LC-DECAL: Label Consistent Deep Collaborative Learning for Face Recognition

Lamha Goel, Mayank Vatsa, and Richa Singh
IIIT-Delhi, India
{lamha15050, mayank, rsingh}@iiitd.ac.in

Abstract

With the advent of deep learning architectures, the performance of face recognition has witnessed significant improvements. However, this has also necessitated the requirement of large labeled training database. While approaches exist to utilize labeled or unlabeled data from related domains, in this paper, we present a collaborative learning framework that utilizes the availability of both labeled and unlabeled data along with the presence of multiple experts, to improve the performance of face analysis related tasks. The proposed Label Consistent Deep Collaborative Learning (LC-DECAL) framework makes use of label consistency, transfer learning, ensemble learning, and co-training for training a deep neural network for the target domain. The efficacy of the proposed algorithm is demonstrated with two existing Convolutional Neural Network architectures, DenseNet and ResNet, via experiments on multiple face databases, namely YTF, PaSC Handheld, PaSC Control, CelebA, and LFW-a. Experimental results show that the proposed framework yields comparable results to state-of-the-art results on all the databases.

1. Introduction

The performance of face recognition has seen impressive improvements since the advent of deep learning algorithms. The availability of large scale face databases such as Celeb-A [21] and VGGFace [9, 26] have facilitated the deep architectures to provide state-of-the-art results [8, 26]. In order to further improve the performance, we require either improved architectures or larger databases with increased variability or both. While collecting larger labeled databases is time-consuming and expensive, unlabeled data is usually available with ease. However, using both labeled and unlabeled data for training the deep architectures require dedicated mechanism such as transfer learning or co-training. In this paper, we introduce a "collaborative learning" framework, built upon transfer learning and co-training approaches, which utilizes labeled and unlabeled data for training a deep network for face recognition.

978-1-7281-1522-1/19/$31.00 ©2019 IEEE

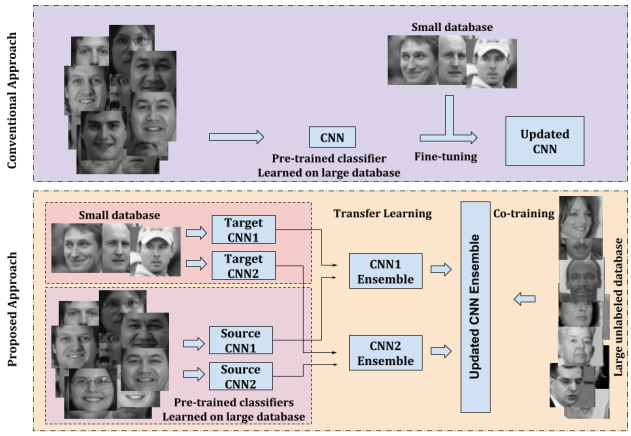


Figure 1: LC-DECAL integrates co-training and transfer learning to obtain a better model than traditional fine-tuning.

Figure 1 summarizes the main concept of the proposed framework. Transfer learning [25] is one of the most popular techniques for domain adaptation. It utilizes the knowledge learned in the source domain to improve the performance in the target domain. Fine-tuning is a common example of transfer learning. On the other hand, co-training [5] takes two views of the data, builds a classifier for each of these views, and pseudo-labels the unlabeled data to train the classifiers. Integrating transfer learning and co-training in this research, the focus is on utilizing the knowledge of multiple experts (models performing well for some task similar to the target task) and propose a framework for collaborative learning of deep experts. This research makes two-fold contributions: (i) collaborative learning framework with two stages: co-training and transfer learning which enables collaborative learning of two models with unlabeled data as well, and (ii) incorporating label consis-

tency to learn discriminative features. The proposed approach, termed as *LC-DECAL*, uses labeled and unlabeled data from target domain along with labeled data from the source domain to improve over traditional fine-tuning (Figure 1). The effectiveness of the proposed algorithm is evaluated on face recognition and face attribute classification databases, and the comparison is performed with state-of-the-art algorithms individually on all the databases.

1.1. Literature Review

A lot of research is being done to utilize multiple experts or unlabeled data to improve the performance for face analysis tasks such as face recognition and attribute prediction. Gao *et al*. [11] proposed Semi-Supervised Sparse Representation based Classification to perform face recognition with few labeled samples and possibly corrupted by nuisance variables such as bad lighting or expression changes. Bhatt *et al*. [2, 3] used co-training to improve face identification for cross-resolution images. Yu *et al*. [35] trained the model for current domain with small database by transferring hierarchical representations of an already learned deep face model. Singh *et al*. [28] proposed supervised COSMOS autoencoder for classification tasks. Majumdar *et al*. [23] proposed class sparsity based supervised encoding for face verification. El Gayar *et al*. [10] compared the performance of several semi-supervised multiple classifier systems for the task of face recognition. Cherniavsky *et al*. [7] explored a semi-supervised approach to learn human facial attributes from video. They also showed that training on video data improved performance as compared to training on image data.

Multiple approaches have been proposed to use deep learning with small datasets. Ma *et al*. [22] introduced self-paced co-training which allows updating the falsely labeled instances. Few-shot learning [8] is being explored to learn classifiers from datasets with very few samples. A Low-Shot Transfer Detector has been proposed for target detection task with very few training examples [6]. Zhou and Goldman [37] introduced democratic co-learning which uses multiple algorithms instead of multiple views. A constructive algorithm for cooperative neural network ensembles which determines ensemble architectures using incremental training and negative correlation learning has also been proposed [20].

Researchers have also focused on the area of collaborative learning for various applications. Blum *et al*. [4] proposed collaborative PAC learning, where multiple players learn the same concept with the goal to get an accurate classifier for all players simultaneously. Wang *et al*. [31] proposed a two-stage Deep Collaborative Learning module which divides a convolutional layer in two steps: smaller convolutions and fusion of their outputs. Vanhaesebrouck *et al*. [29] proposed a collaborative learning framework for multiple agents where the updates to an agents model were governed by both: how the local data is and how it's neighbors behave. Wang *et al*. [30] proposed a Deep Asymmetric Transfer Network for unbalanced domain adaptation. This model accounts for the fact that usually source domain has more reliable knowledge than the target domain.

Name is inspired by the definition of decal (as per Google): a design prepared on special paper for durable transfer on to another surface such as glass and porcelain

2. Proposed: Label Consistent Deep Collaborative Learning (LC-DECAL)

In this paper, a Label Consistent Deep Collaborative Learning model has been presented which utilizes co-training to benefit from unlabeled data samples, transfer learning to benefit from source domain's labeled data, ensemble learning to combine classifiers, and label consistency to learn discriminative features.

2.1. Label Consistent Learning

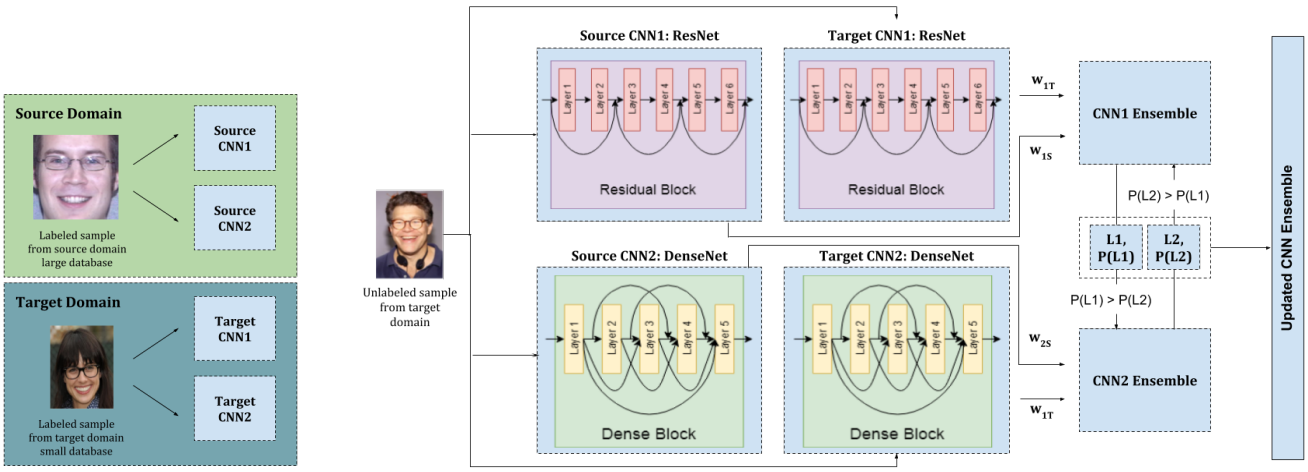
In a convolutional neural network, the initial layers learn general features whereas the last few layers learn more task-specific features [34]. The general cost function for a classification problem is the cross entropy loss for class labels obtained by applying softmax on the last layer of the network. Label consistency helps improve the features learned by the last few hidden layers by considering the error in the predictions made by each of these layers. For each of these layers, the output of the layer is passed through a fully connected layer. For the cost function, the ℓ_2 norm of the difference in the output of this fully-connected layer and the one-hot encoding of the required labels is considered. The final label consistent cost function is a weighted sum of these (Equation 1):

$$\begin{aligned} Cost &= Cost_{CE} + \alpha Cost_{LC} \\ &= \sum_i y_i log_2 y'_i + \alpha \sum_{j=1}^{k} jjy \quad \phi(M_j H_j) jj_2 \end{aligned} \tag{1}$$

where y represents one-hot encoding of the true labels, y' represents the probabilities obtained for each class, H_j is the representation of the j^{th} last hidden layer, M_j is the weights matrix of the fully connected layer attached to layer H_j, ϕ is the activation function, and k is the number of hidden layers being included for label consistency.

2.2. Transfer Learning via Ensembles

Transfer learning is beneficial when there is a similar domain (referred to as the source domain) with availability of a large amount of labeled data. A model trained only on the source domain data does not directly perform well



(a) Pre-training: Source domain and target domain data used for training two classifiers in their respective domain.

(b) LC-DECAL framework: w_{jS} and w_{jT} denote the weight of the source and target domain classifiers respectively for the j^{th} view, Lj and P(Lj) denote the prediction of the CNN_j ensemble and its confidence in the prediction. ResNet [17] is chosen as CNN1 and DenseNet [18] as CNN2 for both the domains.

Figure 2: Proposed Label Consistent Deep Collaborative Learning (LC-DECAL)

in the target domain because of the differences in the data distribution. With convolutional neural networks, conventional transfer learning takes a trained classifier of source domain and fine-tunes it using the target domain data. Instead, LC-DECAL uses transfer learning to leverage models trained in each domain by combining them in an ensemble fashion. We take two classifiers - (i) Source domain classifier, trained on source domain labeled data and (ii) Target domain classifier, trained on limited target domain labeled data. The ensemble is built as a weighted combination of these classifiers. Initially, both classifiers are equally weighted: $w_s = 0.5$ and $w_t = 0.5$. For improving the models, pseudo-labeled data is used (obtained by co-training). The classifiers are fine-tuned by back-propagation and the weights are updated as suggested by [36]:

$$h^i(C_X^i) = exp(\quad 0.5 \quad jjC_x^i \quad y^i jj_2^2) \tag{2}$$

$$w_S^{i+1,j} = w_S^{i,j} \quad h^i(C_S^i)/(w_S^{i,j} \quad h^i(C_S^i) + w_T^{i,j} \quad h^i(C_T^i)) \tag{3}$$

$$w_T^{i+1,j} = w_T^{i,j} \quad h^i(C_T^i)/(w_S^{i,j} \quad h^i(C_S^i) + w_T^{i,j} \quad h^i(C_T^i)) \tag{4}$$

where, $w_X^{i,j}$ denotes the value of weight for classifier X (source or target, i.e., $X \epsilon fS, Tg$) in the i^{th} iteration for the j^{th} view, C_X^i is the vector of probabilities for each class given by classifier X, and y^i is the one-hot encoding of pseudo label given by co-training.

2.3. Co-training

Co-training is applicable for classification problems where at least two independent views of the data are available along with large amount of unlabeled data and some labeled data to train the models. Co-training is used to provide pseudo-labels to the unlabeled data. Given two independent classifiers (referred to as the 2 views on the data), each trained to have better than random accuracy, the pseudo-label for a given unlabeled data sample is obtained in the following way: for each classifier E^i, its prediction L^i is obtained along with the prediction confidence p^i. If E_1 is more confident, the pseudo-label is L^1 and E_2 is trained on this sample using transfer learning. Similarly, if E_2 is more confident, the sample is given the pseudo-label L_2 and E_1 is trained on this sample using transfer learning. If both classifiers are equally confident, the process is repeated with the sample later after the classifiers have been further trained on more samples (Algorithm 1 presents the pseudo-code).

2.4. Label Consistent Deep Collaborative Learning

The LC-DECAL approach uses co-training as described to obtain pseudo-labels for unlabeled data from models for 2 views (the 2 classifiers), uses transfer learning to update the ensemble of source and target domain models in each view, and label consistency to ensure these models learn discriminative features. The complete algorithm is summarized in Figure 2 and Algorithm 2.

L^i = argmax$_l$ $P(l|E^i, X)$
p^i = $P(L^i|E^i, X)$
if $p^2 > p^1$ **then**
  Train E^1 on (X, L^2);
else if $p^1 > p^2$ **then**
  Train E^2 on (X, L^1);
else
  Process X again after the current mini-batch
end

Algorithm 1: Co-training component of LC-DECAL

Input : Trained classifiers: CNN^1_S and CNN^2_S on source domain data and CNN^1_T and CNN^2_T on the target domain labeled data
Symbols : E^1 = Weighted ensemble of CNN^1_S and CNN^1_T, E^2 = Weighted ensemble of CNN^2_S and CNN^2_T
Initialize: $w^1_S = 0.5$, $w^1_T = 0.5$, $w^2_S = 0.5$, $w^2_T = 0.5$
for *each unlabeled sample X in target domain* **do**
  L^1 = `PredictedLabel`(E^1, X);
  p^1 = **P** $(L^1|E^1, X)$;
  L^2 = `PredictedLabel`(E^2, X);
  p^2 = **P** $(L^2|E^2, X)$;
  if $p^2 > p^1$ **then**
    Train E^1 on (X, L^2) with cost function as in Eq. 1;
    $w^1_S = w^1_S * h(CNN^1_S)/(w^1_S * h(CNN^1_S) + w^1_T * h(CNN^1_T))$;
    $w^1_T = 1 - w^1_S$;
  else if $p^1 > p^2$ **then**
    Train E^2 on (X, L^1) with cost function as in Eq. 1;
    $w^2_S = w^2_S * h(CNN^2_S)/(w^2_S * h(CNN^2_S) + w^2_T * h(CNN^2_T))$;
    $w^2_T = 1 - w^2_S$;
  else
    Process X again after the current mini-batch
  end
end

Algorithm 2: Label Consistent Deep Collaborative Learning

3. Implementation Details

Given unlabeled samples of data from target domain, LC-DECAL uses co-training to obtain pseudo-labels. In this research, DenseNet [18] and ResNet [17] are used as the two views for co-training. For each sample, the more confident classifier provides the pseudo-labels while the less confident classifier is trained on the sample with the given pseudo-label using transfer learning. As the classifiers are trained with unlabeled data, the weights are updated as given in Equations 3 and 4. The network is trained using mini-batch gradient descent.

Each 3 3 convolutional layer of DenseNet has a 1 1 convolutional layer preceding it as a bottleneck layer. The (1 1) convolution + (3 3) convolution is regarded as one layer. Each convolution is preceded by batch normalization [19] and ReLU layers. We did not use any dropout or data augmentation. Further, CNN1 (DenseNet) and CNN2 (ResNet) ensembles are used to jointly provide the label using sum of prediction probabilities. We used Theano to implement the system prototype and cuDNN library on K80 GPU to accelerate training and testing on the models.

4. Experimental Analysis

4.1. Databases

The proposed algorithm is evaluated on the tasks of face recognition and attribute prediction. Figure 3 shows some sample images from the face databases.

CMU Multi-PIE [14] face database has more than 750,000 high-resolution images of 337 people with variations in expressions, illumination conditions, and view points. A subset of Multi-PIE containing 50,248 images of 337 people corresponding to expressions and illumination variations. has been used as the source domain database for face recognition problems in this research.

Surveillance Cameras Face Database (SCface) [13] provides 4160 images of 130 subjects captured in an uncontrolled indoor environment via 5 surveillance cameras.

YouTube Faces (YTF) [32] database has 3425 videos of 1595 people obtained from YouTube with an average of 181.3 frames per video.

Point and Shoot Challenge (PaSC) [1] database is split into PaSC control and PaSC handheld databases. There are 1401 videos in each of these databases of 265 subjects captured at 6 locations. The results are shown on both PaSC control and PaSC handheld databases.

Celeb Faces Attributes (CelebA) [21] database contains 202,599 celebrity images, each with 40 attribute annotations and 5 landmark locations, pertaining to 10,177 identities. The images cover various poses and background clutter.

Labeled Faces in the Wild-a (LFW-a) [33] database contains 13,143 gray-scale images, aligned using a commercial face alignment software. For 1680 subjects, there are at least 2 images.

4.2. Experimental Protocol

Face Recognition: The proposed approach has been evaluated on YouTube Faces (YTF) [32] and the Point and Shoot Challenge (PaSC) [1] databases. Both these databases have predefined experimental protocols. The YouTube

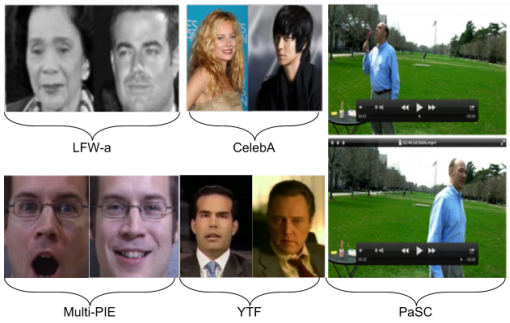


Figure 3: Sample images from some of the face databases used in this research.

Faces database provides 10 splits of data, each consisting of 250 genuine and 250 impostor pairs. The mean verification accuracy at 1% False Accept Rate (FAR) of 10 fold cross validation has been reported. The PaSC handheld database evaluates an algorithm for matching at low-resolution whereas PaSC control database evaluates the algorithm for high-resolution matching. For training, a separate set of training data provided with the PaSC database is used, and for testing, the pairs are generated from the samples given in the main database. The training data of both databases was divided into two parts: first was used to pre-train the target domain models, and the second part was used as unlabeled data to update the source and target models using the proposed LC-DECAL. CMU Multi-PIE and SCface databases were used to pre-train the source models for face recognition experiments.

Attribute Classification: For attribute classification problem, the proposed algorithm has been evaluated on CelebA and LFW-a databases. Standard protocols were followed for both the databases. The CelebA database has been divided into three parts: about 160,000 images for training, and about 20,000 images each for validation and testing sets. The LFW-a database is divided into 6263 training images and 6880 testing images. Classification accuracies have been reported for both these databases.

4.3. Results on Face Databases

Predefined experimental protocols are followed for all the databases, as discussed in Section 4.2. The results are reported on the YTF [32], PaSC [1], CelebA [21] and LFW-a [33] databases.

Face recognition: The results are shown on the YTF and PaSC databases. The results have been compared with state-of-the-art approaches (Table 1). The proposed approach is second only to VGGFace on YTF, and surpasses the state-of-the-art results in all other cases. Figure 4 shows the ROC curves of the proposed approach versus the base models (DL-1: DenseNet and DL-2: ResNet) on the YTF and PaSC databases. It is evident that the proposed approach significantly improves upon the base models.

Table 1: Results on YTF, PaSC Handheld, and PaSC Control databases. Verification accuracies on the YTF database are reported at equal error rate while the results on PaSC are reported at 1% false accept rate. Top two results are highlighted in the table.

Algorithm	**YTF**	**PaSC**	
		Handheld	**Control**
Trunk-Branch Ensemble CNNs with Batch Normalization [9, 27]	94.9%	**97.0%**	98.0%
VGG Face [9, 26]	**97.4%**	87.0%	91.3%
SDAE-DBM Joint Representation [12]	95.4%	97.2%	**98.1%**
DenseNet (DL-1)	95.0%	93.0%	93.7%
ResNet (DL-2)	92.1%	82.9%	79.9%
Proposed LC-DECAL	**96.3%**	**97.6%**	**98.7%**

Table 2: Classification accuracies on the CelebA and LFW-a databases. Best reported results are highlighted in the table.

Algorithm	**CelebA**	**LFW-a**
MCNN +AUX [16]	91.29%	86.31%
DMTL [15]	92.60%	86.15%
Proposed LC-DECAL	**92.86%**	**87.11%**

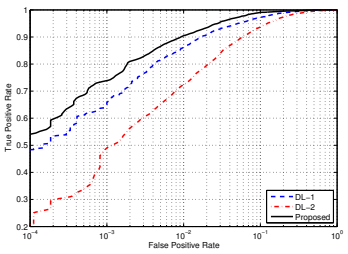
Attribute Classification: The results are computed on two databases (CelebA and LFW-a), and the accuracies are compared with two state-of-the-art attribute classification algorithms. As shown in Table 2, the proposed algorithm yields the best results on both the databases.

4.4. Analysis of Proposed Approach

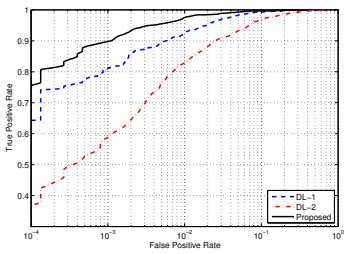
The proposed Label Consistent Deep Collaborative Learning (LC-DECAL) framework has three major components: (1) Co-training, (2) Transfer learning, and (3) Label Consistency. A component wise analysis is performed to analyze the usefulness of the three individual components.

The base models were trained on the target domain labeled data. The models were provided with additional labeled data for transfer learning, and unlabeled data for co-training. Tables 3 and 4 summarize the results for the base models, and for each technique when it is combined with the base models for different databases.

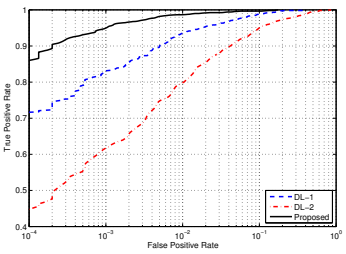
Co-training: Co-training works when we have two independent classifiers with decent accuracies and additional unlabeled data. Co-training assigns pseudo-labels to the unlabeled data using the two classifiers, and then the classifiers are trained with the pseudo-labeled data. For the experiments, DenseNet is chosen as view1 (DL-1) and ResNet as



(a) For the YTF database



(b) For the PaSC Handheld database



(c) For the PaSC Control database

Figure 4: ROC Curves (DL-1: DenseNet, DL-2: ResNet)

Table 3: Analyzing the effect of each component of the proposed LC-DECAL framework.

Database	Model	DenseNet	ResNet
YTF	Base Model	95.0%	92.1%
	With Co-training	95.4%	92.9%
	With Transfer Learning	95.7%	93.1%
	With Label Consistency	95.6%	93.2%
PaSC Handheld	Base Model	93.0%	82.9%
	With Co-training	94.1%	84.8%
	With Transfer Learning	94.4%	86.5%
	With Label Consistency	94.7%	86.7%
PaSC Control	Base Model	93.7%	79.9%
	With Co-training	94.5%	83.3%
	With Transfer Learning	94.9%	84.1%
	With Label Consistency	94.9%	84.4%

view2 (DL-2). We see that co-training improves the performance. In the best case, the accuracy improved by 4.43% when using ResNet for the LFW-a database (Table 4).

Transfer learning: For transfer learning, a pre-trained source domain classifier was used along with the base model. It provided some improvements in the accuracies of both the models and also, performs better than co-training in all cases. On using ResNet for the LFW-a database, transfer learning increases accuracy of base model by 4.57%. Thus, it is understood that it would be extremely beneficial to use transfer learning if there is enough labeled data available in both source and target domains. Co-training solves the problem of availability of labeled data in target domain, thus justifying the utility of the proposed approach.

Label Consistency: Label consistency invariably increased the accuracy of the base models. It increased the accuracy of ResNet by 4.6% and 4.5% for the LFW-a and PaSC Control databases respectively.

Statistical Analysis: For statistical analysis, we performed the McNemar test [24] and compared our approach to the second best result on each database. On the PaSC, CelebA and LFW-a databases, at 99% confidence, the proposed approach is found to be statistically different and performs better than other approaches. On the YTF database, we

Table 4: Analyzing the effect of each component of the proposed LC-DECAL on the CelebA and LFW-a databases.

Database	Model	DenseNet	ResNet
CelebA	Base Model	84.42%	80.37%
	With Co-training	85.04%	84.55%
	With Transfer Learning	85.76%	84.91%
	With Label Consistency	85.92%	84.07%
LFW-a	Base Model	80.29%	76.42%
	With Co-training	82.75%	80.85%
	With Transfer Learning	82.97%	80.99%
	With Label Consistency	83.04%	81.02%

combined the results of all the folds (of 10-fold cross validation) to form 5000 decisions and applied McNemar Test. At 99% confidence interval, the null hypothesis is not rejected, i.e., the results of the proposed algorithm and VGG-Face on the YTF database are statistically not different.

5. Conclusion and Future Work

This paper introduces the LC-DECAL framework in which an ensemble of classifiers are used to (i) collaborate to pseudo-label the unlabeled data and (ii) combine knowledge from the source and target domains using transfer learning. The framework requires that the source and target domains are similar, so as to avoid divergence in training due to wrong pseudo-labeling. Further, the label consistency is incorporated to learn discriminative features. The results on face databases show that LC-DECAL improves upon the performance of the base classifier and achieves state-of-the-art results. This approach can be extended to include multiple source domains or allow collaboration for less similar tasks.

6. Acknowledgements

M. Vatsa and R. Singh are partly supported by the Infosys Center for AI at IIIT Delhi. M. Vatsa is also supported through the Swarnajayanti Fellowship by Government of India.

References

[1] J. Beveridge, P. Phillips, D. Bolme, B. Draper, G. Givens, Y. Lui, M. Teli, H. Zhang, W. T. S., K. Bowyer, et al. The challenge of face recognition from digital point-and-shoot cameras. In *IEEE International Conference on Biometrics: Theory, Applications and Systems*, 2013.

[2] H. S. Bhatt, R. Singh, M. Vatsa, and N. Ratha. Matching cross-resolution face images using co-transfer learning. In *IEEE International Conference on Image Processing*, pages 1453–1456, 2012.

[3] H. S. Bhatt, R. Singh, M. Vatsa, and N. K. Ratha. Improving cross-resolution face matching using ensemble-based co-transfer learning. *IEEE Transactions on image Processing*, 23(12):5654–5669, 2014.

[4] A. Blum, N. Haghtalab, A. D. Procaccia, and M. Qiao. Collaborative pac learning. In *Advances in Neural Information Processing Systems*, pages 2389–2398, 2017.

[5] A. Blum and T. Mitchell. Combining labeled and unlabeled data with co-training. In *ACM Conference on Computational Learning Theory*, pages 92–100, 1998.

[6] H. Chen, Y. Wang, G. Wang, and Y. Qiao. Lstd: A low-shot transfer detector for object detection. *arXiv preprint arXiv:1803.01529*, 2018.

[7] N. Cherniavsky, I. Laptev, J. Sivic, and A. Zisserman. Semi-supervised learning of facial attributes in video. In *European Conference on Computer Vision*, pages 43–56, 2010.

[8] B. Deng, Q. Liu, S. Qiao, and A. Yuille. Unleashing the potential of cnns for interpretable few-shot learning. *arXiv preprint arXiv:1711.08277*, 2017.

[9] C. Ding and D. Tao. Trunk-branch ensemble convolutional neural networks for video-based face recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2017.

[10] N. El Gayar, S. A. Shaban, and S. Hamdy. Face recognition with semi-supervised learning and multiple classifiers. In *WSEAS International Conference on Computational Intelligence, Man-Machine Systems and Cybernetics*, pages 296–301, 2006.

[11] Y. Gao, J. Ma, and A. L. Yuille. Semi-supervised sparse representation based classification for face recognition with insufficient labeled samples. *IEEE Transactions on Image Processing*, 26:2545–2560, 2017.

[12] G. Goswami, M. Vatsa, and R. Singh. Face verification via learned representation on feature-rich video frames. *IEEE Transactions on Information Forensics and Security*, 12(7):1686–1698, 2017.

[13] M. Grgic, K. Delac, and S. Grgic. Scface — surveillance cameras face database. *Multimedia Tools and Applications Journal*, 51(3):863–879, Feb. 2011.

[14] R. Gross, I. Matthews, J. Cohn, T. Kanade, and S. Baker. Multi-pie. *Image and Vision Computing*, 28(5):807–813, 2010.

[15] H. Han, A. Jain, S. Shan, and X. Chen. Heterogeneous face attribute estimation: A deep multi-task learning approach. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2017.

[16] E. Hand and R. Chellappa. Attributes for improved attributes: A multi-task network utilizing implicit and explicit relationships for facial attribute classification. In *Association for the Advancement of Artificial Intelligence Conference*, pages 4068–4074, 2017.

[17] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 770–778, 2016.

[18] G. Huang, Z. Liu, K. Q. Weinberger, and L. van der Maaten. Densely connected convolutional networks. In *IEEE Conference on Computer Vision and Pattern Recognition*, volume 1, page 3, 2017.

[19] S. Ioffe and C. Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *arXiv preprint arXiv:1502.03167*, 2015.

[20] M. M. Islam, X. Yao, and K. Murase. A constructive algorithm for training cooperative neural network ensembles. *IEEE Transactions on Neural Networks*, 14(4):820–834, 2003.

[21] Z. Liu, P. Luo, X. Wang, and X. Tang. Deep learning face attributes in the wild. In *International Conference on Computer Vision*, December 2015.

[22] F. Ma, D. Meng, Q. Xie, Z. Li, and X. Dong. Self-paced co-training. In *International Conference on Machine Learning*, pages 2275–2284, 2017.

[23] A. Majumdar, R. Singh, and M. Vatsa. Face verification via class sparsity based supervised encoding. *IEEE transactions on pattern analysis and machine intelligence*, 39(6):1273–1280, 2016.

[24] Q. McNemar. Note on the sampling error of the difference between correlated proportions or percentages. *Psychometrika*, 12(2):153–157, 1947.

[25] S. J. Pan and Q. Yang. A survey on transfer learning. *IEEE Transactions on Knowledge and Data Engineering*, 22(10):1345–1359, 2010.

[26] O. Parkhi, A. Vedaldi, A. Zisserman, et al. Deep face recognition. In *British Machine Vision Conference*, volume 1, page 6, 2015.

[27] W. Scheirer, P. Flynn, C. Ding, G. Guo, V. Struc, M. Al Jazaery, K. Grm, S. Dobrisek, D. Tao, Y. Zhu, et al. Report on the btas 2016 video person recognition evaluation. In *8th IEEE International Conference on Biometrics Theory, Applications and Systems*, 2016.

[28] M. Singh, S. Nagpal, M. Vatsa, R. Singh, and A. Noore. Supervised cosmos autoencoder: Learning beyond the euclidean loss! *arXiv preprint arXiv:1810.06221*, 2018.

[29] P. Vanhaesebrouck, A. Bellet, and M. Tommasi. Decentralized collaborative learning of personalized models over networks. In *International Conference on Artificial Intelligence and Statistics*, 2017.

[30] D. Wang, P. Cui, and W. Zhu. Deep asymmetric transfer network for unbalanced domain adaptation. In *Association for the Advancement of Artificial Intelligence Conference*, 2018.

[31] Y. Wang, L. Xie, Y. Zhang, W. Zhang, and A. Yuille. Deep collaborative learning for visual recognition. *arXiv preprint arXiv:1703.01229*, 2017.

[32] L. Wolf, T. Hassner, and I. Maoz. Face recognition in unconstrained videos with matched background similarity. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 529–534, 2011.

[33] L. Wolf, T. Hassner, and Y. Taigman. Effective unconstrained face recognition by combining multiple descriptors and learned background statistics. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(10):1978–1990, 2011.

[34] J. Yosinski, J. Clune, Y. Bengio, and H. Lipson. How transferable are features in deep neural networks? In *Advances in Neural Information Processing Systems*, pages 3320–3328, 2014.

[35] H. Yu, Z. Luo, and Y. Tang. Transfer learning for face identification with deep face model. In *International Conference on Cloud Computing and Big Data*, pages 13–18, 2016.

[36] P. Zhao and S. Hoi. Otl: A framework of online transfer learning. In *International Conference on Machine Learning*, pages 1231–1238, 2010.

[37] Y. Zhou and S. Goldman. Democratic co-learning. In *IEEE International Conference on Tools with Artificial Intelligence*, pages 594–602, 2004.