Domain Adaptation for Robust Face Recognition Using Transfer Kernel Learning

João Renato Ribeiro Manesco
UNESP - São Paulo State University
Bauru, Brazil
joao.r.manesco@unesp.br

Aparecido Nilceu Marana
UNESP - São Paulo State University
Bauru, Brazil
nilceu.marana@unesp.br

***Abstract*—In the last decades, for reasons of safety or convenience, biometric characteristics are increasingly being used to identify individuals who wish to have access to systems or places, and facial features are one of the most used characteristics for this purpose. For biometric identification to be effective, the recognition accuracy rates must be high. However, these rates can be very low depending on the difference (displacement) between the domain of the images stored in the database of the biometric system (source images) and the images used at the moment of identification (target images). In this work, we evaluated the performance of a domain adaptation method called Transfer Kernel Learning (TKL) in the face recognition problem. Results obtained in our experiments on two face datasets, ARFace and FRGC, corroborates that TKL is suitable for domain adaptation and that it is capable of improving significantly the accuracy rates of face recognition, even when considering facial images with occlusions, variations in illumination and complex backgrounds.**

***Index Terms*—biometrics, face recognition, domain adaptation, transfer kernel learning.**

I. Introduction

Recently, either for security or convenience reasons, biometric recognition is gaining popularity among applications that aim to provide access to a particular system or place. Since facial features can be extracted from most people in our society, they end up being one of the most used features for these kinds of applications [1], [2].

Even though facial features are widely used, a few factors can decrease the system's performance in facial recognition tasks. Among these, we can cite the differences among the capture sensors, illumination changes, age disparity, and occlusion of the facial region [3].

An example in which we can see these variations is in mobile banking, where they need to authenticate users with their faces, to open accounts, or authorize transactions, usually from images of distinct origins like IDs and *selfies* [4], [5]. In Figure 1 we can see those differences, wherein the upper row we have images from the Brazilian national ID card, with constant illumination, and without pose variation, whereas in the lower row we have *selfie* images with differences in the capture sensor and variations among illumination and pose.

This disparity between image domains brings up a problem called domain shift, in which the distribution of the source classification train data differs from the test data distribution, decreasing the performance of the classification task [6]. This kind of performance loss is a big problem in biometric identification systems, in which high accuracy is needed for the system to work effectively.

Fig. 1: Examples of face images obtained from four people on the domain characterized by the Brazilian national ID card pictures (upper row) and on the domain characterized by *selfie* pictures (lower row).

A way to deal with the domain shift problem is using domain adaptation techniques, a subarea of transfer learning that uses labeled data from a source domain to improve the classification task in a target domain [7].

In this paper, we approach the face recognition problem using a domain adaptation technique called Transfer Kernel Learning [8] to improve the accuracy of the identification task.

II. Domain Adaptation

Domain adaptation is a subarea of transfer learning that aims to learn from a source data distribution a model with good performance on a distinct target data distribution [9]. Pan and Yang [10] define the following key concepts related to domain adaptation theory.

Definition 1 (Domain)**.** *A domain D is composed of a feature space $\mathcal{F}$ with d dimensions and a marginal probability function $P(x)$, which means that $D = \{\mathcal{F}, P(x)\}$, with $x \in \mathcal{F}$.*

Definition 2 (Task)**.** *Given a domain D, a task $\mathcal{T}$ consists of a set of labels $\mathcal{Y}$ and a classifier $f(x)$, which means that $\mathcal{T} = \{\mathcal{Y}, f(x)\}$, with $y \in \mathcal{Y}$ and $f(x) = P(y|x)$.*

Definition 3 (Domain Adaptation)**.** *Given a source domain $\mathcal{D}_\mathcal{S}$ and a target domain $\mathcal{D}_\mathcal{T}$ and assuming that $\mathcal{D}_\mathcal{S} \neq \mathcal{D}_\mathcal{T}$ regarding their marginal probabilities $P(X^\mathcal{S}) \neq P(X^\mathcal{T})$, and two tasks $\mathcal{T}_\mathcal{S} \approx \mathcal{T}_\mathcal{T}$, with conditional distribution $P(Y^\mathcal{S}|X^\mathcal{S}) \approx P(Y^\mathcal{T}|X^\mathcal{T})$. The goal of the domain adaptation is to improve the prediction $f_\mathcal{T}(\cdot)$ in the target domain $\mathcal{D}_\mathcal{T}$, using the source domain $\mathcal{D}_\mathcal{S}$ data.*

Essentially, the domain adaptation objective is to improve the predictive characteristic of a target domain with a different marginal probability, using data found in the source domain.

A. Transfer Kernel Learning

Transfer Kernel Learning (TKL) [8] is a promising domain adaptation technique in tasks related to visual recognition. This technique aims to use source and target data to learn a domain-invariant kernel that minimizes the domain variance and is used to feed a kernel machine. Its formal problem is described in Problem 1.

Problem 1 (Transfer Kernel Learning [8])**.** *Given a labeled domain $\mathcal{Z} = \{(z_1, y_1), ..., (z_m, y_m)\}$ and an unlabeled target domain $\mathcal{X} = \{x_1, \ldots, x_n\}$, with $\mathcal{F}_\mathcal{Z} = \mathcal{F}_\mathcal{X}$, $\mathcal{Y}_\mathcal{Z} = \mathcal{Y}_\mathcal{X}$, $P(z) \neq P(x)$ and $P(y|z) \neq P(y|x)$, learn a domain-invariant kernel $k(z, x) = \langle \phi(z), \phi(x) \rangle$, such that $P(\phi(z)) \approx P(\phi(x))$. Assume $P(y|\phi(z)) \approx P(y|\phi(x))$ so kernel machines trained on $\mathcal{Z}$ can generalize well on $\mathcal{X}$.*

The TKL method follows the principle that even though metrics like the Maximum Mean Discrepancy can find information about the domain variation, they aren't explored properly, being used only as a penalty to standard learning methods and this will not properly achieve a local minimum in the variation.

The problem is explored by applying standard eigen decomposition on the target kernel matrix K_X and then evaluating the eigensystem on the source data, by using the Mercer Theorem [8], finding the extrapolated eigenvectors of the source domain data by the equation 1

$$\overline{\Phi}_Z \simeq K_{ZX} \Phi_X \Lambda_X^{-1} \tag{1}$$

in which Φ refers to the set of eigenvalues of a particular domain, Λ is set of eigenvectors, and K_{ZX} the cross-domain kernel between Z and X, evaluated using the kernel function k. The eigenvectors are then used in the Nyström Kernel Approximation [11] to find a family of kernels K_Z, extrapolated from a target eigensystem but evaluated on source data.

This family preserves the key structures of the target domain but does not necessarily minimize the domain variance, this is achieved by relaxing the target domain eigenvalues Λ_X to a set of Λ eigenvalues that can be used in a quadratic minimization problem involving $\overline{\Phi}_Z$, K_Z and a damping factor ζ that can be tuned.

After finding the optimized Λ eigenvalues, it is possible to find a domain invariant kernel matrix, $\overline{K_A}$, described in the equation 2.

$$\overline{K_A} = \begin{bmatrix} \overline{\Phi}_Z \Lambda \overline{\Phi}_Z{}^T & \overline{\Phi}_Z \Lambda \overline{\Phi}_X{}^T \\ \overline{\Phi}_X \Lambda \overline{\Phi}_Z{}^T & \overline{\Phi}_X \Lambda \overline{\Phi}_X{}^T \end{bmatrix} \tag{2}$$

By finding the domain invariant kernel matrix, we can use the source data portion $\overline{\Phi}_Z \Lambda \overline{\Phi}_Z{}^T$,or simply K_{AZZ}, to train a kernel machine, like an SVM and evaluate the performance on the target data portion K_{AXZ} found in the kernel matrix as $\overline{\Phi}_X \Lambda \overline{\Phi}_Z{}^T$. We can see the overall procedure of the TKL method in Figure 2.

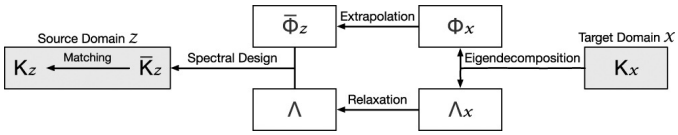

Fig. 2: Overall procedure of the Transfer Kernel Learning (TKL) method [8].

III. Face Recognition

Face recognition is the most common identification method used by humans since it has a high acceptance in society and provides a non-intrusive collaboration with the system, as opposed to iris or fingerprint recognition, in which an individual has to directly interact with the system [3].

A face recognition system can operate in two ways, authentication and identification [1]. An authentication system will match the user face with another face, of who he claims to be, acquired from a face database, and assert if he is that person. On the other hand, an identification system will receive a face as an input and will verify which person he or she is.

Two fundamental phases are required for the proper behavior of a facial recognition system, the detection of the face region, and the feature extraction of the detected faces.

A. Face Detection

Face detection is an essential phase of face recognition, responsible for detecting the face region in an image and enabling proper feature extraction in the posterior phases.

Given the wide range of variations among facial images, face detection is a challenging task, especially in an unconstrained environment, but recently, with the advances in deep learning, some very effective approaches are appearing. In our work, face detection is conducted by using a method named Multi-Task Cascaded Convolutional Neural Network (MTCNN) [12].

The MTCNN method uses a structure of three cascaded neural networks for: (i) detecting the faces in different stages, (ii) filtering the possible face regions, and (iii) refining the final result. It also returns a relationship between face detection and face alignment, returning fiducial points of the eyes and mouth, so proper alignment can be done after the detection.

In the first stage, a CNN P-NET is used to predict the probable face positions. After that a CNN R-NET is used in the second stage to filter the face region, removing the non-candidate faces. In the last stage, the output of the previous

network enters a CNN O-NET and outputs the face region, and the positions of the eyes, nose, and mouth. The whole pipeline of the MTCNN can be seen in Figure 3.

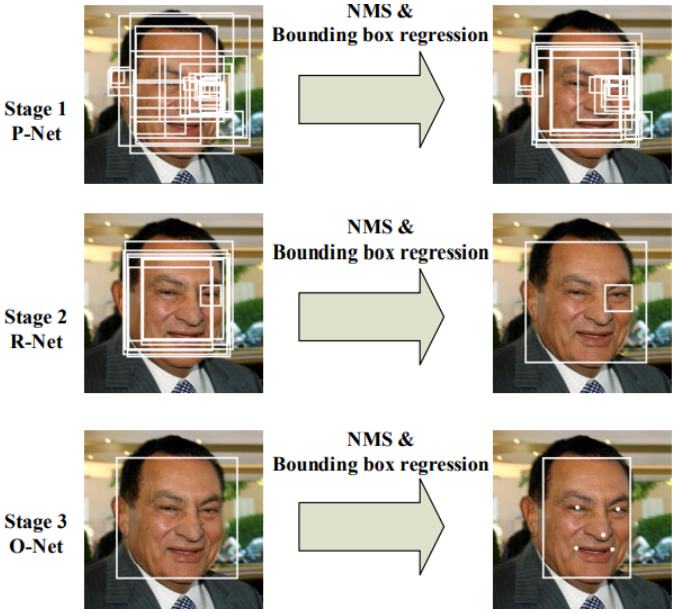

Fig. 3: Pipeline of the Multi-Task Cascaded Convolutional Neural Network face detection [12].

B. Feature Extraction

With the objective of reducing the dimensionality of the features and improve the data representation for the classification task, a Feature Extraction stage is needed, to map each face to a n-dimensional feature space. In our work, the feature extraction is done with a pre-trained convolutional neural network named VGG Face [13], whose architecture can be seen in Figure 4.

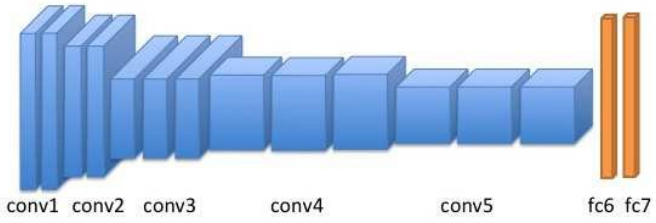

Fig. 4: VGG Face architecture [14].

VGG Face architecture is based on the VGG-16 architecture [15] and was trained in a database with 2.6 million images over a total of 2622 subjects. The input of the network consists of a $224 \times 224 \times 3$ facial image, and the output is a feature vector obtained from the fully connected layer fc7.

IV. Experiments

During all experiments, the face detection was performed using the MTCNN method, described in section III-A, after that, feature extraction was done, by inputting the face regions in the pre-trained VGG Face network described in section III-B. All the faces were normalized per channel using the standard score normalization, which can be seen in equation 3, with the mean and standard deviation values provided by the authors.

$$X_{\text{norm}} = \frac{X - \mu}{\sigma} \tag{3}$$

After feature extraction, the data was divided into different domains, according to their respective database. In all cases, the tests were carried out in the identification variation of face recognition, with three classifier instances, one K-Nearest Neighbors, with $k = 1$, a regular SVM, and the TKL method.

The parameters were also the same among all tests, with the damping factor $\zeta = 10.0$, the SVM regularization parameter was set to 1.1 and the kernel used in both the SVM and the TKL was the Gaussian kernel with $\sigma = 1.0$. All the results were compared through the accuracy metric.

A. Databases

Two databases were used for evaluation in our work, ARFace [16], and the Face Recognition Grand Challenge database [17].

1) ARFace: The ARFace [16] is a database which contains face images from 126 subjects with 26 images each, all the images are obtained in a constrained background with different contexts. In Figure 5 it is possible to see the different contexts available in the database, involving variations in facial expression, illumination, ocular region occlusion, and mouth region occlusion.

Fig. 5: Context differences in the ARFace database, involving variations in facial expression, illumination, ocular region occlusion, and mouth region occlusion [16], [18].

For the domain adaptation task, the following domains were proposed for analysis:

- **N:** Faces with neutral and other expression variations;
- **O:** Faces with occlusion in the ocular region;
- **C:** Faces with occlusion in the mouth region;
- **I:** Faces with illumination variations.

B. Face Recognition Grand Challenge

The Face Recognition Grand Challenge (FRGC) [17] is a database proposed to advance and develop research in face recognition.

The database contains colored facial (RGB) images collected on different seasons for three years, captured in a constrained or unconstrained setting. It also provides three dimensional face image data for 3D face recognition tasks. In Figure 6 we can see the differences between the two settings, in the first two columns we have images obtained in a constrained environment with face expression variations, while in the last three columns, we have images obtained in unconstrained environments, with differences in the background complexity and the illumination intensity.

Fig. 6: Examples of images of the FRGC database [17].

For the domain adaptation task, the constrained images were used as the source domain and the unconstrained images were used as the target domain.

V. Results

In Table I we can see the accuracy rates obtained on the three classification tasks for the ARFace database in the identification face recognition. As aforementioned, **N** refers to facial images in neutral or with expression variations, **I** refers to facial images with changes on illumination, **O** refers to facial images with occlusion in the ocular region, and **C** refers to facial images with occlusion in the mouth region. In the notation **X**→**Y**, **X** represents the source domain and **Y** represents the target domain. In our experiments, the neutral setting **N** was always used as the source domain, while the other settings were used as target domains.

Method	N→O	N→C	N→I
1-NN	67.29%	95.89%	99.62%
SVM	64.81%	94.52%	**100%**
TKL	**89.73%**	**97.71%**	99.76%

TABLE I: Accuracy rates obtained on ARFace database considering the identification task (**N** refers to faces with neutral or with expression variations, **I** refers to faces with changes on illumination, **O** refers to faces with occlusion in the ocular region, and **C** refers to faces with occlusion in the mouth region).

As we can see, the TKL method performed very well in all settings. Particularly, when classifying faces with occlusion in the ocular region, **N**→**O**, the domain adaptation provided by TKL greatly improved the classification results. Another important result that must be noted is that the feature vector obtained from the fully connected layer fc7 of VGG-Face showed to be robust when dealing with changes in illumination, given that the accuracy rates on the domain adaptation **N**→**I** were very high and they did not change that much for the three compared methods (SVM obtained 100% of accuracy rate, TKL obtained 99.76% and 1-NN obtained 99.62%).

The results obtained on the ARFace dataset also tell us about the importance of the ocular region for face recognition, since when this area is occluded, the accuracy rates drop significantly for the three assessed methods. In this case, **N**→**O**, the domain adaptation provided by TKL was of paramount importance to overcome this problem.

Regarding the experiments carried out on the FRGC dataset, Figure 7 shows the results obtained. In these experiments, the constrained images were used as the source domain, while the unconstrained images were used as the target domain.

We can see in Figure 7 that the TKL method provided, also for this more challenging dataset and difficult settings, significant gain in the accuracy rates, showing its suitability for facial recognition tasks. While TKL obtained an accuracy rate of 82.63%, the second best result, reached by SVM, was 76.35%, that is a 6.28% lower result.

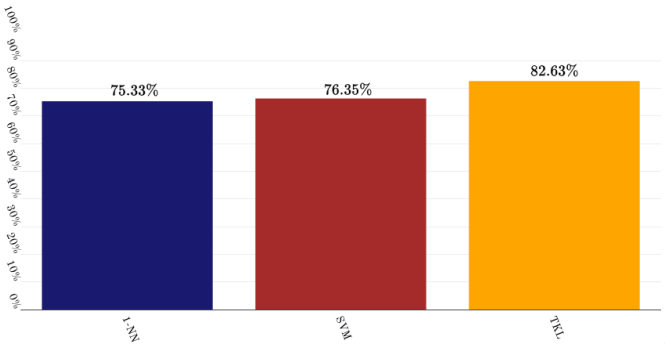

Fig. 7: Face recognition accuracy rates obtained by 1-NN, SVM and TKL methods on the FRGC database.

Since the purpose of this paper is to verify the effectiveness of domain adaptation methods in the facial identification task, this paper focused on a domain adaptation protocol for its experiments, therefore it would be unfair to compare the results with methods that follow different protocols and approach a different recognition task. That being the case, the importance of domain adaptation tasks for face recognition can be verified and even different state of the art methods could benefit from using them.

VI. Conclusions

In this work we evaluated the performance of a domain adaptation method called Transfer Kernel Learning (TKL) in the face recognition problem. Results obtained in experiments

carried out on two face datasets, ARFace and FRGC, corroborate the results found in literature that TKL is a powerful method for domain adaptation. Besides, the results showed that TKL is capable of improving the accuracy rates of face recognition, even when considering challenging scenarios, with face images presenting occlusions, variations in illumination and complex backgrounds.

VII. Acknowledgments

This paper is a result of the ongoing research of the Scientific Initiation named Robust Face Recognition Based on Domain Adaptation and has the financial support of FAPESP, process nº: 2019/15357-8.

References

[1] A. K. Jain and S. Z. Li, *Handbook of face recognition*. Springer, 2011.
[2] C. Ding and D. Tao, "A comprehensive survey on pose-invariant face recognition," *ACM Transactions on intelligent systems and technology (TIST)*, vol. 7, no. 3, p. 37, 2016.
[3] A. K. Jain, A. A. Ross, and K. Nandakumar, *Introduction to biometrics*. Springer Science & Business Media, 2011.
[4] G. Folego, M. A. Angeloni, J. A. Stuchi, A. Godoy, and A. Rocha, "Cross-domain face verification: Matching id document and self-portrait photographs," *arXiv preprint arXiv:1611.05755*, 2016.
[5] J. S. Oliveira, G. B. Souza, A. R. Rocha, F. E. Deus, and A. N. Marana, "Cross-domain deep face matching for real banking security systems," in *2020 Seventh International Conference on eDemocracy & eGovernment (ICEDEG)*. IEEE, 2020, pp. 21–28.
[6] W. M. Kouw and M. Loog, "An introduction to domain adaptation and transfer learning," *arXiv preprint arXiv:1812.11806*, 2018.
[7] G. Csurka, "Domain adaptation for visual applications: A comprehensive survey," 2017.
[8] M. Long, J. Wang, J. Sun, and S. Y. Philip, "Domain invariant transfer kernel learning," *IEEE Transactions on Knowledge and Data Engineering*, vol. 27, no. 6, pp. 1519–1532, 2015.
[9] V. M. Patel, R. Gopalan, R. Li, and R. Chellappa, "Visual domain adaptation: A survey of recent advances," *IEEE signal processing magazine*, vol. 32, no. 3, pp. 53–69, 2015.
[10] S. J. Pan and Q. Yang, "A survey on transfer learning," *IEEE Transactions on Knowledge and Data Engineering*, vol. 22, no. 10, pp. 1345–1359, Oct 2010.
[11] C. K. Williams and M. Seeger, "Using the nyström method to speed up kernel machines," in *Advances in neural information processing systems*, 2001, pp. 682–688.
[12] K. Zhang, Z. Zhang, Z. Li, and Y. Qiao, "Joint face detection and alignment using multitask cascaded convolutional networks," *IEEE Signal Processing Letters*, vol. 23, no. 10, pp. 1499–1503, Oct 2016.
[13] O. M. Parkhi, A. Vedaldi, and A. Zisserman, "Deep face recognition," in *Proceedings of the British Machine Vision Conference (BMVC)*. BMVA Press, September 2015, pp. 41.1–41.12.
[14] M. Nakada, H. Wang, and D. Terzopoulos, "Acfr: Active face recognition using convolutional neural networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2017, pp. 35–40.
[15] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.
[16] A. Martinez and R. Benavente, "The AR Face Database," Tech. Rep., June 1998.
[17] P. J. Phillips, P. J. Flynn, T. Scruggs, K. W. Bowyer, and W. Worek, "Preliminary face recognition grand challenge results," in *7th International Conference on Automatic Face and Gesture Recognition (FGR06)*. IEEE, 2006, pp. 15–24.
[18] J. Zhou and B. Zhang, "Collaborative representation using non-negative samples for image classification," *Sensors*, vol. 19, no. 11, p. 2609, 2019.