reaches 92.6% validation accuracy when combined with similar systems in the literature. This result is comparable to the state of the art techniques in the literature that use complex deep architectures in the learning.

The paper is organised as follows: In Section 1 we provide an introduction and the literature review of the subject. In Section 2, the methodology is presented as well as the theoretical background. Once the notation and the objectives are set in this section, we discuss in detail the differences and similarities of this work to the literature in Section 3. In Section 4, the experiments are presented after which the conclusions are drawn in Section 5.

2 Methodology

In this section, we provide a mathematical and theoretical background of the model proposed.

2.1 Assumptions and preliminaries

This paper presents an early fusion methodology for the task referred to as SML.

- The 'similarity' in SML is function $f(\boldsymbol{x}, \boldsymbol{y}): \mathscr{R}^d \mathrm{x} \mathscr{R}^d \rightarrow \mathscr{R}$, of the representations of two images $\boldsymbol{x}$ and $\boldsymbol{y} \in \mathscr{R}^d$, such that this function outputs a high value for images that belong to the same person, and a low value for images that belong to different people. For notational simplicity, we denote images by their vector representations $(\boldsymbol{x}, \boldsymbol{y})$ and we call $\boldsymbol{x}$ and $\boldsymbol{y}$ friends if they are extracted from the images of the same person, and *foes* otherwise.

- The 'metric' in SML, is in fact a misnomer. The metric spaces are defined with a distance measure $d(\boldsymbol{x}, \boldsymbol{y})$, that satisfies the axioms: (i) non-negativity, (ii) symmetry, (iii) triangle inequality and (iv) identity of discernibles [31]. On the other hand, since there is no formal definition of a 'similarity metric', axioms for $f(\boldsymbol{x}, \boldsymbol{y})$ are defined per application. Hence, we do not claim that we conform to the axioms of metric spaces when we call the name 'similarity metric'.

- The 'learning' in SML is the essence of this work. SML is actually an optimisation task. Given a set of pairs $\mathscr{X} = (\boldsymbol{x}_t, \boldsymbol{y}_t)$ and their labels r_t denoting if they are friends or foes, for $t = 1 \cdots T$; the goal is to find a function $f(\boldsymbol{x}, \boldsymbol{y})$ such that it is higher for friends than foes for all pairs that are not seen in $\mathscr{X}$. Therefore, a proper generalisation and learning methodology is desired on top of the cost minimisation procedures.

As stated in the introduction section, we propose a methodology for jointly training a combination of several Siamese neural networks, which have shared layers along each feature path. For the sake of simplicity, we will be building our methodology using two of the most common image representations, i.e. (i) SIFT [32, 33] and (ii) LBP [34, 35]. However, the methodology can simply be generalised to using pairs of other features and using multiple features.

2.2 From similarity metric learning to coSiM

The CoSiM approach follows the Siamese neural network structure called the *sigma distance*, which has recently been proposed for speech features to be used in dynamic time warping [36]. Given $\boldsymbol{x}$, $\boldsymbol{y} \in \mathscr{R}^d$, the similarity measure for this pair of feature vectors is obtained by projection onto a new space and calculating the inner product (4). This Mahalanobis inner product is then applied to a sigmoid function so that the similarity function represents a measure of the probability of the input vectors being friends [29]:

$$f(\boldsymbol{x}, \boldsymbol{y}) = \sigma(\boldsymbol{x}^{\mathrm{T}} \boldsymbol{W}^{\mathrm{T}} \boldsymbol{W} \boldsymbol{y} + b) \tag{1}$$

where

$$\sigma(z) = \frac{1}{1 + \mathrm{e}^{-z}}. \tag{2}$$

$\sigma(z)$ is the sigmoid function that reaches 1 and 0 asymptotically as z reaches $+\infty$ and $-\infty$, respectively. This basis SML methodology, which works for a single feature can be seen graphically in Fig. 1.

In training, we use the training set of triplets $(\boldsymbol{x}_t, \boldsymbol{y}_t, r_t)$ where r_t is the label indicating the friendship of the inputs $\boldsymbol{x}_t$ and $\boldsymbol{y}_t$. As for the labels r_t, we use 1 for friends and 0 for foes and minimise the cross-entropy (CE) objective function. If we call the total set of parameters in the network Θ, the CE cost function is defined as:

$$J_{\mathrm{CE}}(\Theta; \boldsymbol{x}_t, \boldsymbol{y}_t, r_t) = -r_t \log(f) - (1 - r_t) \log(1 - f) \tag{3}$$

where $f(\boldsymbol{x}_t, \boldsymbol{y}_t)$ is expressed as f for the sake of simplicity.

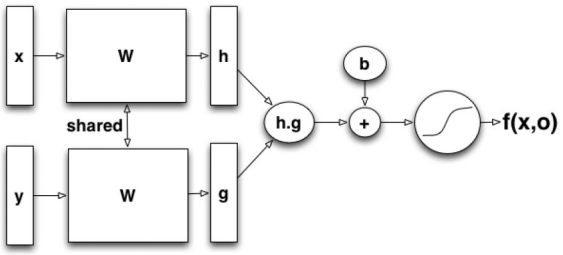


Fig. 1 *The single feature SML model with from [36]. The learned parameters are* $\boldsymbol{W}$ *and* b

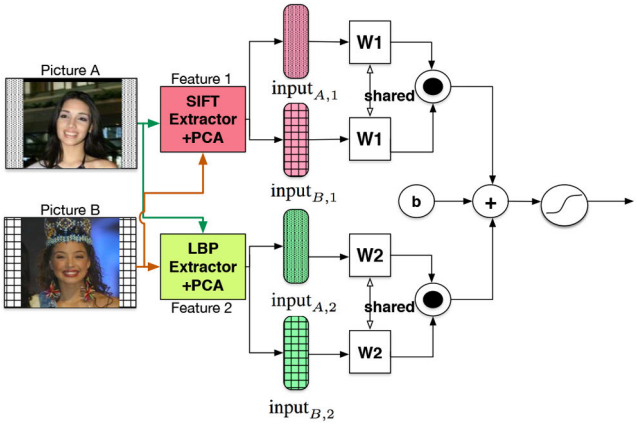


Fig. 2 *Flowchart of the mass CoSiM model*

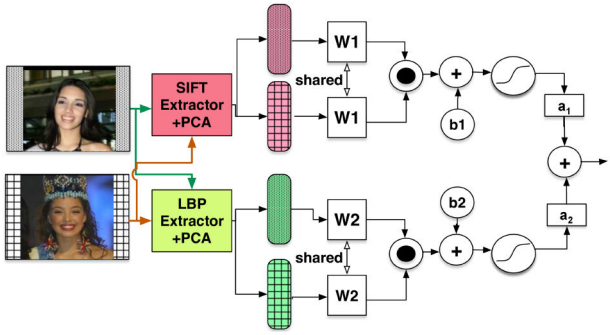


Fig. 3 *Flowchart of the average CoSiM model*

The CoSiM model, on the other hand, takes two (or more) feature representations of the images as input and outputs a similarity function. Let $\boldsymbol{x}_1, \boldsymbol{y}_1 \in \mathscr{R}^{d_1}$ and $\boldsymbol{x}_2, \boldsymbol{y}_2 \in \mathscr{R}^{d_2}$ be the first and second pair of features for two images, we aim to learn a function $f(\boldsymbol{x}_1, \boldsymbol{y}_1, \boldsymbol{x}_2, \boldsymbol{y}_2)$ that exploits the complementary information content in both feature representations. For this, we propose two different CoSiM network topologies: (i) mass CoSiM and (ii) average CoSiM.

2.3 Mass CoSiM

The first topology we refer to as the mass CoSiM merges the two input after the inner product layer. There are two Siamese neural networks that are shared across the same feature representation. After both input features are projected onto their corresponding subspaces, the inner products are added before the sigmoid function application. The sigma distance depends on the similarity value obtained by the weighted inner product. These similarity values are added together in mass CoSiM, just like the accumulation of mass. The mass CoSiM architecture can be seen in Fig. 2. The similarity value, obtained by the mass CoSiM network can be expressed thus:

$$f(\boldsymbol{x}_1, \boldsymbol{y}_1, \boldsymbol{x}_2, \boldsymbol{y}_2) = \sigma(\boldsymbol{x}_1^{\mathrm{T}}\boldsymbol{W}_1^{\mathrm{T}}\boldsymbol{W}_1\boldsymbol{y}_1 + \boldsymbol{x}_2^{\mathrm{T}}\boldsymbol{W}_2^{\mathrm{T}}\boldsymbol{W}_2\boldsymbol{y}_2 + b) \tag{4}$$

Using the CE cost function, we provide the gradient calculation derivations here for both of the CoSiM methodologies in order to facilitate the mathematical interpretations and intuitions that are given in the following sections:

$$\begin{aligned}\nabla_b J &= \frac{\mathrm{d}J}{\mathrm{d}f}\frac{\mathrm{d}f}{\mathrm{d}z}\frac{\mathrm{d}z}{\mathrm{d}b} \\ &= \frac{r-f}{f(1-f)} \cdot f \cdot (1-f) = r - f\end{aligned} \tag{5}$$

$$\begin{aligned}\nabla_{\boldsymbol{W}_i} J &= \frac{\mathrm{d}J}{\mathrm{d}f}\frac{\mathrm{d}f}{\mathrm{d}z}\frac{\mathrm{d}z}{\mathrm{d}\boldsymbol{W}_i} \\ &= (r-f)\boldsymbol{W}_i(\boldsymbol{x}_i\boldsymbol{y}_i^{\mathrm{T}} + \boldsymbol{y}_i\boldsymbol{x}_i^{\mathrm{T}})\end{aligned} \tag{6}$$

where

$$z = \boldsymbol{x}_1^{\mathrm{T}}\boldsymbol{W}_1^{\mathrm{T}}\boldsymbol{W}_1\boldsymbol{y}_1 + \boldsymbol{x}_2^{\mathrm{T}}\boldsymbol{W}_2^{\mathrm{T}}\boldsymbol{W}_2\boldsymbol{y}_2 + b \tag{7}$$

2.4 Mathematical interpretation of mass CoSiM

It is observed that the projection matrix is updated to the direction decided by the corresponding input feature (6). In other words, $\boldsymbol{W}_1$ is updated to the direction $\boldsymbol{W}_1(\boldsymbol{x}_1\boldsymbol{y}_1^{\mathrm{T}} + \boldsymbol{y}_1\boldsymbol{x}_1^{\mathrm{T}})$ and vice versa. The main advantage we obtain in adding before the sigmoid is seen on the size of this step. The update rule of the network is very similar to the single feature SML network, except with one difference. The step size of the update for each shared parameter is decided by the term $(r - f)$, which carries the collective decision information using both features. Also, the bias term (b) is updated by the joint decision of the two input pairs, hence it is less susceptible to outliers and noisy decisions.

2.5 Average CoSiM

The second topology conducts the merging of the different features after the individual decisions of the two Siamese neural networks are made. We refer to this system as the average CoSiM since a weighted sum of the individual decisions is taken to obtain the ultimate output. The two Siamese neural networks are still shared across the same feature representation. The sigma similarities for each feature are calculated and their weighted sum is taken to obtain a joint similarity measure. The average CoSiM architecture can be seen in Fig. 3.

The mathematics of the model and the training is as follows:

$$g(\boldsymbol{x}_1, \boldsymbol{y}_1, \boldsymbol{x}_2, \boldsymbol{y}_2) = a_1 f_1(\boldsymbol{x}_1, \boldsymbol{y}_1) + a_2 f_2(\boldsymbol{x}_2, \boldsymbol{y}_2) \tag{8}$$

where $a_1 + a_2 = 1$. a_i are the weight parameters, which can be taken as uniform or learned through cross-validation. In this paper, we took $a_1 = a_2 = 0.5$

$$f_i(\boldsymbol{x}_i, \boldsymbol{y}_i) = \sigma(\boldsymbol{x}_i^{\mathrm{T}}\boldsymbol{W}_i^{\mathrm{T}}\boldsymbol{W}_i\boldsymbol{y}_i + b_i) \tag{9}$$

The gradients with respect to the CE cost function is calculated as follows:

$$\begin{aligned}\nabla_{b_i} J &= \frac{\mathrm{d}J}{\mathrm{d}g}\frac{\mathrm{d}g}{\mathrm{d}f_i}\frac{\mathrm{d}f_i}{\mathrm{d}z_i}\frac{\mathrm{d}z_i}{\mathrm{d}b_i} \\ \nabla_{\boldsymbol{W}_i} J &= \frac{\mathrm{d}J}{\mathrm{d}g}\frac{\mathrm{d}g}{\mathrm{d}f_i}\frac{\mathrm{d}f_i}{\mathrm{d}z_i}\frac{\mathrm{d}z_i}{\mathrm{d}\boldsymbol{W}_i}\end{aligned} \tag{10}$$

where

$$z_i = \boldsymbol{x}_i^{\mathrm{T}}\boldsymbol{W}_i^{\mathrm{T}}\boldsymbol{W}_i\boldsymbol{y}_i + b_i \tag{11}$$

Equation (10) follows the fact that the cross-derivatives are zero, that is

$$\frac{\mathrm{d}f_i}{\mathrm{d}z_j} = 0, \quad \frac{\mathrm{d}z_i}{\mathrm{d}\boldsymbol{W}_j} = \boldsymbol{0} \quad \text{and} \quad \frac{\mathrm{d}z_i}{\mathrm{d}b_j} = 0 \quad \text{for} \quad j \neq i$$

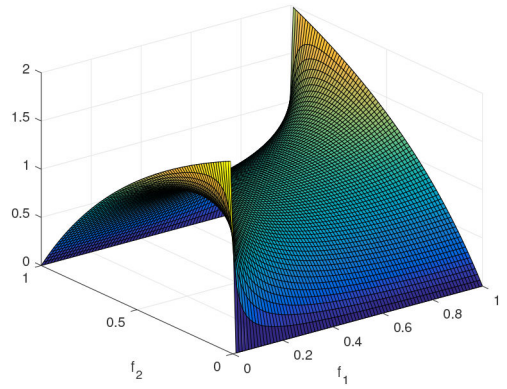


Fig. 4 *Surface plot of the fusing coefficient* $\eta(f_1, f_2)$

By taking each partial derivative, we get

$$\nabla_{b_i} J = \frac{r-g}{g(1-g)} \cdot a_i \cdot f_i \cdot (1-f_i) \tag{12}$$

$$\nabla \boldsymbol{W}_i = \frac{r-g}{g(1-g)} \cdot a_i \cdot f_i \cdot (1-f_i) \boldsymbol{W}_i (\boldsymbol{x}_i \boldsymbol{y}_i^{\mathrm{T}} + \boldsymbol{y}_i \boldsymbol{x}_i^{\mathrm{T}}) \tag{13}$$

2.6 Mathematical interpretation of average CoSiM

The term $(r-g)$ that appear in equations (12) and (13) also exists in bare SML [36] and mass-CoSiM in equations (5) and (6). It basically decides the size (and the direction) of the update step. In other words, if the two input vectors are friends but the system output is close to zero, the bias is increased (as much as the difference), and decreased otherwise. For the shared matrices, the projected outer products (6) are added or subtracted by this scale.

In average CoSiM, however, the value $(r-g)$ is scaled with the following term:

$$\begin{aligned} \eta(f_i, f_j) &= a_i \frac{f_i(1-f_i)}{g(1-g)} \\ &= a_i \frac{f_i(1-f_i)}{(a_1 f_1 + a_2 f_2)(1 - a_1 f_1 - a_2 f_2)} \end{aligned} \tag{14}$$

It is not straightforward to see how η contributes to the update. Apart from the pre-decided or learned weights a_i, the fractional term, η governs the update for each channel in the Siamese neural network, based on their decisions.

When $f_1 = f_2$, in other words the two Siamese networks are in perfect accordance, $\eta = 1$ for all values. For off-diagonal values where $f_1 \neq f_2$, the coefficient η reinforces or annihilates the step according to the balance between the two decisions. Given that $0 \leq f_i \leq 1$ due to the sigmoid layer, we can observe the magnitude of this term for various values of f_1 and f_2 for a better interpretation. Taking $a_1 = a_2 = 0.5$, the surface plot of $\eta(f_1, f_2)$ can be seen in Fig. 4.

Particularly, when $f_1 < f_2$ and both of the values are close to zero, it means that the one feature (say f_1) is potentially detecting correctly and it has a large coefficient. Likewise, when $f_1 > f_2$ and both of the values are close to one, the coefficient is again >1 (yellow edges on the surface curve). On the other hand, when there is a discrepancy between f_1 and f_2 scores, the coefficient η reduces as much as the discrepancy. The blue edges that touch the zero level depict such regions.

This feature of the automatic scaling of the Siamese networks brought by average CoSiM is the reason for its name. Due to its particularly automated choice of update steps, average CoSiM possesses a slow and distinct learning curve. The comparison of the learning curves along with the training details will be provided in the next section.

Table 1 Mass CoSiM neural network topology

Layer	Input shape	Output shape	No. of params	Connected to
$\text{input}_{A,1}$	300	—	0	—
$\text{input}_{A,2}$	300	—	0	—
$\text{input}_{B,1}$	300	—	0	—
$\text{input}_{B,2}$	300	—	0	—
batchnorm1	300×2	300×2	1200 $(\mu, \sigma)_1$	$\text{input}_{A,1}$&$\text{input}_{B,1}$
batchnorm2	300×2	300×2	1200 $(\mu, \sigma)_2$	$\text{input}_{A,2}$&$\text{input}_{B,2}$
dropout1	300×2	300×2	0	batchnorm1
dropout2	300×2	300×2	0	batchnorm2
shared1	300×2	300×2	90,000 ($\boldsymbol{W}_1$)	dropout1
shared2	300×2	300×2	90,000 ($\boldsymbol{W}_2$)	dropout2
dot1	300×2	1	0	shared1
dot2	300×2	1	0	shared2
mass	2	1	0	dot1&dot2
sigmoid	1	1	1 (b)	mass

2.7 Training topologies and implementation details

In our implementation, the two inputs (SIFT and LBP) are whitened by principal component analysis and dimension reduction is applied to 300 dimensions for both features. This preprocessed data was obtained from [10]. In each of the two CoSiM networks, we added a batch normalisation before the shared layers for faster convergence. Batch normalisation forces the input to have zero mean and unity variance.

Furthermore, a dropout rate of 0.7 was added between the batch normalisation and shared layer, which was found to be crucial to avoid over-fitting to the training set. Dropout randomly zeroes a portion of the weights in training. In implementation, we used *Keras* toolkit with *Tensorflow* back-end [37]. In the optimisation procedure we used the *adam* optimiser [38] with the following parameters:

- $\alpha = 0.001$ (learning rate)
- $\beta_1 = 0.9$ (first moment estimate exponential decay)
- $\beta_2 = 0.999$ (second moment estimate exponential decay)
- $\epsilon = 10^{-8}$ (denominator de-nullifier)
- $\lambda = 0.001$ (learning rate decay)

The scalar weights were initialised with random Gaussian distributed numbers. For the shared matrices, the experimental analyses showed that better convergence points were obtained when they are initialised with identity matrices for the corresponding size. We also applied an early stopping with a patience parameter of 1000 epochs to avoid overtraining. The neural network topology summaries for the mass-CoSiM and average-CoSiM training are given in Tables 1 and 2, respectively.

The training epoch behaviour of the CoSiM models can be seen in Fig. 5. The figure demonstrates the change of validation accuracy with respect to training epochs for the mass-CoSiM, average-CoSiM as well as the baseline single feature-based SML training. Since the early stopping trick is implemented, each model lasts for different time spans. It can be seen that the proposed mass-CoSiM training achieves significantly higher validation accuracy when compared to the baseline systems.

Average-CoSiM, on the other hand, exhibits an interesting training behaviour. The convergence is late so that it has very poor performance at the time the mass-CoSiM has already converged to its optimum. However, after enough training time, it surpasses the baseline and the mass-CoSiM. This is due to the addition of smart fusion weight allocation, which is also learned by each path in the Siamese neural network.

Table 2 Average CoSiM neural network topology

Layer	Input shape	Output shape	No. of params	Connected to
$\text{input}_{A,1}$	300	—	0	—
$\text{input}_{A,2}$	300	—	0	—
$\text{input}_{B,1}$	300	—	0	—
$\text{input}_{B,2}$	300	—	0	—
batchnorm1	300×2	300×2	1200 $(\mu, \sigma)_1$	$\text{input}_{A,1}$&$\text{input}_{B,1}$
batchnorm2	300×2	300×2	1200 $(\mu, \sigma)_2$	$\text{input}_{A,2}$&$\text{input}_{B,2}$
dropout1	300×2	300×2	0	batchnorm1
dropout2	300×2	300×2	0	batchnorm2
shared1	300×2	300×2	90000 $(\boldsymbol{W}_1)$	dropout1
shared2	300×2	300×2	90000 $(\boldsymbol{W}_2)$	dropout2
dot1	300×2	1	0	shared1
dot2	300×2	1	0	shared2
sigmoid1	1	1	1(b_1)	dot1
sigmoid2	1	1	1(b_2)	dot1
weighted average[a]	1	1	0[a]	sigmoid1&sigmoid2

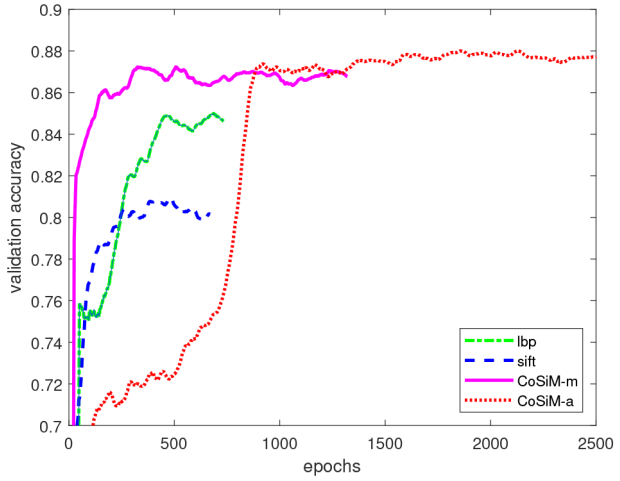


Fig. 5 *Validation accuracy versus epochs for the proposed models and the single feature-based models*

3 Comparison with the similar work in the literature

This section mathematically compares the work with the literature and points out the similarities and dissimilarities. The notations in the original papers are altered to match the notation of this paper to facilitate direct comparison. The derivations and the formulations are coined by the referenced work, we only alter their notation to match our methodology chapter for the sake of the ease of the reader. The comparison of performances with the following works will be provided in the next section along with the experimental results.

Guillaumin *et al*. [7] proposed learning a Mahalanobis metric over the representation space. With the learned Mahalanobis distance (15), they define a probability that the two images belong to the same person to be as shown in (16), which is very similar to (4) in our work.

$$d_{\boldsymbol{W}}(\boldsymbol{x}, \boldsymbol{y}) = (\boldsymbol{x} - \boldsymbol{y})^{\mathrm{T}} \boldsymbol{W} (\boldsymbol{x} - \boldsymbol{y}) \tag{15}$$

$$p = \sigma(b - d_{\boldsymbol{W}}(\boldsymbol{x}, \boldsymbol{y})) \tag{16}$$

This value is then used with the CE cost (3), as in our work. Some of the key differences are that we learn a similarity metric based on projected inner products and use them in the sigmoid function. That way, what corresponds to the $\boldsymbol{W}$ matrix in our work is forced to be symmetric and positive definite.

Nair and Hinton [8] proposed using restricted Boltzmann machines with rectified linear units to extract features that are used in DML. They propose calculating the cosine distance of the extracted features from the two images, which is defined in (17). They also describe the probability of belonging to the same person to be $p = \sigma(-(wd + b))$ with w and b being trainable parameters, which is very similar to [7] and this work

$$d_{\cos}(\boldsymbol{x}, \boldsymbol{y}) = 1 - \frac{\boldsymbol{x}^{\mathrm{T}} \boldsymbol{y}}{\| \boldsymbol{x} \| \, \| \boldsymbol{y} \|} \tag{17}$$

Nguyen and Bai [19] proposed finding a projection space on which the cosine similarity is optimised. The cosine similarity on the new space with the projection matrix $\boldsymbol{W}$, is defined in to be

$$\mathrm{CS}(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{W}) = 1 - d_{\cos}(\boldsymbol{W}\boldsymbol{x}, \boldsymbol{W}\boldsymbol{y}) \tag{18}$$

The shared projection matrix is very similar to the methodology in this work. While in [19] the target-output control is done by the L2-normalisation of the cosine distance, this work uses the sigmoid non-linearity. Furthermore, Nguyen and Bai use a hinge-like loss to obtain an optimised projection matrix, whereas we use CE

$$\begin{aligned} J(\boldsymbol{W}) = & \sum_{t \in \text{friends}} \mathrm{CS}(\boldsymbol{x}_t, \boldsymbol{y}_t, \boldsymbol{W}) \\ & - \alpha \sum_{t \in \text{foes}} \mathrm{CS}(\boldsymbol{x}_t, \boldsymbol{y}_t, \boldsymbol{W}) \\ & - \beta \, \| \boldsymbol{W} - \boldsymbol{W}_0 \|_2^2 \end{aligned} \tag{19}$$

The last term serves for regularisation and it is interesting to note that they used an identity matrix for $\boldsymbol{W}_0$, which turned out to be the empirically obtained best initialiser of this work.

In another SML-based work, Cao *et al*. proposed a convex generalised similarity function, to address the non-convex behaviour of the L2-normalised cosine distance, described in (17) [10, 20]. Using the learned Mahalanobis distance in (15), a projected inner product similarity $s_{\boldsymbol{W}}(\boldsymbol{x}, \boldsymbol{y})$ is defined as follows:

$$s_{\boldsymbol{W}}(\boldsymbol{x}, \boldsymbol{y}) = \boldsymbol{x}^{\mathrm{T}} \boldsymbol{W} \boldsymbol{y} \tag{20}$$

The convex generalised similarity metric is then defined as :

$$f_{(\boldsymbol{W}_1, \boldsymbol{W}_2)}(\boldsymbol{x}, \boldsymbol{y}) = s_{\boldsymbol{W}_1}(\boldsymbol{x}, \boldsymbol{y}) - d_{\boldsymbol{W}_2}(\boldsymbol{x}, \boldsymbol{y}) \tag{21}$$

Using this generalised similarity measure, the learning problem is done via the following constraint optimisation procedure

$$\begin{aligned} \min_{\boldsymbol{W}_1, \boldsymbol{W}_2} \quad & \sum_{\forall \boldsymbol{x}_t, \boldsymbol{y}_t \in \mathcal{X}} \xi_t + \frac{\gamma}{2}(\parallel \boldsymbol{W}_1 - I \parallel_2^2 + \parallel \boldsymbol{W}_2 - I \parallel_2^2), \\ \text{s.t.} \quad & r_t f_{(\boldsymbol{W}_1, \boldsymbol{W}_2)}(\boldsymbol{x}, \boldsymbol{y}) \geq 1 - \xi_t, \\ & \xi_t \geq 0 \end{aligned} \tag{22}$$

where ξ_t's are the slack variables. One other difference from our work is that they take $r_t \in \{-1, 1\}$, while we take $r_t \in \{0, 1\}$ as the target values in optimisation. This dual formulation corresponds to the following cost function, which can be directly compared with our CE and (19):

$$J(\boldsymbol{W}_1, \boldsymbol{W}_2) = \sum_{\forall \boldsymbol{x}_t, \boldsymbol{y}_t} (1 - r_t f_{(\boldsymbol{W}_1, \boldsymbol{W}_2)}(\boldsymbol{x}, \boldsymbol{y})) \tag{23}$$

Hu *et al*. proposed the discriminative deep metric learning (DDML) approach, which is also based on Siamese neural networks [39]. This work proposes taking image representations to another feature subspace, by applying shared neural network operations and then calculating the Euclidean distance on this new space:

$$d_f^2(\boldsymbol{x}, \boldsymbol{y}) = \parallel f(\boldsymbol{x}) - f(\boldsymbol{y}) \parallel_2^2 \tag{24}$$

where

$$\begin{aligned} \boldsymbol{h}^{(1)} &= \sigma(\boldsymbol{W}^{(1)}\boldsymbol{x} + \boldsymbol{b}^{(1)}) \\ \boldsymbol{h}^{(2)} &= \sigma(\boldsymbol{W}^{(2)}\boldsymbol{h}^{(1)} + \boldsymbol{b}^{(2)}) \\ &\cdots \\ f(\boldsymbol{x}) = \boldsymbol{h}^{(M)} &= \sigma(\boldsymbol{W}^{(M)}\boldsymbol{h}^{(M-1)} + \boldsymbol{b}^{(M)}) \end{aligned} \tag{25}$$

The distance value calculated in the new space is used to enforce a margin between *friends* and *foes* via a threshold value. The pairs that do not conform to this margin are designed to contribute to the cost. As the cost function, the generalised logistic loss function is used, which is a smoothed approximation of the rectified linear unit function [40]

$$\begin{aligned} J = & \sum_t \log\Big(1 + \exp(\beta(1 - r_t(\tau - d_f^2(\boldsymbol{x}_t, \boldsymbol{y}_t))))\Big) \\ & + \lambda \sum_m (\parallel \boldsymbol{W}^{(m)} \parallel_2^2 + \parallel \boldsymbol{b}^{(m)} \parallel_2^2) \end{aligned} \tag{26}$$

In (26), the first term is the generalised logistic function, steepness of which is decided by β. The part inside the exponential is contributed to the cost when it is non-zero. This means that a margin (τ) is forced to be kept between the distances of friend and foe pairs. One point to make here is that although DDML, like our study, use the term 'metric', $d_f(\boldsymbol{x}, \boldsymbol{y})$ is not guaranteed to conform to the axioms of metric spaces, on the space that contains $\boldsymbol{x}$ and $\boldsymbol{y}$. Such 'metric' learning approaches are, in fact, representation or space learning approaches such that the target distance measure performs better on the new feature space.

Zheng et al. proposed the logistic similarity learning approach [21], which also aims to learn a shared projection matrix to optimise over the learned cosine similarity (18), similar to [19]. The interesting part is that they use generalised logistic loss function, given by (26), as in [39], which penalises for wrong signed cosine similarities of pairs.

$$\begin{aligned} J = & \sum_t \log\Big(1 + \exp\Big(-\frac{r_t(\mathrm{CS}(\boldsymbol{x}_t, \boldsymbol{y}_t, \boldsymbol{W}) - b)}{T}\Big)\Big) \\ & + \lambda \parallel \boldsymbol{W} - \boldsymbol{W}_0 \parallel_2^2 \end{aligned} \tag{27}$$

The positive constants b and T are used to shift the decision boundary and control the steepness, respectively. Although the original works adopt very different techniques, their objective functions depicted by (26) and (27) are directly comparable. The former aims to maximise the gap between Euclidean distance measures of friend and foe pairs, while the latter aims to push the cosine distances of friends and foes to the limits $\{+1, -1\}$.

Two very recent studies have worked on the idea of jointly training a set of feature vectors for a common similarity metric. Lu *et al*. [13] extended their earlier DDML work, to include the joint training of different features and called it discriminative deep multi-metric learning (DDMML). The function f in (25) is now obtained for K different feature representations. If we call the kth feature representation of an image $\boldsymbol{x}^k$, each DDML learns a feature space representation $\boldsymbol{h}^{(M,k)}$, and a distance measure $d_{fk}(\boldsymbol{x}^k, \boldsymbol{y}^k)$, where M and k denote layer depth and feature representation, respectively. Each DDML in DDMML also learns a weight parameter a_k for each feature. The joint distance is defined as

$$\begin{aligned} d_f^2(\boldsymbol{x}, \boldsymbol{y}) &= \sum_{k=1}^{K} a_k d_{fk}^2(\boldsymbol{x}^k, \boldsymbol{y}^k) \\ &= \sum_{k=1}^{K} a_k \parallel f_k(\boldsymbol{x}) - f_k(\boldsymbol{y}) \parallel_2^2 \end{aligned} \tag{28}$$

The cost function of DDMML involves the weighted sum of each DDML cost function, given with (26), plus a penalty term calculated with the squared differences of d_{fk}'s for the same image. The accordance of different feature representations for the same image is enforced with this term.

Another very relevant work addresses the problem of joint similarity learning as a multi-view metric learning [28]. Similar to DDMML, they define a joint distance measure obtained by the summation of individual and shared neural network activations. Given a pair of different representations of the same image, $\boldsymbol{x}^k$ and $\boldsymbol{x}^\ell$; two projection matrices are defined:

- $\boldsymbol{W}_k^s$: specific projection matrix for feature k, symmetric positive definite
- $\boldsymbol{W}_{k,\ell}^c$: common projection matrix for features k and ℓ, symmetric positive definite

The total distance is then defined as

$$\begin{aligned} d_\Theta^2(\boldsymbol{x}, \boldsymbol{y}) = & \sum_{k=1}^{K} (\boldsymbol{x}_k - \boldsymbol{y}_k)^{\mathrm{T}} \boldsymbol{W}_k^s (\boldsymbol{x}_k - \boldsymbol{y}_k) \\ & + \sum_{k=1}^{K} \sum_{\ell=1}^{L} (\boldsymbol{x}_k - \boldsymbol{y}_\ell)^{\mathrm{T}} \boldsymbol{W}_{k,\ell}^c (\boldsymbol{x}_k - \boldsymbol{y}_\ell) \end{aligned} \tag{29}$$

The methodologies of [13, 28] can be compared to our work in that all three works suggest joint learning of similarity metric, for multiple features. While these two studies work on finding neural network-based representations for Euclidean space, our proposed approach incorporates this idea to the inner product similarity-based realm.

4 Experiments

The experiments are conducted on the YTF data set and the view-2 configuration of the LFW data set. YTF data set consists of video snippets of that are split to frames at 24 frames per second. It was ensured prior to the data set creation that the videos are not still-image slide shows and there are no identical videos. Each image in the video is re-scaled and variants of LBP features are extracted, including centre-symmetric LBP and Four-Patch LBP. We took the LBP and FLBP features in CoSiM training in this work. The verification task is set with the randomly collected 5000 video pairs from the database. These videos are divided into 10 splits, each containing 250 friends and foes each. The goal of the task is to determine if the videos in a pair is friends or foes. Hence, in order to match the first set-up of image verification, we took one random frame from each of the videos provided in the database and conducted CoSiM experiments on these images. The results reported with this data set strictly follows the restricted protocol

Fig. 6 *Samples of same (left) and different (right) person images from LFW data set*

Table 3 List and naming of systems

Exp.Name	Explanation
SML-S	SIFT features as input, single feature SML as in [36], i.e. sigma distance with SIFT features.
SML-L	LBP features as input, single feature SML as in [36], i.e. sigma distance with LBP features.
SML-F	FPLBP features as input, single feature SML as in [36], i.e. sigma distance with LBP features.
CoSiM-m	proposed mass CoSiM structure in Fig. 2
CoSiM-a	proposed average CoSiM structure in Fig. 3
JSML	joint SML: late fusion of the SML-based systems

Table 4 Experimental results on YTF data set

Feature	Distance	Accuracy
FPLBP	cosine	61.94 ± 1.23
FPLBP	Euclidean	62.32 ± 1.08
LBP	cosine	62.72 ± 1.41
LBP	Euclidean	62.91 ± 1.36
LBP	SML	65.88 ± 1.86
FPLBP	SML	68.04 ± 3.03
both feats collaboratively	CoSiM-m	69.92 ± 2.50
both feats collaboratively	CoSiM-a	$\mathbf{71.27} \pm 2.48$

since no additional data is used in either feature extraction or distance metric training.

The LFW data set includes 13,233 images that belong to 5749 different people, which can be viewed and downloaded at `http://vis-www.cs.umass.edu/lfw/`. One aspect that makes this data set difficult in machine learning tasks is that > 70% of individuals in the data set have only one image. A second difficulty is that the people in the training-test split are mutually exclusive. In other words, none of the people who are seen in the training set exist in the test set. This issue is referred to as the unseen pair matching problem. Sample image pairs from the data set can be seen in Fig. 6. It can be observed that two pictures of the same person can possess a considerable degree of variation whereas pictures of different people may even look more similar than those of same people.

4.1 Experiment set-up

For each of the ten subsets of the two data sets, we conducted individual SML experiments with separate features and observed the effect of applying CoSiM, i.e. collaborative learning. Table 3 shows the naming and the description of the systems proposed and experimented in this paper. For the YTF data set, we used LBP and FPLBP features and for the LFW data set we used LBP and SIFT features.

With the experiments on the systems shown in Table 3, we investigated the following:

(i) The discrimination power of the SML methodology, compared to other common metrics like Euclidean distance and the cosine distance
(ii) Effect of joint learning as an early fusion, by comparing performances of SML-S, SML-L and CoSiM-a/CoSiM-m.
(iii) CoSiM-a/CoSiM-m as an early fusion (as proposed in this paper) versus late fusion ($\oplus$) of individual soft decisions, by comparing CoSiM-a/CoSiM-m and SML-S $\oplus$SML-L
(iv) Late fusion of the proposed system with other high performance works in the literature, namely Fisher vector face (FVF) [41] and Sub-SML [10], for which the similarity scores and source code were publicly available.

Both data sets have the same face verification setup. YTF data set has 5000 pairs of images from videos, with equal number of friends and foes, while in LFW this number is 6000. To measure the robustness of the proposed algorithms, the data sets are divided into ten subsets each for ten-fold cross-validation experiments. The results reported here are the average accuracy and the standard deviation measured along ten different experiments. In each of the ten experiments, the models are trained with nine of the subsets and tested with the tenth subset. No experience is allowed to be transferred along different experiments, hence the models are initialised randomly from the beginning for each of the experiments. The features for the YTF data set are obtained from [14] and the features for the LFW data set are obtained from [10]. The results for YTF data set follows the image restricted paradigm, in which no additional data is utilised, whereas the data from [10] use the aligned version of the LFW data set (LFW-a) [42]. Therefore, the results for the LFW follow image restricted, label-free outside data paradigm. In this setting, the identity information of the face images are unknown and only the friend/foe information is known. Consequently, the results for LFW were unsurprisingly better than YTF. For each of the ten subsets, we have conducted individual SML experiments with the features and CoSiM experiments that use both features.

In addition to the DML performance of the proposed system, we also investigated the effect of late fusion of our systems, with two similar systems, FVF [41] and Sub-SML [10], for which the similarity scores and the source code were publicly available for LFW data set. In the late fusion process, we follow the procedures explained in [7, 10, 16]. Specifically, we concatenate the similarity scores of the fusion systems to train a linear SVM on the score vector to make predictions. Score calibration is needed before fusion, especially when combining with scores of other work. This procedure is conducted by the generalised version of the bnorm [43] introduced for score normalisation in the keyword search.

4.2 Numerical results

4.2.1 Experiments on YTF: We begin by evaluating the performance of verification using the two most widely used distance metrics, i.e. the cosine distance (17) and the Euclidean distance (30).

$$d_{\text{Euc}}(\boldsymbol{x}, \boldsymbol{y}) = \sqrt{(\boldsymbol{x}-\boldsymbol{y})^{\text{T}}(\boldsymbol{x}-\boldsymbol{y})} \tag{30}$$

We train the SML network with each of two features and compare the verification performance using the three distance/similarity metrics, i.e. cosine, Euclidean and SML. Table 4 shows the mean accuracy and the standard deviation over the ten-fold cross-validation. The performance of verification when the CoSiM similarity values are used as the discrimination criteria is also calculated. It can be observed that the SML provides about 3% absolute improvement in the validation accuracy. Furthermore, the validation performance is increased for another 2% absolute, when

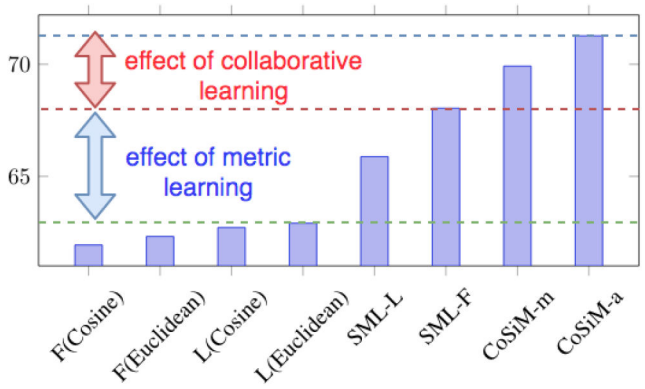


Fig. 7 *Effect of SML and CoSiM on validation accuracy, compared with other distance metrics on YFT data set*

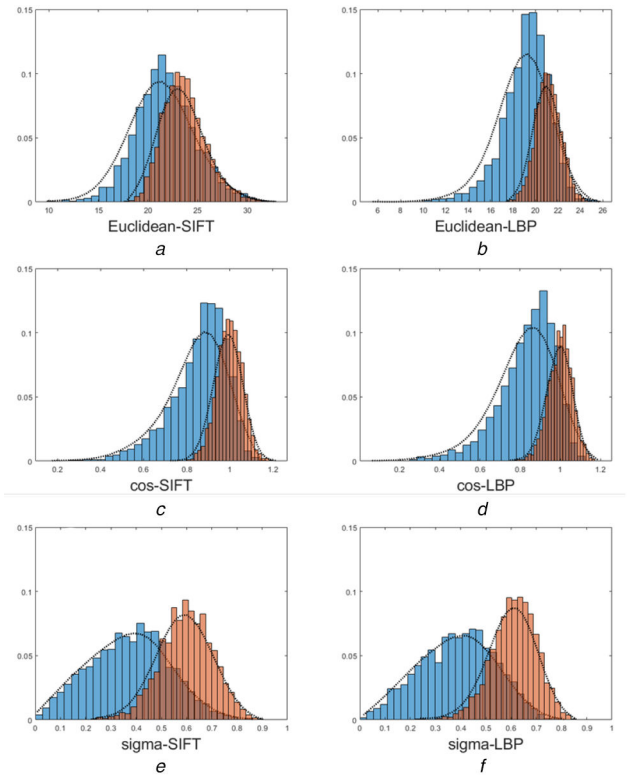


Fig. 8 *Normalised histograms of dispersion between friends (blue) and foes (orange) using different distance metrics and face representations. The drastic difference between (a,b,c,d) and (e,f) shows the effect of metric learning*

(a) Euclidean-SIFT, ***(b)*** Euclidean-LBP, ***(c)*** cos-SIFT, ***(d)*** cos-LBP, ***(e)*** sigma-SIFT, ***(f)*** sigma-LBP

the similarity is learned collaboratively using the two features. The effects of the proposed systems are visualised in Fig. 7.

4.2.2 Experiments on LFW: In this data set we also evaluate the performance of our system in comparison with a very similar method of LDML [7]. We see that both SML-S and SML-L outperform this baseline which use SIFT features as input and the similarity metric given in (16) where it was compared to our work. Furthermore, we see that both of the CoSiM methodologies yield better results than the late fusion of SML outputs (SML-S ⊕ SML-L). When we combine our work with similar performing methods of FVF [41] and Sub-SML [10], we observe a more significant increase in the performance due to the complementary nature of our work. Table 5 shows the mean accuracy and the standard deviation over the ten-fold cross validation.

4.2.3 Discriminative power of CoSiM: As another interrogation of the proposed approach we compared our similarity learning

Table 5 Experimental results on LFW view-2 data set

Method	Accuracy
LDML [7]	79.27 ± 0.60
SML-S	82.05 ± 1.1
SML-L	84.01 ± 1.30
SML-S ⊕ SML-L	86.66 ± 1.29
CoSiM-m	86.90 ± 1.06
CoSiM-a	87.01 ± 1.01
JSML	87.33 ± 1.21
FVF [41]	87.47 ± 1.49
sub-SML [10]	89.73 ± 0.38
JSML ⊕ Sub-SML	90.20 ± 1.12
JSML ⊕ FVF	90.72 ± 0.84
sub-SML ⊕ FVF	91.87 ± 1.14
JSML ⊕ FVF ⊕ Sub-SML	**92.64** ± 0.86

Table 6 Dispersion statistics of different distance metrics

Distance calculation		Dispersion between	
Representation	Metric	Friends	Foes
SIFT	Euclidean (30)	21.757	23.691
LBP	Euclidean (30)	19.180	21.163
SIFT	cosine (17)	0.859	0.999
LBP	cosine (17)	0.837	0.101
SIFT	sigma (31)	0.376	0.600
LBP	sigma (31)	0.389	0.612
CoSiM-m (32)		0.309	0.683
CoSiM-a (32)		0.299	0.690

approach with the two most widely used distance metrics, i.e. the cosine distance (17) and the Euclidean distance (30). The distance equivalent of the sigma similarity in (4), is obtained by calculating the complement since it can be interpreted as a probability measure [36],

$$d_\sigma = 1 - \sigma(\boldsymbol{x}^{\mathrm{T}}\boldsymbol{W}^{\mathrm{T}}\boldsymbol{W}\boldsymbol{y} + b)\,. \tag{31}$$

Similarly, the collaborative distance values that make use of two features are calculated with the CoSiM similarity values given in (4) and (8):

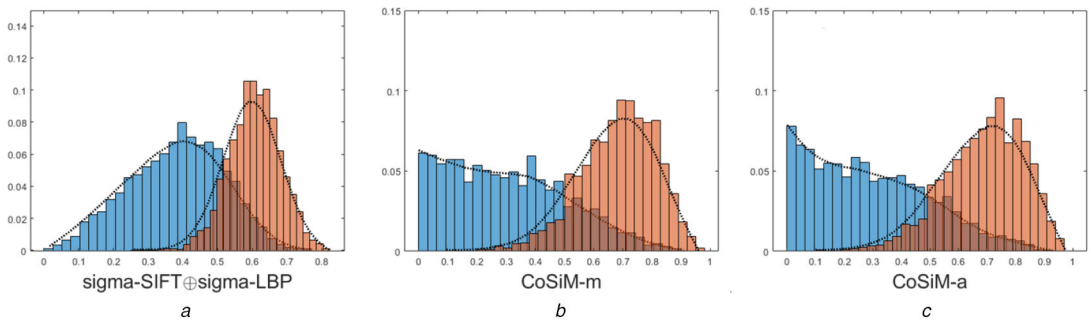
$$d_{\mathrm{CoSiM}} = 1 - f(\boldsymbol{x}_1, \boldsymbol{x}_2, \boldsymbol{y}_1, \boldsymbol{y}_2) \tag{32}$$

For this set of experiments, we calculated the distances between the pairs of friends and the pairs of foes in the test data sets. We calculated the histograms of distance values between 3000 friends and 3000 foes in the LFW data set, using different image representations and distance metrics. It should be noted that the test images are strictly excluded for pertinent training subset for the learning-based distance metrics. In other words, we train a new network from scratch for each of the subsets. Fig. 8 exhibits the histograms of friend and foe distances, calculated with Euclidean distance, cosine distance and the sigma distance.

The means of the distances between friends and foes (i.e. dispersion statistics) can be seen in Table 6. It can be seen both on the table and Fig. 8 that the Euclidean distance and the cosine distance perform poorly on discriminating friends and foes.

The mean distance values show that the learned distance metrics help discriminate *friends* and *foes* better than the existing metrics. Furthermore, the proposed collaborative distance metrics provide better discrimination than a single feature-based ones. The histogram plots of the distances calculated with the CoSiM methods and late fusion (⊕) of the sigma distance-based systems is given in Fig. 9. The comparison of Fig. 9*a* and *b* with *c* shows how collaborative learning is more efficient than learning and then fusing.

The comparison of the proposed performance to the works in literature is given in Table 7. The experiments on this table are



a *b* *c*

Fig. 9 *Normalised histograms of dispersion between friends (blue) and foes (orange) as a comparison of late versus early fusion techniques* ***(a)*** Sigma-SIFT⊕sigma-LBP, ***(b)*** CoSiM-m, ***(c)*** CoSiM-a

Table 7 Comparison with the literature

Method	Accuracy
MERL [2]	70.52 ± 0.60
MERL+Nowak [2]	76.18 ± 0.58
LDML [7]	79.27 ± 0.60
NReLU [8]	80.73 ± 1.34
Single LE + holistic [5]	81.22 ± 0.53
DML-eig SIFT [44]	81.27 ± 2.30
Hybrid, aligned [3]	83.98 ± 0.35
LARK supervised [3]	85.10 ± 0.59
LBP + CSML [19]	85.57 ± 0.52
Hybrid on LFW3D [45]	85.63 ± 0.53
DML-eig combined [44]	85.65 ± 0.56
combined b/g samples [4]	86.83 ± 0.34
TSML with OCLBP [9]	87.10 ± 0.43
FVF [41]	87.47 ± 1.49
pose adaptive filter [46]	87.77 ± 0.51
convolutional DBN [47]	87.77 ± 0.62
CSML + SVM [19]	88.00 ± 0.37
HT Brain-Inspired Feat.s [6]	88.13 ± 0.58
SFRD+PMML [48]	89.35 ± 0.50
LM3L [49]	89.57 ± 0.43
spartans [50]	89.69 ± 0.36
sub-SML [10]	89.73 ± 0.38
TSML with feature fusion [9]	89.80 ± 0.47
DDML [39]	90.68 ± 1.41
sub-SML + Hybrid on LFW3D [45]	91.65 ± 1.04
HPEN + HD-LBP + DDML [51]	92.57 ± 0.36
this work (JSML+FVF+Sub-SML)	$\mathbf{92.64 \pm 0.86}$
HPEN + HD-Gabor + DDML [51]	92.80 ± 0.47
MSBSIF-SIEDA [52]	94.63 ± 0.95

employed on exactly the same data set, with the same training and test set splits. The performance of the methodology presented in this work yields comparable results to the state of the art. It should be noted that the methods outperform our numbers generally use considerably deeper architectures or multiple feature representations. Since the YTF data set is originally designed for an unconstrained video verification task, a direct comparison with the literature is not feasible. The numbers reported in the literature for the YTF data set are obtained by a statistic of the frames of the videos to be compared. In our paper, we converted this task into face image verification by selecting a random frame from each video.

5 Discussion and conclusion

In this paper, a joint Siamese neural network training methodology is proposed for exploiting multiple face image features. The training methodology we refer to as CoSiM is an extension of the sigma DML network that was previously applied for speech features in a keyword search task. It was shown that the SML technique also works for image features such as SIFT and LBP and furthermore, the CoSiM methodology acts as an effective means of early fusion of several features. The experiments conducted on the YTF and LFW data set show that the results of the CoSiM training perform better than the late fusion of the individual distance metrics. The performance increase obtained by combining different approaches demonstrates the complementary nature of our approach. We see that SML-based methods enhanced with the joint training setup yield competitive results compared to other works in literature. The methods which outperform the proposed method generally use significantly deeper architectures or more sophisticated feature representations.

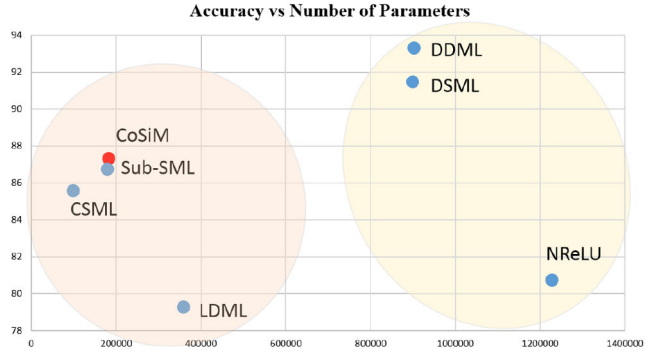


Fig. 10 *Accuracy versus the number of training parameters compared to similar works in the literature: Deep learning-based SML techniques tend to have higher parameters when compared to the cluster of projection matrix learning-based techniques, to which CoSiM belongs*

As stated throughout the paper, there are several approaches to the problem of face verification in the wild. The approaches that yield the highest scores use pre-trained CNNs with face frontalisation algorithms. They use, however, millions of additional face images to pre-train the CNNs and hence optimise for multiples of more training parameters. Our approach to the problem is comparable with the similarity metric and projection space learning-based approaches that mainly operate on Siamese nets. Our contribution to this aspect is the investigation of joint or (as we refer) collaborative learning of different sub-spaces and their fusion. The increase in accuracy obtained upon fusion with a system that assumes a methodology rooting for learning a compact discriminative space (FVF) shows that the two approaches learn aspects in discrimination that the other doesn't. Furthermore, when we fuse with an SML-based system that uses a similar methodology as ours and the same set of features as input, a further gain in the accuracy is observed. Hence, we conclude that the methodology proposed in this paper possesses a complementary nature to the other work used in fusion. A comparison of similar work with respect to the number of training parameters is exhibited in Fig. 10. It can be seen that the performance of the deep learning-based techniques is affected by the higher number of training parameters, and our approach is favourable in the cluster of SML-based techniques.

The experimental set-up for this paper was constrained to using two features per network. However, it should be noted that the methodology can very well be extended to multiple features and

the addition of more features can be quantified as a future study. Also, we used hand-crafted features in this paper, on the Siamese ends of the networks. The feed-forward nature of the network's two sides can be changed into CNN networks and the method can be made end-to-end.

6 Acknowledgments

Authors would like to thank the anonymous reviewers of this paper for their invaluable comments that made this work better.

7 References

[1] Learned-Miller, E., Huang, G.B., RoyChowdhury, A., *et al.*: 'Labeled faces in the wild: a survey', in Kawulok, M., Celebi, M.E., Smolka, B. (Eds.): '*Advances in face detection and facial image analysis*' (Springer, Poland, 2016), pp. 189–248

[2] Huang, G.B., Jones, M.J., Learned-Miller, E.: 'LFW results using a combined nowak plus merl recognizer'. Workshop on Faces in 'Real-Life' Images: Detection, Alignment, and Recognition, Marseille, France, 2008

[3] Taigman, Y., Wolf, L., Hassner, T., *et al.*: 'Multiple one-shots for utilizing class label information'. British Machine Vision Conf. (BMVC), London, England, vol. 2, 2009, pp. 1–12

[4] Wolf, L., Hassner, T., Taigman, Y.: 'Similarity scores based on background samples'. Asian Conf. on Computer Vision, Xi'an, China, 2009, pp. 88–97

[5] Cao, Z., Yin, Q., Tang, X., *et al.*: 'Face recognition with learning-based descriptor'. IEEE Conf. on Computer Vision and Pattern Recognition, San Francisco, CA, USA, 2010, pp. 2707–2714

[6] Cox, D., Pinto, N.: 'Beyond simple features: a large-scale feature search approach to unconstrained face recognition'. IEEE Int. Conf. on Automatic Face & Gesture Recognition and Workshops, Santa Barbara, CA, USA, 2011, pp. 8–15

[7] Guillaumin, M., Verbeek, J., Schmid, C.: 'Is that you? Metric learning approaches for face identification'. IEEE Int. Conf. on Computer Vision, Xi'an, China, 2009, pp. 498–505

[8] Nair, V., Hinton, G.E.: 'Rectified linear units improve restricted Boltzmann machines'. Int. Conf. on Machine Learning, Haifa, Israel, 2010, pp. 807–814

[9] Zheng, L., Idrissi, K., Garcia, C., *et al.*: 'Triangular similarity metric learning for face verification'. IEEE Int. Conf. and Workshops on Automatic Face and Gesture Recognition, Ljubljana, Slovenia, vol. 1, 2015, pp. 1–7

[10] Cao, Q., Ying, Y., Li, P.: 'Similarity metric learning for face recognition'. IEEE Int. Conf. on Computer Vision, Sydney, Australia, 2013, pp. 2408–2415

[11] Shen, W., Wang, B., Wang, Y., *et al.*: 'Face identification using reference-based features with message passing model', *Neurocomputing*, 2013, **99**, pp. 339–346

[12] Seo, H.J., Milanfar, P.: 'Face verification using the lark representation', *IEEE Trans. Inf. Forensics Sec.*, 2011, **6**, (4), pp. 1275–1286

[13] Lu, J., Hu, J., Tan, Y.P.: 'Discriminative deep metric learning for face and kinship verification', *IEEE Trans. Image Process.*, 2017, **26**, (9), pp. 4269–4282

[14] Wolf, L., Hassner, T., Maoz, I.: 'Face recognition in unconstrained videos with matched background similarity'. Proc. of the 2011 IEEE Conf. on Computer Vision and Pattern Recognition, Colorado Springs, CO, USA, 2011, pp. 529–534

[15] Huang, G.B., Ramesh, M., Berg, T., *et al.*: '*Labeled faces in the wild: a database for studying face recognition in unconstrained environments*' (University of Massachusetts, Amherst, 2007)

[16] Wolf, L., Hassner, T., Taigman, Y.: 'Descriptor based methods in the wild'. Workshop on Faces in 'Real-Life' Images: Detection, Alignment, and Recognition, Marseille, France, 2008

[17] Pinto, N., DiCarlo, J.J., Cox, D.D.: 'How far can you get with a modern face recognition test set using only simple features?'. IEEE Conf. on Computer Vision and Pattern Recognition, Miami, FL, USA, 2009, pp. 2591–2598

[18] Bar-Hillel, A., Hertz, T., Shental, N., *et al.*: 'Learning a Mahalanobis metric from equivalence constraints', *J. Mach. Learn. Res.*, 2005, **6**, pp. 937–965

[19] Nguyen, H.V., Bai, L.: 'Cosine similarity metric learning for face verification'. Asian Conf. on Computer Vision, Queenstown, New Zealand, 2010, pp. 709–720

[20] Cao, Q.: 'Some topics on similarity metric learning'. PhD thesis, University of Exeter, 2015

[21] Zheng, L., Idrissi, K., Garcia, C., *et al.*: 'Logistic similarity metric learning for face verification'. IEEE Int. Conf. on Acoustics, Speech and Signal Processing, Queensland, Australia, 2015

[22] Schroff, F., Kalenichenko, D., Philbin, J.: 'Facenet: a unified embedding for face recognition and clustering'. IEEE Conf. on Computer Vision and Pattern Recognition, Boston, MA, USA, 2015, pp. 815–823

[23] Snoek, C.G., Worring, M., Smeulders, A.W.: 'Early versus late fusion in semantic video analysis'. Proc. of the 13th Annual ACM Int. Conf. on Multimedia, Singapore, 2005, pp. 399–402

[24] Gunes, H., Piccardi, M.: 'Affect recognition from face and body: early fusion vs. late fusion'. 2005 IEEE Int. Conf. on Systems, Man and Cybernetics, Hawaii, HI, USA, 2005, vol. 4, pp. 3437–3443

[25] Dong, Y., Gao, S., Tao, K., *et al.*: 'Performance evaluation of early and late fusion methods for generic semantics indexing', *Pattern Anal. Appl.*, 2014, **17**, (1), pp. 37–50

[26] Lan, Z.z., Bao, L., Yu, S.I., *et al.*: 'Double fusion for multimedia event detection'. Int. Conf. on Multimedia Modeling, Klagenfurt, Austria, 2012, pp. 173–185

[27] Seeland, M., Rzanny, M., Alaqraa, N., *et al.*: 'Plant species classification using flower images–a comparative study of local feature representations', *PloS One*, 2017, **12**, (2), p. e0170629

[28] Hu, J., Lu, J., Tan, Y.-P.: 'Sharable and individual multi-view metric learning', *IEEE Trans. Pattern Anal. Mach. Intell.*, 2018, **40**, (9), pp. 2281–2288

[29] Gündoğdu, B., Yusuf, B., Saraçlar, M.: 'Joint learning of distance metric and query model for posteriorgram-based keyword search', *IEEE J. Sel. Top. Signal Process.*, 2017, **11**, (8), pp. 1318–1328

[30] Huang, G.B., Learned-Miller, E.: '*Labeled faces in the wild: updates and new reporting procedures*' (University of Massachusetts, Amherst, 2014), UM-CS-2014-003

[31] Chen, S., Ma, B., Zhang, K.: 'On the similarity metric and the distance metric', *Theor. Comput. Sci.*, 2009, **410**, (24-25), pp. 2365–2376

[32] Lowe, D.G.: 'Distinctive image features from scale-invariant keypoints', *Int. J. Comput. Vis.*, 2004, **60**, (2), pp. 91–110

[33] Luo, J., Ma, Y., Takikawa, E., *et al.*: 'Person-specific SIFT features for face recognition'. IEEE Int. Conf. on Acoustics, Speech and Signal Processing, Honolulu, HI, USA, 2007, vol. 2, pp. II–593

[34] Ahonen, T., Hadid, A., Pietikäinen, M.: 'Face recognition with local binary patterns'. European Conf. on Computer Vision, Prague, Czech Republic, 2004, pp. 469–481

[35] Ahonen, T., Hadid, A., Pietikainen, M.: 'Face description with local binary patterns: application to face recognition', *IEEE Trans. Pattern Anal. Mach. Intell.*, 2006, **28**, (12), pp. 2037–2041

[36] Gündoğdu, B., Saraçlar, M.: 'Distance metric learning for posteriorgram based keyword search'. IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP), New Orleans, LA, USA, 2017, pp. 5660–5664

[37] Chollet, F., *et al.*: 'Keras', 2015, Available at https://keras.io

[38] Kingma, D.P., Ba, J.: 'Adam: a method for stochastic optimization'. Int. Conf. on Learning Representations (ICLR), San Diego, CA, USA, 2015

[39] Hu, J., Lu, J., Tan, Y.P.: 'Discriminative deep metric learning for face verification in the wild'. IEEE Conf. on Computer Vision and Pattern Recognition, Ohio, OH, USA, 2014, pp. 1875–1882

[40] Mignon, A., Jurie, F.: 'PCCA: a new approach for distance learning from sparse pairwise constraints'. IEEE Conf. on Computer Vision and Pattern Recognition, Rhode Island, RI, USA, 2012, pp. 2666–2672

[41] Simonyan, K., Parkhi, O.M., Vedaldi, A., *et al.*: 'Fisher vector faces in the wild'. British Machine Vision Conf., Bristol, UK, 2013

[42] Wolf, L., Hassner, T., Taigman, Y.: 'Effective unconstrained face recognition by combining multiple descriptors and learned background statistics', *IEEE Trans. Pattern Anal. Mach. Intell.*, 2011, **33**, (10), pp. 1978–1990

[43] Gündogdu, B.: '*Keyword search for low resource languages*' (Bogaziçi University, Istanbul, Turkey, 2017)

[44] Ying, Y., Li, P.: 'Distance metric learning with eigenvalue optimization', *J. Mach. Learn. Res.*, 2012, **13**, pp. 1–26

[45] Hassner, T., Harel, S., Paz, E., *et al.*: 'Effective face frontalization in unconstrained images'. IEEE Conf. on Computer Vision and Pattern Recognition, Boston, MA, USA, 2015, pp. 4295–4304

[46] Yi, D., Lei, Z., Li, S.Z.: 'Towards pose robust face recognition'. IEEE Conf. on Computer Vision and Pattern Recognition, Portland, OR, USA, 2013, pp. 3539–3545

[47] Huang, G.B., Lee, H., Learned-Miller, E.: 'Learning hierarchical representations for face verification with convolutional deep belief networks'. IEEE Conf. on Computer Vision and Pattern Recognition, Rhode Island, RI, USA, 2012, pp. 2518–2525

[48] Cui, Z., Li, W., Xu, D., *et al.*: 'Fusing robust face region descriptors via multiple metric learning for face recognition in the wild'. IEEE Conf. on Computer Vision and Pattern Recognition, Rhode Island, RI, USA, 2013, pp. 3554–3561

[49] Hu, J., Lu, J., Yuan, J., *et al.*: 'Large margin multi-metric learning for face and kinship verification in the wild'. Asian Conf. on Computer Vision, Singapore, 2014, pp. 252–267

[50] Juefei-Xu, F., Luu, K., Savvides, M.: 'Spartans: single-sample periocular-based alignment-robust recognition technique applied to non-frontal scenarios', *IEEE Trans. Image Process.*, 2015, **24**, (12), pp. 4780–4795

[51] Zhu, X., Lei, Z., Yan, J., *et al.*: 'High-fidelity pose and expression normalization for face recognition in the wild'. IEEE Conf. on Computer Vision and Pattern Recognition, Boston, MA, USA, 2015, pp. 787–796

[52] Ouamane, A., Bengherabi, M., Hadid, A., *et al.*: 'Side-information based exponential discriminant analysis for face verification in the wild'. IEEE Int. Conf. and Workshops on Automatic Face and Gesture Recognition, Ljubljana, Slovenia, 2015, vol. 2, pp. 1–6