Face Verification based on Relational Disparity Features and Partial Least Squares Models

Rafael Vareto, Samira Silva, Filipe Costa, William Robson Schwartz
Smart Surveillance Interest Group, Department of Computer Science
Universidade Federal de Minas Gerais, Belo Horizonte, Brazil
{rafaelvareto, samirasilva, fcosta, william}@dcc.ufmg.br

***Abstract*—Face verification approaches aim at determining whether two given faces are from the same person. This scenario has several applications, such as information security, forensics, surveillance and smart cards. Several works extract features independently from each face image, i.e., any sort of relation between the two faces is not modeled *a priori* to either training or classification stages. In this work, we propose an approach that compares a pair of faces by extracting relational features, assuming the hypothesis that modeling the relation between two faces can be useful for increasing the robustness and performance of the face verification task. Then, we employ multiple classification models based on Partial Least Squares to verify whether a given pair of images belongs the same subject (genuine) or belongs to different subjects (impostor). We validate our approach on the Labeled Faces in the Wild (LFW) and on the Public Figures (Pubfig) datasets, using only few images for training. According to the experiments, our approach achieves results up to 0.966 of area under the curve (AUC) for the LFW dataset using its unrestricted, labeled outside data protocol and an average equal error (EER) of 13.65% on PubFig dataset.**

I. Introduction

Face recognition has been one of the most important tasks in computer vision during the last decades. Due to the wide range of applications in several environments (e.g., social medias, surveillance systems, access control) and the accessibility of feasible technology in the last years, face recognition tasks received significant attention from the scientific community. Furthermore, the approaches developed for face recognition still have some limitations caused by the conditions of real applications, such as partial occlusion, illumination variation, and camera resolution [1].

The face recognition problem can be divided into three main categories [2]: *face verification*, where the goal is to determine whether a pair of images corresponds to the same subject; *face identification*, when we assume that every queried subject was previously cataloged, ensuring that the probe face holds a corresponding identity in the gallery set; and *watch-list*, which is similar to face identification with the difference that it does not guarantee that all query subjects are registered in the face gallery (open-set task). Several researchers have developed approaches to improve the performance of automatic face recognition [3]–[8].

The face verification task can be described as a 1:1 matching problem. In this task, the main goal is to determine whether two given faces are from the same subject (i.e., genuine) or from different subjects (i.e., impostor). This scenario has several applications, including to check whether a person is the owner of an informed bank account or whether a specific person can access a restrict place. There are several works addressing face verification tasks [8]–[10]. Most of these works extract features independently from each face image. Therefore, any sort of relation between the two faces is not modeled *a priori* to either training or classification stages.

Assuming the hypothesis that modeling the relation between two faces can be useful for increasing the robustness and performance for face verification tasks, in this work we propose an approach that compares a pair of faces that extracts relational features by computing the absolute difference between their feature vectors. We believe that any pair of features of the same subject would present small differences. Consequently, this difference increases when we deal with images from different persons. In our method, we consider the difference of features as a new feature vector for a pair of different faces and categorize this new array into one of two classes: *same person* (genuine) and *different persons* (impostor). Such categorization is performed using multiple classification models based on Partial Least Squares (PLS) [11]–[13], each learned from a subset of the data.

According to experimental results, our approach reports competitive matching accuracy in comparison with other state-of-the-art works on two well-known datasets, Labeled Faces in the Wild (LFW) [14] and Public Figures (PubFig) [15]. We achieved up to 0.966 of the area under the curve (AUC) on the LFW dataset using the *unrestricted, labeled outside data* protocol and an average equal error (EER) of 13.66% on the PubFig dataset.

II. Related Works

Face recognition has been broadly studied in the past decade [5], [16]–[22]. For that reason, this paper is engaged in estimating the similarity between two face images despite their pose variations, illumination changes, age discrepancies, expression diversities, occlusions. We focus on the face verification task with unconstrained face images, that is, an environment whose images were taken having no standard expression, pose, or lighting condition.

Ouamane et al. [23] adopt a rich multi-scale facial texture representation to enhance performance. They propose a new dimensionality reduction technique that transforms the problem of face verification under weakly labeled data into a

generalized eigenvalue problem. Barkan et al. [24] build high-dimensional face representations using hand-crafted feature descriptors such as LBP and SIFT. Then, they employ different dimensionality reduction techniques in LFW's supervised and unsupervised cases [25]. In the final step, multiple representations and image features are combined together using uniform weighting of cosine similarities. Chen et al. [26] propose a two-step scheme to obtain sparse linear projections. They compress the original space into a low-dimensional feature so that a sparse matrix, which maps high-dimensional features into a low-dimensional representation, can be learned. Ouamane et al. [27] partition the image into many patches. Thus, features are extracted and summarized as histograms that are concatenated to form a high-dimensional feature vector. They reduce the dimensionality to increase their approach's performance. In general, methods comprised of high-dimensional spaces bring along several obstacles that may prevent further exploration, such as training, computation and storage issues.

Simonyan et al. [28] detect facial landmarks in favor of aligning and cropping face images before extracting compact feature descriptors derived from fisher vectors on densely sampled SIFT features. Ding et al. [29] design a new feature descriptor that computes the first derivative of Gaussian operator to lessen illumination effects before detecting feature patterns at both holistic and landmark levels. Landmark detection-based methods may attain higher performance at the cost of massive labeled training data, which seldom is available in practical applications.

The work of Hassner et al. [30] generates frontal face views of unconstrained photos. The authors approximate the shape of all input images using single 3D unmodified surfaces. First, they detect facial landmarks to render textured 3D models. Then, these 3D models are rotated to a desired pose and a new normalized 2D image is generated. Similarly, Zhu et al. [31] present a method that normalizes poses and expressions in pursuance of canonical-view face images. They also search for facial landmarks that are later used for meshing the entire image into a 3D object. Taigman et al. [32] come up with a facial alignment algorithm found on fiducial points detection and facial 3D modeling. They also introduce a deep neural architecture with nine layers to represent face images in a generalized manner. Three-dimensional models tend to work well, but depending on the subject's pose, information rendered from 3D techniques may end up hindering the recognition performance. Besides, if the faces contain occluded regions, these regions are generally mirrored, resulting in poor normalization results.

Köestinger et al. [33] present a method that learns distance metrics from constraints of equivalence, derived from inference perspective. They manage to escape optimization issues in order to revolve costly computational iterations. Hu et al. [34] present a deep metric learning method that aims to learn a Mahalanobis distance metric, maximize inter-class variations and minimize intra-class variations. A deep neural network learns hierarchical nonlinear transformations to fit a pair of face images into the same feature subspace so that discriminative information can be spotted. Zheng et al. [35] propose a linear cosine similarity metric learning method based on triangle inequalities and gradient functions. Cost and gradient functions are handled as a mathematical problem, which is solved with an optimization algorithm. Metric learning methods do not usually hold the nonlinear manifolds faces images lie on. Furthermore, nonlinear mapping functions are not explicitly acquired, causing scalability problems.

Sun et al. [9] introduced a hybrid convolutional network that learns relational visual features so that identity similarities can be pointed out. Their network compute local visual features from two face images that are processed through multiple layers for the sake of extracting high-level holistic features. The work of Schroff et al. [8] contemplates a deep convolutional neural network in an approach that projects face images into a compact Euclidean space in such a way that distances correspond to face similarity measures. Ding et al. [10] proposes a deep learning framework to represent faces using multi-modal information. The framework is made up of complementary convolutional neural networks that extracted features, which are concatenated with a three-layer stacked auto-encoder. Neural networks are usually hard to train and regularly require the tuning of numerous parameters. Depending on the problem, there are simpler and faster alternatives that may attain better performance, such as support vector machines and decision trees.

As detailed in this section, face verification methods first extract features from two query images separately. Some approaches compute low-level features [36]–[38] whereas others generate mid-level features [39], [40]. Notably, most face verification approaches extract features independently from each face image. Therefore, any sort of relation between the two faces are not modeled prior to the training or classification stages. Unlike the majority of works in the literature, we approach a pair of face images by extracting relational features as we compute the absolute difference between their feature vectors. We further detail the proposed approach in the next section.

III. Methodology

In this section, we describe our proposed face verification approach[1]. Figure 1 illustrates the designed face verification process, described in details in the next sections.

A. Partial Least Squares Model

Partial least squares (PLS) is a fast and effective regression technique based on covariance [11], [12]. It captures the relationship between observed variables through latent variables and associates aspects from principal component analysis and multiple regression. In addition, it works very well when the number of explanatory variables is both high and likely to be correlated and does not require a large number of training samples. The latter aspect is our main motivation

[1]https://github.com/rafaelvareto/HPLS-verification.

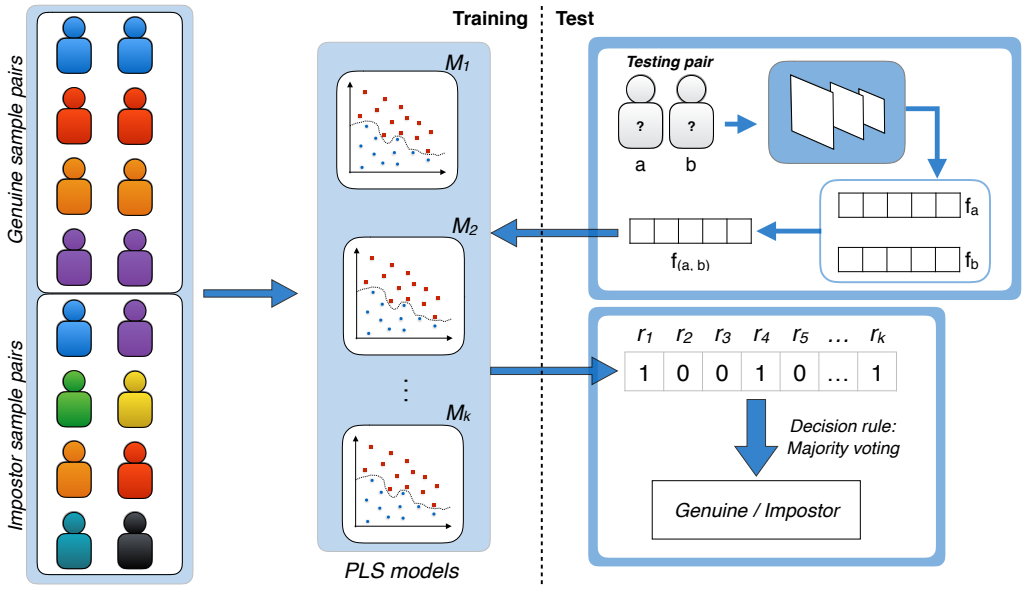

Fig. 1. Overview of the proposed face verification approach. *Training:* Disparity feature vectors are obtained for all pair of subjects before they are partitioned into genuine (same) and impostor (not same) sets. Then, different classification models are learned containing different feature samples in each subset. *Testing:* The disparity features are extracted from a pair of testing images to compose a feature vector which is then classified by all PLS models and their response values are used to estimate the label (genuine or impostor), based on a majority voting scheme.

for employing PLS in this work, since there are not many samples available for learning the models, preventing the employment of deep learning techniques in the process [41].

The goal of PLS [11] is to build latent variables as a linear combination of the predictor zero-mean variables X and Y. More precisely, X describes a matrix of feature descriptors whereas Y portrays a vector of response variables. Then, PLS seeks for latent vectors so they can be simultaneously decompounded into $X = TP^T + E$ and $Y = UQ^T + F$ in order to determine the maximum covariance between variables T and U. Matrix $T_{n\times p}$ characterizes latent variables from feature vectors and matrix $U_{n\times p}$ denotes latent variables from target values. Variables $P_{p\times d}$ and $Q_{1\times d}$ can be compared to the loading matrices from principal component analysis. Eventually, variables E and F represent residuals.

We employ the Non-linear Iterative PLS (NIPALS) [12] algorithm to estimate the low-dimensional data representation. NIPALS computes the highest covariance between latent variables T and U and produces a matrix of weight vectors $W_{d\times p}$. Then, it determines the regression coefficients vector β using least squares as follows: $\beta = W(P^TW)^{-1}T^TY$. The PLS regression output for query image's feature vector is given by $\hat{y} = \bar{y} + \beta^T(x - \bar{x})$ where $\bar{y}$ is the sample mean of Y and $\bar{x}$ the average values of X.

Each classification model encompasses a binary PLS regression model. In other words, every single PLS comprises a binary classifier: *genuine* and *impostor*. The *genuine* class holds the absolute difference of feature vectors for matching pairs of identities and the *impostor* class comprises "new generated" feature vectors that result from the combination of features of different subjects. We attribute the target value 1 to samples lying in the genuine class and values equal to 0 when samples belong to the impostor class.

The PLS models here described could be replaced by a variety of binary classifiers such as SVM (Support Vector Machines) [42] and a Fully Connected Networks (FCN). However, they might not be the best fit in this case because they would require a large number of samples in training stage, while PLS efficiently takes over scarce number of feature samples [17].

B. Feature Extraction

Different from considering a feature vector for each image independently, we extract relational features for pair of faces as follows. First, we extract deep features for all images employing VGGFace convolutional neural network descriptor. Then, we compute the absolute difference between them and use this new feature vector to build and execute the classifier.

Our main hypothesis lies on the fact that two face images of the same subject hold small differences. However, this difference increases when we cope with a pair of images from different subjects. Feature vectors that represent a pair of faces from the same person are labeled as *same person* (genuine) and feature vectors extracted from a pair of faces of different people are labeled as *not same persons* (impostor). Figure 2 illustrates this process. From now on, we refer to feature vectors derived from the absolute difference as *disparity feature vectors* or simply *disparity features*.

C. Training Stage

The training stage randomly samples disparity feature vectors that were previously distributed into two disjoint sets *same*, and *not same*, while the former relate to pairs of samples

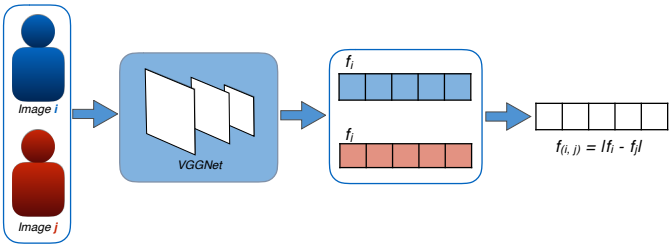

Fig. 2. Feature extraction overview for a pair of face images.

from the same subject, the latter refers to pairs of samples from different subjects. In pursuance of a balanced division, these disparity feature samples are drawn from a uniform distribution. The positive class contains only samples selected from the *same* collection and the negative class only contains samples selected from the *not same* collection. Then, a binary PLS model is learned considering the selected samples.

The generation of binary classifiers is repeated k times (the number of models is a parameter defined by the user) by selecting different disparity feature vectors from the classes *same* and *not same*, to capture different aspects of the data and allow the complementarity among the classifiers.

D. Testing Stage

In the testing stage, the method computes the disparity feature vector between the target and query samples and presents the feature to each of the k PLS classifiers, which results the response value r_i, where $i = 1, 2, \ldots, k$. In the end, it computes the majority voting to find which label must be designated to the probe, i.e., genuine or impostor. The former label is attributed when the pair of samples is classified as belonging to the same subject.

Rather than just outputting *same* or *not same* binary labels, the algorithm computes the ratio between the number of positive matches to the total number of PLS classification models. Therefore, we obtain a probability estimate of the positive class (target score), which is used to compute the Receiver Operating Characteristic (ROC) curves.

IV. Experimental Results

In this section, we evaluate our algorithm, which generates several binary Partial Least Squares (PLS) models coupled with majority voting to determine whether two faces belong to a same subject. We summarize the datasets in Section IV-A. Section IV-B details the feature descriptor employed. Evaluation metrics and protocols are detailed in Section IV-C. In Section IV-D we specify algorithm parameters. Finally, Section IV-E presents all results, including our comparison with methods available in the literature.

A. Datasets

For the sake of demonstrating the effectiveness of our approach, we chose two challenging datasets with different characteristics, ranging from pose variations and illumination changes to images with age discrepancies and expression diversities. We evaluate our method on the Labeled Faces in the Wild (LFW) [14], [25] and on the Public Figures (PubFig) [15].

1) Labeled Faces in the Wild (LFW): The LFW[2] [14], [25] dataset can be considered the genuine state-of-the-art benchmark for face verification. It also comprises face images aligned with an unsupervised deep feature algorithm, commonly known as LFW-A or deep-funneled LFW [43]. This dataset contains approximately $13,000$ uncontrolled face images of more than five thousand individuals. In contrast to the majority of existing face datasets, these images were taken in entirely unconstrained situations with non-cooperative individuals. Thus, there is also large divergence in pose, lighting, expression, scene, and camera. For fair comparison, the creators of LFW suggest reporting performance as a 10-fold cross validation using splits they have randomly generated. As other works, we used deep-funneled LFW face images (LFW-A).

2) Public Figures (PubFig): The PubFig[3] [15] dataset is larger than the LFW in terms of image samples, consisting of nearly $60,000$ images of 200 subjects gathered from across the Internet[4]. The database is considered as very difficult as it evidences vast variations in pose, lighting, facial expression, age, gender, and ethnicity. The PubFig dataset is divided into two units, the evaluation set with 140 subjects, designed to evaluate methods, and the development set with 60 individuals, which holds no overlap with the evaluation set.

B. Feature Descriptor

In this work, we employ the VGGFace CNN descriptor, computed using the implementation of Parkhi et al. [44] and based on the VGG-Very-Deep-16 CNN architecture [45], a descriptor that comprises a long sequence of convolutional layers. Furthermore, we do not employ any sort of fine tuning towards LFW or PubFig. Instead, we consider the network already learned using the standard training protocol proposed by Parkhi et al. [44], which considers a dataset with more than two million images and approximately $2,700$ identities.

C. Evaluation Protocol

The Receiver Operating Characteristic (ROC) curve reports true positive rate (on the y-axis) as a function of the false positive rate (on the x-axis). That indicates the top leftmost corner as the optimal point. Accurate face verification systems present true positive rates close to 1 even at very low false positive rates. A measure extracted from the ROC curve and commonly employed, is the Area Under Curve (AUC), which ranges from 0.5 to 1 (the closer to 1, the better).

The Equal Error Rate (EER) is another measure employed on face verification and biometrics in general. It indicates the value where the false rejection rate (i.e., fraction of genuine

[2]http://vis-www.cs.umass.edu/lfw/

[3]http://www.cs.columbia.edu/CAVE/databases/pubfig/

[4]The PubFig dataset was released long ago and they do not distribute image files due to copyright issues. Thus, only $26,787$ out of $58,797$ images remain available as links to these files are gradually disappearing over time.

samples classified as impostor) is equal to the false acceptance rate (i.e., fraction of impostor samples classifier as genuine). The lower the equal error rate, the higher the accuracy of the biometric system.

For the evaluation performed on the LFW dataset, we use the protocol *unrestricted, labeled outside data* for all experiments. We show the ROC curve, its AUC and the standard deviation error (STD) on the deep-funneled LFW. The *unrestricted* protocol allows researchers to exploit identities in the training set so that it is possible to generate more training pairs and add them to the training stage. For the PubFig, we present the equal error rate and the standard deviation.

Differently from many approaches that achieve state-of-the-art results following LFW's *unrestricted, labeled outside data* protocol, we do not focus on grouping millions of images in the interest of learning discriminative face representations using convolutional neural networks [8], [10], [32], [48]. Furthermore, many works [10], [32] make use of additional face datasets to train *same/not-same* classifiers as they claim that employing either LFW-A or PubFig to produce more training pairs substantially overfits the training data due to their redundant characteristics. On contrary, we carry out a minimal training, that is, we only work with pairs of images recommended by the dataset. The only outside data we use are the samples required in the learning process of the VGGFace CNN descriptor [44].

D. Experimental Setup

All experiments were performed on a Intel Xeon E5-2630 CPU with 2.30 GHz and 16GB of RAM using Ubuntu 14.04 LTS operating system, no more than 6 GB of RAM was required though. Our method has mainly three parameters: the number of PLS classification models (*hm*), the number of positive and negative disparity features per PLS model (*hs*) and the number of PLS dimensions in the latent space (d). We conducted experiments varying *hm* and *hs* from 100 to 500 in steps of 100. The best results were achieved with *hm* and *hs* equal to 300. Moreover, we also ranged d from 4 to 30 in a 2-step increase to conclude that it had little impact on our algorithm's performance. Therefore, we set d to 10.

E. Evaluation and Comparisons

The algorithm proposed in Section III is evaluated with LFW-A and PubFig datasets following the *unrestricted, labeled outside data* protocol. Our experiments are grouped in two categories: *same-dataset* evaluation and *cross-dataset* evaluation. In the former, we follow the LFW and PubFig splits strictly with no use of additional labeled training examples to increase the amount of data available when learning our PLS models, as we understand that outside datasets only for the purposes of extracting features is significantly different than using outside data to train classifiers. In the latter, we conduct experiments in a cross-dataset scenario. We trained our classifiers using PubFig development set and evaluated the performance on LFW splits for cross validation. We used

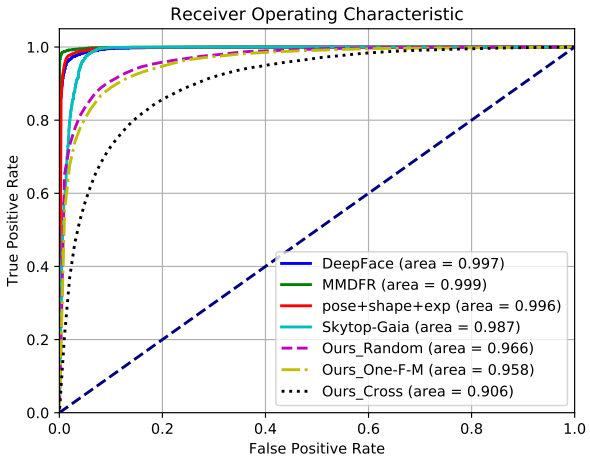

Fig. 3. Average ROC curves for the LFW-A dataset and its respective area under the curve (AUC). Some curves represent experiments conducted with deep funneled face images. We repeat our experiments ten times for each setting. The plot considers the following methods: DeepFace [32], MMDFR [10], Pose+Shape+Exp [48] and a commercial recognition system called SkyTop.

the PubFig development set because it is entirely disjoint of LFW identities and PubFig evaluation set individuals.

Table I shows the results on PubFig whereas Figure 3 shows the experiments on the LFW-A dataset. The cross-validation evaluation is adopted among the available folds, and we report the averaged results. Our approach was evaluated for different settings, described as follows.

- ***Cross:*** it categorizes a cross-dataset verification, in which the training stage uses images from PubFig development set and LFW folds are used during the testing stage. PubFig development set does not have a list of *same/not-same* training pairs. Therefore, training tuples were symmetrically sampled in a random manner.
- ***Dev-Eval:*** it is analogous to the cross-dataset experiment, but it comprises PubFig development set in the training stage and its evaluation set for testing. Thus, it does not constitute a cross-dataset experiment.
- ***One-F-M:*** the number of PLS models is associated with the number of folds in the cross validations scheme – one fold per PLS binary model. Particularly, in each iteration we pick a fold to test and each one of the remaining folds comprises a PLS model. Since the datasets have ten folds, we only generated nine PLS classification models.
- ***Random:*** it randomly allocates *same/not-same* training pairs into each PLS binary model as explained in Section III-C, ensuring that all training pairs are evenly distributed among all models.

According to our observations, running our algorithm ten times for every setting, except for *One-F-M* once it is deterministic, provides fair stability and small standard deviation error. According to the results showed in Table I and Figure 3, our method achieves comparable performance on both benchmarks making use of much less data during the

TABLE I
AVERAGE EQUAL ERROR RATE (EER) AND STANDARD DEVIATION (STD) FOR THE PUBFIG DATASET. TOP ROWS INDICATE APPROACHES WITH STATE-OF-THE-ART PERFORMANCE, OUR PERFORMANCE IS SHOWN IN MID ROWS AND BOTTOM ROWS PRESENT OTHER RELEVANT METHODS.

Approaches	EER (%)	STD
DRM-WV [49]	2.78	0.57
RNP [50]	10.79	0.83
OURS-Dev-Eval	13.65	2.11
OURS-Random	14.73	2.02
OURS-One-F-M	16.63	3.05
CHISD [51]	19.15	0.71
GEDA [52]	23.90	1.29

training stage and applying no preprocessing algorithm.

It is worth mentioning that even fixing the number of disparity feature samples in each PLS model to 100, the approach achieves good results. To check how our method responds to some parameter adjustments, we analyzed our approach behavior by varying the number of PLS classification models for both LFW-A and PubFig datasets under the *Random* and *Cross* setting. Table II shows how this parameter affects our method.

According to the results presented in Table II, there is a large improvement when the number of PLS classification models is increased from one to 100, indicating the need for multiple PLS models. However, there is no clear improvement when the number of models is increased from 100 to 500. The small AUC improvement for both datasets with increasingly classification models between 100 and 500 may be justified by the fact that algorithms trained with few-sample or few-subject gallery sets – LFW and PubFig, respectively – are inclined to remain invariable because most PLS models may be very similar to one another. Then, adding more PLS models only increases computational time.

The cross-dataset setting can also be analyzed according to Table II. We can see a slightly decrease since training data (i.e., PubFig development set) are not aligned and the testing dataset images (i.e., LFW-A), are aligned. Such alignment lessens undesired pose variations though actual systems cannot count on the cooperation of people being framed in order to assist the recognition process. However, the cross-dataset evaluation of PubFig development set and LFW folds demonstrates that our system can consistently achieve promising results while maintaining very good generalization ability.

Although the proposed method has not outperformed state-of-the-art methods, the experiments show that the proposed verification system attains favorable results. Furthermore, the approach remains stable even under different domains with limited number of training samples. Overall, this work confirms that there is no need of large amount of data in pursuance of quality results on the chosen benchmarks. With few thousands of face images, simple but robust algorithms can achieve very accurate results.

TABLE II
EVALUATION OF OUR METHOD'S PERFORMANCE (AUC) ON DIFFERENT DATASETS WITH AN INCREASINGLY NUMBER OF PLS CLASSIFICATION MODELS AND SUBJECT SAMPLES FIXED TO 100.

	Number of Models			
Approaches	1	100	300	500
OURS-Cross-LFW-A	0.620	0.886	0.897	0.899
OURS-Random-LFW-A	0.880	0.959	0.960	0.960
OURS-Dev-Eval-PubFig	0.801	0.941	0.942	0.942
OURS-Random-PubFig	0.810	0.936	0.937	0.938

V. CONCLUSIONS AND FUTURE DIRECTIONS

In this work, we designed and developed an approach to determine whether two face images belong to the same subject (face verification task). In addition, we also presented a feature extraction able to capture relational features between two samples.

Results have shown that our method achieves competitive results in comparison to state-of-the-art approaches even though being straightforward and simple. A literature comparison indicates an AUC of 0.966 using the LFW deep-funneled face images. We also performed a cross-dataset experiment to analyze how robust our algorithm can be when adapting its domain, achieving promising results. It is important to emphasize that most literature methods use additional face images from external datasets in spite of boosting their classification performance. Our algorithm belongs to a small group of approaches in the LFW benchmark that handles just pairs of images provided by the selected datasets (it uses outside images only for feature learning due to the use of the VGGFace CNN), yet it was able to achieve accurate results.

As future directions, we intend to perform experiments comprising the addition of massive datasets to the training stage for the sake of accomplishing better facial discrimination. Besides, since very few approaches learn a neural network from scratch, we intend to fine-tune some of the higher-level layers of the VGGFace CNN descriptor. We believe that these adjustments might boost our performance.

ACKNOWLEDGMENTS

The authors would like to thank the Brazilian National Research Council – CNPq, the Minas Gerais Research Foundation – FAPEMIG (Grants APQ-00567-14 and PPM-00540-17) and the Coordination for the Improvement of Higher Education Personnel – CAPES (DeepEyes Project). The authors gratefully acknowledge the support of NVIDIA Corporation with the donation of the GeForce Titan X GPU used for this research.

REFERENCES

[1] X. Zhang and Y. Gao, "Face Recognition Across Pose: A Review," vol. 42, no. 11, pp. 2876–2896, 2009.

[2] R. Chellappa, P. Sinha, and P. J. Phillips, "Face recognition by computers and humans," *Computer*, 2010.

[3] T. Ahonen, A. Hadid, and M. Pietikainen, "Face description with local binary patterns: Application to face recognition," *Pattern Analysis and Machine Intelligence*, 2006.

[4] B. F. Klare and A. K. Jain, "Heterogeneous face recognition using kernel prototype similarities," *Pattern Analysis and Machine Intelligence*, 2013.

[5] J. Lu, Y.-P. Tan, and G. Wang, "Discriminative multimanifold analysis for face recognition from a single training sample per person," *Pattern Analysis and Machine Intelligence*, 2013.

[6] D. Yi, Z. Lei, and S. Z. Li, "Towards pose robust face recognition," in *Computer Vision and Pattern Recognition*. IEEE, 2013.

[7] Y. Lei, M. Bennamoun, M. Hayat, and Y. Guo, "An efficient 3d face recognition approach using local geometrical signatures," *Pattern Recognition*, 2014.

[8] F. Schroff, D. Kalenichenko, and J. Philbin, "Facenet: A unified embedding for face recognition and clustering," in *Conference on Computer Vision and Pattern Recognition*. IEEE, 2015.

[9] Y. Sun, X. Wang, and X. Tang, "Hybrid deep learning for face verification," in *ICCV*. IEEE, 2013.

[10] C. Ding and D. Tao, "Robust face recognition via multimodal deep face representation," *Transactions on Multimedia*, 2015.

[11] H. Wold, "Partial least squares," *Encyclopedia of statistical sciences*, 1985.

[12] R. Rosipal and N. Krämer, "Overview and recent advances in partial least squares," in *Subspace, latent structure and feature selection*. Springer, 2006.

[13] C. E. dos Santos Jr., E. Kijak, G. Gravier, and W. R. Schwartz, "Partial least squares for face hashing," *Neurocomputing*, vol. 213, pp. 34–47, 2016.

[14] G. B. Huang, M. Ramesh, T. Berg, and E. Learned-Miller, "Labeled faces in the wild: A database for studying face recognition in unconstrained environments," Dept. Computer Science, University of Massachusetts – Amherst, USA, Tech. Rep., 2007.

[15] N. Kumar, A. C. Berg, P. N. Belhumeur, and S. K. Nayar, "Attribute and simile classifiers for face verification," in *ICCV*, 2009.

[16] J. Wright, A. Y. Yang, A. Ganesh, S. S. Sastry, and Y. Ma, "Robust face recognition via sparse representation," *TPAMI*, 2009.

[17] H. Guo, W. R. Schwartz, and L. S. Davis, "Face verification using large feature sets and one shot similarity," in *International Joint Conference on Biometrics*. IEEE, 2011.

[18] W. R. Schwartz, H. Guo, J. Choi, and L. S. Davis, "Face Identification Using Large Feature Sets," *IEEE Transactions on Image Processing*, vol. 21, no. 4, pp. 2245–2255, 2012.

[19] A. Wagner, J. Wright, A. Ganesh, Z. Zhou, H. Mobahi, and Y. Ma, "Toward a practical face recognition system: Robust alignment and illumination by sparse representation," *TPAMI*, 2012.

[20] L. Best-Rowden, H. Han, C. Otto, B. F. Klare, and A. K. Jain, "Unconstrained face recognition: Identifying a person of interest from a media collection," *Information Forensics and Security*, 2014.

[21] S. Hu, J. Choi, A. L. Chan, and W. R. Schwartz, "Thermal-to-visible Face Recognition using Partial Least Squares," *Journal of the Optical Society of America A*, vol. 32, no. 3, pp. 431–442, Mar 2015.

[22] B. F. Klare, B. Klein, E. Taborsky, A. Blanton, J. Cheney, K. Allen, P. Grother, A. Mah, and A. K. Jain, "Pushing the frontiers of unconstrained face detection and recognition: Iarpa janus benchmark a," in *Computer Vision and Pattern Recognition*. IEEE, 2015.

[23] A. Ouamane, M. Bengherabi, A. Hadid, and M. Cheriet, "Side-information based exponential discriminant analysis for face verification in the wild," in *Automatic Face and Gesture Recognition Workshop*. IEEE, 2015.

[24] O. Barkan, J. Weill, L. Wolf, and H. Aronowitz, "Fast high dimensional vector multiplication face recognition," in *ICCV*, 2013.

[25] G. B. Huang and E. Learned-Miller, "Labeled faces in the wild: Updates and new reporting procedures," Dept. Computer Science, University of Massachusetts – Amherst, USA, Tech. Rep., 2014.

[26] D. Chen, X. Cao, F. Wen, and J. Sun, "Blessing of dimensionality: High-dimensional feature and its efficient compression for face verification," in *Computer Vision and Pattern Recognition*. IEEE, 2013.

[27] A. Ouamane, B. Messaoud, A. Guessoum, A. Hadid, and M. Cheriet, "Multi scale multi descriptor local binary features and exponential discriminant analysis for robust face authentication," in *International Conference on Image Processing*. IEEE, 2014.

[28] K. Simonyan, O. M. Parkhi, A. Vedaldi, and A. Zisserman, "Fisher vector faces in the wild." in *British Machine Vision Conference*, 2013.

[29] C. Ding, J. Choi, D. Tao, and L. S. Davis, "Multi-directional multi-level dual-cross patterns for robust face recognition," *Pattern Analysis and Machine Intelligence*, 2016.

[30] T. Hassner, S. Harel, E. Paz, and R. Enbar, "Effective face frontalization in unconstrained images," in *CVPR*, 2015.

[31] X. Zhu, Z. Lei, J. Yan, D. Yi, and S. Z. Li, "High-fidelity pose and expression normalization for face recognition in the wild," in *Computer Vision and Pattern Recognition*. IEEE, 2015.

[32] Y. Taigman, M. Yang, M. Ranzato, and L. Wolf, "Deepface: Closing the gap to human-level performance in face verification," in *Computer Vision and Pattern Recognition*. IEEE, 2014.

[33] M. Köestinger, M. Hirzer, P. Wohlhart, P. M. Roth, and H. Bischof, "Large scale metric learning from equivalence constraints," in *Computer Vision and Pattern Recognition*. IEEE, 2012.

[34] J. Hu, J. Lu, and Y.-P. Tan, "Discriminative deep metric learning for face verification in the wild," in *CVPR*, 2014.

[35] L. Zheng, K. Idrissi, C. Garcia, S. Duffner, and A. Baskurt, "Triangular similarity metric learning for face verification," in *Automatic Face and Gesture Recognition Workshop*. IEEE, 2015.

[36] T. Ojala, M. Pietikainen, and T. Maenpaa, "Multiresolution gray-scale and rotation invariant texture classification with local binary patterns," *Pattern Analysis and Machine Intelligence*, 2002.

[37] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *International Journal of Computer Vision*, 2004.

[38] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *Computer Vision and Pattern Recognition*. IEEE, 2005.

[39] H. Lee, R. Grosse, R. Ranganath, and A. Y. Ng, "Convolutional deep belief networks for scalable unsupervised learning of hierarchical representations," in *International Conference on Machine Learning*. ACM, 2009.

[40] G. B. Huang, H. Lee, and E. Learned-Miller, "Learning hierarchical representations for face verification with convolutional deep belief networks," in *Computer Vision and Pattern Recognition*. IEEE, 2012.

[41] Y. Bengio, A. Courville, and P. Vincent, "Representation learning: A review and new perspectives," *TPAMI*, vol. 35, no. 8, pp. 1798–1828, Aug. 2013.

[42] M. A. Hearst, S. T. Dumais, E. Osuna, J. Platt, and B. Scholkopf, "Support vector machines," *Intelligent Systems and their Applications*, 1998.

[43] G. Huang, M. Mattar, H. Lee, and E. G. Learned-Miller, "Learning to align from scratch," in *Advances in Neural Information Processing Systems*, 2012, pp. 764–772.

[44] O. M. Parkhi, A. Vedaldi, and A. Zisserman, "Deep face recognition," in *British Machine Vision Conference*. IEEE, 2015.

[45] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *International Conference on Learning Representations*, 2015.

[46] D. G. Lowe, "Object recognition from local scale-invariant features," in *Internation Conference on Computer Vision*. IEEE, 1999.

[47] H. Bay, A. Ess, T. Tuytelaars, and L. Van Gool, "Speeded-up robust features," *Computer Vision and Image Understanding*, 2008.

[48] I. Masi, A. T. Trãn, T. Hassner, J. T. Leksut, and G. Medioni, "Do we really need to collect millions of faces for effective face recognition?" in *European Conference on Computer Vision*. Springer, 2016.

[49] M. Hayat, M. Bennamoun, and S. An, "Deep reconstruction models for image set classification," *TPAMI*, 2015.

[50] M. Yang, P. Zhu, L. Van Gool, and L. Zhang, "Face recognition based on regularized nearest points between image sets," in *Automatic Face and Gesture Recognition Workshop*. IEEE, 2013.

[51] H. Cevikalp and B. Triggs, "Face recognition based on image sets," in *Computer Vision and Pattern Recognition*. IEEE, 2010.

[52] M. T. Harandi, C. Sanderson, S. Shirazi, and B. C. Lovell, "Graph embedding discriminant analysis on grassmannian manifolds for improved image set matching," in *Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2011.