Heterogeneous Face Recognition Using Domain Specific Units

Tiago de Freitas Pereira, *Member, IEEE*, André Anjos, and Sébastien Marcel, *Senior Member, IEEE*

***Abstract*— The task of Heterogeneous Face Recognition consists in matching face images that are sensed in different domains, such as sketches to photographs (visual spectra images), and thermal images to photographs or near-infrared images to photographs. In this paper, we suggest that the high-level features of Deep Convolutional Neural Networks trained in visual spectra images are potentially domain independent and can be used to encode faces sensed in different image domains. A generic framework for Heterogeneous Face Recognition is proposed by adapting Deep Convolutional Neural Networks low-level features in, so-called, Domain Specific Units. The adaptation using the Domain Specific Units allows the learning of shallow feature detectors specific for each new image domain. Furthermore, it handles its transformation to a generic face space shared between all image domains. Experiments carried out with four different face databases covering three different image domains show substantial improvements, in terms of recognition rate, surpassing the state-of-the-art for most of them. This work is made reproducible: all the source code, scores, and trained models of this approach are made publicly available.**

***Index Terms*— Face recognition, heterogeneous face recognition, reproducible research, domain adaptation, deep neural networks.**

I. Introduction

FACE recognition has existed as a field of research for more than 30 years and has been particularly active since the early 1990s. Researchers of many different fields (from psychology, pattern recognition, neuroscience, computer graphics and computer vision) have attempted to create and understand face recognition [2].

Heterogeneous Face Recognition (HFR) consists in matching faces from different image modalities. Figure 1 demonstrates possible HFR comparison scenarios. Use-cases are many, even in situations where no real face even exists such as in sketch recognition.

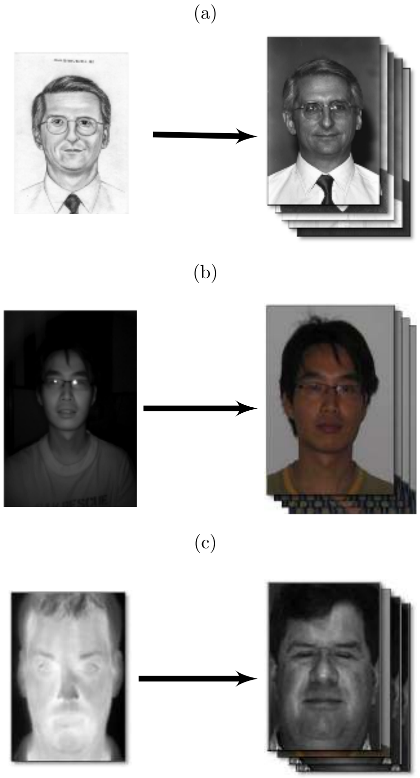

Fig. 1. Possible HFR comparison scenarios. (a) VIS-Sketches. (b) VIS-NIR. (c) VIS-Thermal.

The key difficulty in matching faces from heterogeneous conditions is that images of the same subject may differ in appearance due to changes in image domain, e.g. between visual spectra images (VIS) and near-infrared images (NIR), between VIS images and sketches. This shift introduces high within class variations, and a direct comparison of images across these domains can potentially degrade recognition accuracies.

The contributions of this work are three fold. First, we analyze the effectiveness of some state-of-the-art Deep Convolutional Neural Networks (DCNN) architectures trained with VIS images in the HFR task. Such analysis establishes a baseline for comparison. As a second contribution we introduce a light weight framework that learns domain specific feature detectors for HFR called Domain Specific Units. The application of such framework in different HFR scenarios substantially improves the recognition rates compared with DCNNs trained with VIS images and with state-of-the-art. Finally we aim to make this reproducible: all the source code, trained models and scores are made publicly available. Details on how to reproduce this work can be found on the provided link.[1]

Manuscript received May 2, 2018; revised August 10, 2018 and October 8, 2018; accepted November 19, 2018. Date of publication December 6, 2018; date of current version March 20, 2019. This work was supported by the Swiss National Science Foundation (SNSF) under the HFACE Project. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Karthik Nandakumar. *(Corresponding author: Tiago de Freitas Pereira.)*

T. de Freitas Pereira is with the Idiap Research Institute, 1920 Martigny, Switzerland, and also with the Ecole Polytechnique Fédérale de Lausanne, 1920 Martigny, Switzerland (e-mail: tiago.pereira@idiap.ch).

A. Anjos and S. Marcel are with the Idiap Research Institute, 1920 Martigny, Switzerland.

This paper has supplementary downloadable material available at http://ieeexplore.ieee.org, provided by the author. The supplementary material contains two appendices. The first one describes how our approach performs using a completely different prior. The second one describes how the prior was trained.

Digital Object Identifier 10.1109/TIFS.2018.2885284

[1] https://gitlab.idiap.ch/bob/bob.paper.tifs2018_dsu

The organization of the paper is the following. In Section II we present prior work for Heterogeneous Face Recognition and databases used in this work. In Section III we provide an overview of Face Recognition using Deep Convolutional Neural Networks establishing baseline recognition rates. In Section IV we present our Domain Specific Units framework followed by Section V with the presentation and analysis of our experiments. Finally in Section VI, we address conclusions and possible routes to pursue.

II. Formalized Related Work

In this section we formalize the task of Face and Heterogeneous Face Recognition using the notations from [12]. Then we review prior work in HFR by connecting it to the introduced notation. We also introduce an overview of datasets for HFR.

A. Formalization of HFR

Given a domain $\mathcal{D}$ composed by a d-dimensional feature space $X \in \mathbb{R}^d$ and marginal distribution $P(X)$, the face recognition task $\mathcal{T}^f$ can be defined by a label space Y whose conditional probability is $P(Y|X, \Theta)$, where X and Y are random variables and Θ are model parameters. Given a face dataset $X = \{x_1, x_2, \ldots, x_n\}$ with their corresponding identities $Y = \{y_1, y_2, \ldots, y_n\}$, $P(Y|X, \Theta)$ can be learnt via any supervised machine learning strategy.

Let's assume now that we have two domains $\mathcal{D}^s = \{X^s, P(X^s)\}$ and $\mathcal{D}^t = \{X^t, P(X^t)\}$ called respectively **source domain** and **target domain** with both sharing the same set of labels Y. Hence, the goal of Heterogeneous Face Recognition task $\mathcal{T}^h$ is to find a Θ, where $P(Y|X^s, \Theta) = P(Y|X^t, \Theta)$.

Several assumptions to model Θ were proposed during the last years and [1] organized such techniques into three main categories and they are described in the following three subsections.

B. Synthesis Methods

In these methods a synthetic version of $\mathcal{D}^s$ is generated from $\mathcal{D}^t$. Once a synthetic version from $\mathcal{D}^t$ is generated, the matching can be done with regular face recognition approaches. Wang *et al.* [9] proposed a patch based synthesis in order to synthesize VIS images to sketches and vice-versa using Multiscale Markov Random Fields. They evaluated the synthetic images using several face recognition algorithms, such as Eigenfaces, Fisherfaces, dual space LDA [23] and Random Sampling LDA [24] with a combination of three photo-sketch databases[2] (CUHK, XM2VTS and the AR database). Jin *et al.* [25] learnt a pixel level mapping between VIS images and viewed sketches with Locally Linear Embeddings (LLE). In [13], it was proposed a model based on Generative Adversarial Networks (GANs) in order to reconstruct thermogram images from visual spectra images for further identification using the Pola Thermal dataset [14]. The identification was carried out using the Visual Geometry Group (VGG) network [15] embeddings and achieving an average Equal Error Rate of 34.58%. Similarly, Zhang *et al.* [16] also proposed a strategy based on GANs to synthesize thermograms to visual light images for further comparison using the VGG embeddings. Experiments using the the Iris dataset[3] (with 29 subjects in total) showed a rank one recognition rate of 19%.

C. Crafted Features-Based Methods

In these methods raw face images from both domains ($\mathcal{D}^s$ and $\mathcal{D}^t$) are encoded with descriptors that are invariant between them. Liao *et al.* [17] proposed a method that normalizes both VIS and NIR images using Tan & Triggs filter [18]. The local descriptor MutiScale Local Binary Patterns (MLBP) [19] (with different radii) is extracted from each one of the pre-processed images and after a feature selection step, LDA is used to classify each subject. A verification rate of 67.5% was reported under a false acceptance rate of 0.1% on the CASIA-HFB [8] database. Similarly, Sifei *et al.* [20] used a set of different band-pass filters to "normalize" both VIS and NIR images for subsequent recognition. A rank one recognition rate of 98.51% was reported in the same dataset. Inspired in gravitational fields to model pixel values, Roy *et al.* [6] proposed a illumination invariant feature extractor that requires no training. Experiments carried out with CUHK-CUFS with a biased protocol (see Section II-E.1) and the CASIA HFB [8] showed a rank one recognition rate of 99.96% and 99.78% respectively.

D. Feature Learning Based Methods

The idea of these approaches is to learn a joint mapping between $\mathcal{D}^s$ and $\mathcal{D}^t$ where the image projections from those domains can be directly compared. Klare *et al.* [1] proposed a generic framework in which faces are represented in terms of nonlinear similarities (via a kernel function) to a collection of prototype face images from different modalities. The proposed approach, called prototype random subspace (P-RS) was benchmarked on three different heterogeneous scenarios: NIR to VIS, thermal images to VIS, sketch to VIS. VIS-sketch reference results were reported using the CUHK-CUFS database with a rank one recognition rate of 99%. As a VIS-NIR reference, the CASIA HFB was used and a rank one recognition rate of 98% was reported.

Jin *et al.* [26] proposed a filter learning approach where the goal is to find the convolutional filter α where the pixel difference between images from different modalities are the minimum. Experiments with CUHK-CUFSF showed an average rank one recognition rate of 81.3%.

Based on DCNNs to model the joint mapping between $\mathcal{D}^s$ and $\mathcal{D}^t$, [54] proposes a framework for VIS-NIR face matching where the low level feature detectors are learnt with VIS images only. The high level feature detectors are jointly learnt with VIS and NIR images and it is divided in: NIR layers, VIS layers and NIR-VIS shared layers

[2] http://mmlab.ie.cuhk.edu.hk/archive/facesketch.html

[3] IEEE OTCBVS WS Series Bench; DOE University Research Program in Robotics under grant DOE-DE-FG02-86NE37968; DOD/TACOM/NAC/ARC Program under grant R01-1344-18; FAA/NSSA grant R01-1344-48/49; Office of Naval Research under grant N000143010022.

(which are domain invariant). Experiments carried out using the CASIA NIR-VIS 2.0 dataset showed an average rank one recognition rate of 95.82%. An extension of this work is presented in [56], where the Wasserstein distance between the NIR and VIS signal distributions is incremented to cost function. Experiments with CASIA NIR-VIS 2.0 dataset showed an average rank one recognition rate of 98.7%.

Built on top of Gaussian Mixture Models (GMMs), [27] model the map between $\mathcal{D}^s$ and $\mathcal{D}^t$ as a channel offset of GMM mean supervectors. Experiments with CUHK-CUFS and CASIA NIR-VIS 2 showed an average recognition rate of 96.93% and 72.39% respectively.

Based on Geodesic Flow Kernels (GFK) the work in [28] models $\mathcal{D}^s$ and $\mathcal{D}^t$ in separated d-dimensional linear subspaces (ϕ_s and ϕ_t) and embeds them onto a Grassmann manifold. Then, an explicit map between these subspaces is built (called Geodesic Flow); such map is encoded in a kernel G. The comparison between projected samples from the source domain (x_s) and target domain (x_t) is carried out by the kernalized dot product $\phi_s(x_s) \cdot G \cdot \phi_t(x_t)$. An Equal Error Rate of 1.65% was achieved under the Cross-Spectral Iris/Periocular Recognition Competition [28].

E. HFR Databases

Several databases were built along the years to support Heterogeneous Face Recognition research. This work reports experimental results and analysis under five different image databases publicly available covering three different pairs of image domains: VIS to Sketches, VIS to NIR and VIS to Thermal. The next subsections describe each one and their respective evaluation protocols.

1) CUHK Face Sketch Database (CUFS): CUHK Face Sketch Database (CUFS) is composed by viewed sketches. The viewed sketches are made by an artist looking to the corresponding photograph of a subject. It includes 188 faces from the Chinese University of Hong Kong (CUHK) student database, 123 faces from the AR database and 295 faces from the XM2VTS database. There are 606 face images in total. For each face image there is a sketch drawn by an artist based on a photo taken in a frontal pose, under normal lighting condition and with a neutral expression.

Unfortunately there is no defined evaluation protocol established for this database. Each work that uses this database implements a different way to report results. For comparison reasons, we will follow the same strategy as in [1] and do 5 fold cross-validation splitting the 606 identities in two sets with 404 identities for training and 202 for testing and use the average rank one recognition rate, in the evaluation set as a metric. For reproducibility purposes, this evaluation protocol is published in a python package format.[4] Hence, future researchers will be able to reproduce exactly the same tests with the same identities in each fold.

2) CASIA NIR-VIS 2.0 Face Database (CASIA): CASIA NIR-VIS 2.0 database [7] offers pairs of mugshot images and their corresponding NIR photos. The images of this database were collected in four recording sessions: 2007 spring, 2009 summer, 2009 fall and 2010 summer, in which the first session is identical to the CASIA HFB database [8]. It consists of 725 subjects in total. There are from one to twenty two VIS and five to fifty NIR face images per subject. The eye positions are also distributed with the images.

This database has a well defined protocol and it is publicly available for download. We also organized this evaluation protocol in the same way as for CUFS database and it is also freely available for download.[5] The average rank one recognition rate in the evaluation set (called view 2) is used as evaluation metric.

3) Near-Infrared and Visible-Light (NIVL) Dataset: Collected by University of Notre Dame, the NIVL contains VIS and NIR face images from the same subjects. The capturing process was carried out over the course of two semesters (fall 2011 and spring 2012) [22]. The dataset contains a total of 574 subjects with 2,341 VIS and 22,264 NIR images. A total of 402 subjects had both VIS and NIR images acquired during at least one session during both the fall and spring semesters.

Originally this dataset was designed and released with the intention of evaluate the error rates of commercial face recognition matchers in the VIS-NIR task under different image processing algorithms. Since there is no need to train background models for commercial matchers, the original database evaluation protocol does not have a training set. In order to evaluate our proposed approach we designed a 5-fold cross-validation strategy, where the 574 subjects were split in 344 identities for training and 230 identities for test. The average rank one recognition rate in the test set is used as evaluation metric. This evaluation protocol is equally available for download.[6]

4) Polarimetric and Thermal Database (Pola Thermal): Collected by the U.S. Army Research Laboratory (ARL), the Polarimetric Thermal Face Database (first of this kind), contains polarimetric LWIR (long-wave infrared) imagery and simultaneously acquired visible spectrum imagery from a set of 60 distinct subjects [14].

Two types of thermal images are provided in this database, the first one is the Conventional Thermal [Figure 1 (c)] and the Polarimetric Thermal. In this work, we present results only using the Conventional Thermal images. As opposed to the original protocol, that proposes a 100-fold cross-validation evaluation, we applied a 5-fold cross validation evaluation protocol where the 60 clients are split in 25 identities for training and 35 identities for testing. The average rank one recognition rate in the test set is used as evaluation metric. The protocol called "overall", which probes data from the 3 ranges, is used in this work. This evaluation protocol is also available for download.[7]

III. From Face Recognition to Heterogeneous Face Recognition

The success of Deep Convolutional Neural Networks in computer vision research, the availability of several

[4] https://pypi.python.org/pypi/bob.db.cuhk_cufs

[5] https://pypi.python.org/pypi/bob.db.cbsr_nir_vis_2

[6] https://pypi.python.org/pypi/bob.db.nivl

[7] https://pypi.python.org/pypi/bob.db.pola_thermal

TABLE I

AVERAGE RANK ONE RECOGNITION RATE UNDER SIX FACE RECOGNITION CNN SYSTEMS

Database	**L.CNN**	**VGG16-Face**	**IncepResN v2-gray**	**IncepResN v2**	**IncepResN v1-gray**	**IncepResN v1**	Best SOTA
CUFS	76.63 (2.9)	73.17 (1.6)	67.03 (2.3)	67.62 (2.6)	69.80 (3.2)	81.48 (2.6)	99. (-)[1]
CASIA	65.17 (0.6)	67.92 (1.4)	73.80 (1.2)	79.92 (0.9)	74.25 (1.3)	81.79 (1.6)	98.7 (0.3)
NIVL	86.24(1.4)	89.99 (0.6)	88.14 (0.6)	86.06 (1.3)	87.48 (1.3)	92.77 (0.4)	- (-)[2]
PolaThermal	22.36 (3.6)	15.43 (2.6)	17.8 (3.3)	17.12 (2.1)	15.5 (1.9)	27.68 (1.7)	78.72 % (-)[1]

[1] There is not a precise evaluation protocol available for comparison.
[2] There is no reference baseline that considers a training set

frameworks to instrument such networks and the possibility to work with massive amounts of labeled data (CASIA WebFace [29], MS-Celeb [30] and Megaface [31]) made face recognition error rates decrease steadily.

Despite the lack of deep understanding on why such neural networks work so well in several different pattern recognition tasks [35], practical heuristics were developed in the last five years to regularize the training and they are responsible for its success in practice. Among those, we would like to highlight three that, in our experience, have direct impact in boosting face recognition rates:

A. VGG Networks

The VGG networks [45] were the first to use small kernels in each convolutional layer (3×3). Chained in a long sequence of convolutions, such small filters followed by sub-samplings, are able to detect image symmetries in larger areas of the face image that was thought possible only via larger kernels (9×9 or 11×11) such as in the Alexnet [47].

B. Inception Modules

Szegedy *et al.* [32] introduced Inception modules. Composed by a parallel combination of different convolutional kernels (1×1, 3×3, and 5×5), such idea allows a dramatic reductions of free parameters to be learnt, increasing the recognition accuracies and generalization for several computer vision tasks.

C. Residual Connections

Practical evidences in several areas of computer vision have shown that depth of a DCNN seems to be a crucial factor in terms for accurate learning. One of the main obstacles to explore depth in CNNs is the well know gradient vanishing/exploding [33] problem. He *et al.* [34] approached this issue by passing the output of one intermediate layer and concatenating as the input of one of the layers ahead (two or three layers). Such approach allowed the training of CNNs larger than 1000 layers.

Grounded by the aforementioned seminal improvements, several face models with remarkable recognition rates in various face databases were made public available.

In this work, we will explore six different DCNN models based on three base architectures for HFR. This set of experiments establishes the baseline results for further analysis. The first is the **VGG16-Face** network [15], which was made publicly available by the Visual Geometry Group[8] and consists of 16 hidden layers where the first 13 are composed by convolutions and pooling layers. The last three layers are fully-connected (named fc6, fc7, and fc8). As a feature representation, we use the embeddings produced by the 'fc7' layer. The input signal of such network are RGB images of 224×224 pixels. Since all our databases are one channel only (NIR, Sketch and Thermal), we convert them from one channel images to three channels by replicating the signal along the extra channels.

The second network used is the **Light CNN**. Xiang *et al.* [36] proposed an architecture that has ten times less free parameters than the VGG16-Face and claimed that it is naturally able to handle mislabeled data during the training (very common in datasets mined automatically). This is achieved through the use of a newly introduced Max-Feature-Map (MFM) activation. The input signal of such network are gray scaled images of 112×112.

The third one is the **Facenet** by David Sandberg [51]. This is the closest open-source implementation of the model proposed in [37], where neither training data or source code were made available. Sandberg's FaceNet implements an Inception-ResNet v1 and Inception-ResNet v2 CNN architectures [38]. For this evaluation we have used the 20170512-110547 model (Inception-ResNet v1), trained on the MS-Celeb-1M dataset [30], which input signals are RGB images of 160×160 pixels. Furthermore, we trained ourselves two Inception-ResNet v2 models, one with gray scaled images and one with RGB using the CASIA WebFace [29] dataset and one Inception-ResNet v1 with gray scaled images. Both models work with images of 160×160 pixels as input and its source code is available for download.[9]

Summarizing, we have six different deep face models representing the VIS source domain $\mathcal{D}^s$, the **VGG16-Face**, **LightCNN**, **Inception-ResNet v1**, **Inception-ResNet v1**, **Inception-ResNet v2** and **Inception-ResNet v2-gray**. Comparisons between samples are made with the embeddings of each DCNN using the cosine similarity metric. Given the embeddings e_s and e_t from source and target domains respectively, the similarity S is given by Equation 1.

$$S(e_s, e_t) = \frac{e_s \cdot e_t}{\|e_s\|_2 \cdot \|e_t\|_2} \tag{1}$$

Table I presents the average rank one recognition rate for each face recognition baseline with their corresponding best

[8] http://www.robots.ox.ac.uk/ vgg/

[9] https://gitlab.idiap.ch/bob/bob.bio.face_ongoing

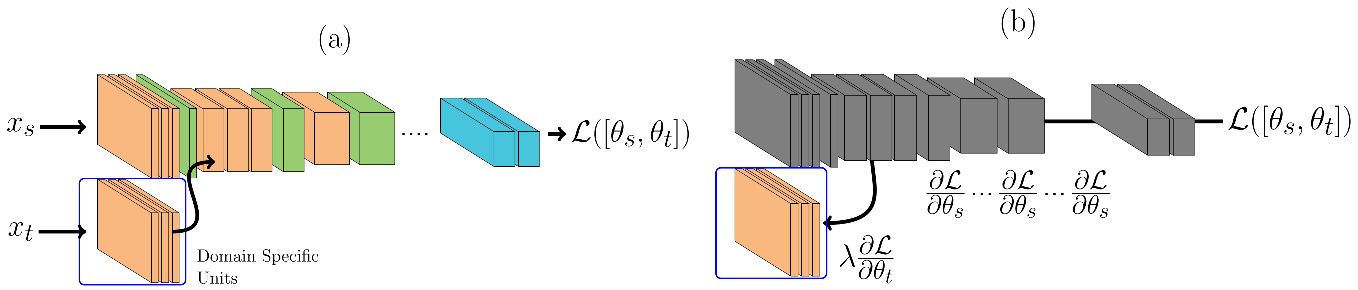
(a)
x_s
x_t
....
$\mathcal{L}([\theta_s, \theta_t])$
Domain Specific
Units
(b)
$\mathcal{L}([\theta_s, \theta_t])$
$\frac{\partial \mathcal{L}}{\partial \theta_s} \cdots \frac{\partial \mathcal{L}}{\partial \theta_s} \cdots \frac{\partial \mathcal{L}}{\partial \theta_s}$
$\lambda \frac{\partial \mathcal{L}}{\partial \theta_t}$

Fig. 2. Domain Specific Units learnt with Siamese neural networks given a pair of samples x_s and x_t from source and target domain respectively. (a) Forward pass behaviour. (b) Backward pass behaviour.

state-of-the-art recognition rates (Best SOTA). It's possible to observe, that despite the fact such DCNNs don't have any prior knowledge about $\mathcal{D}^t$, the feature detectors of such models were still able to detect discriminant features in all them (above a hypothetical random classifier). However, those recognition rates are lower than the state-of-the-art recognition rates in each image database (which consider a joint modeling of both $\mathcal{D}^s$ and $\mathcal{D}^t$).

The VIS-NIR databases (CASIA and NIVL) presented the highest rank one recognition rates in the majority of the tests. For instance, the best DCNN model in CASIA (Inception-ResNet v1) achieved a rank one recognition rate of 81.79%. For NIVL, which compared with CASIA has higher resolution images, the average rank one recognition rate is even better (92.77%). Among all image domains, NIR seems to be visually similar to VIS images, which can explain why the feature detectors from our $\mathcal{D}^s$ are very accurate in this target domain.

The images taken from sketches are basically composed by shapes, and because of that, have lots of high frequency components. Moreover, all the texture of the image comes from the texture of the paper where the sketch was drawn. Because of those two factors, it's reasonable to assume that the feature detectors of our baseline DCNNs are not suitable for VIS-Sketch task. However, in practice, we observe the opposite. The Inception-ResNet v1 CNN presented an average rank one recognition rate of 81.48% in the CUFS database. These experiments show that such feature detectors are very robust, even though the recognition rates are lower than the state-of-the-art.

The most challenging task seems to be the VIS-Thermal domain. For this one, the best CNN (Inception-ResNet v1-rgb) achieved an average recognition rate of only 27.68%.

In this section we presented an overview of Face Recognition using DCNNs and we analysed the effectiveness of six different face models trained with VIS face images in the HFR task (covering three different image domains). It was possible to observe that despite those new image domains were not used to train the DCNN, their feature detectors achieved recognition rates way above a random guess. For some of them, it was possible to achieve recognition rates above 80%. With those experiments we argue that some set of feature detectors suitable for VIS ($\mathcal{D}^s$) are also suitable for different spectral domains ($\mathcal{D}^t$).

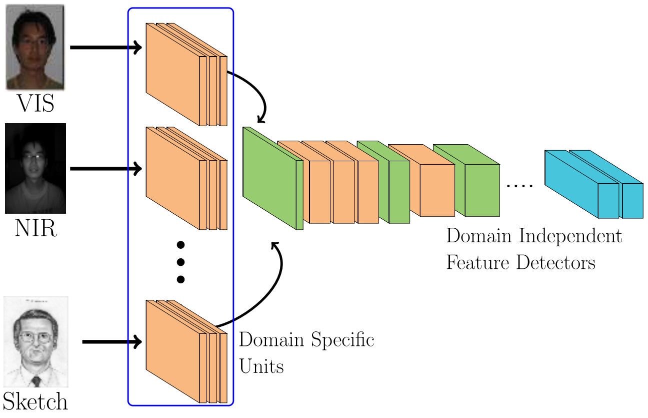

Fig. 3. Domain specific units - general schematic.

IV. Proposed Approach

Many researchers pointed out that DCNNs progressively compute more powerful feature detectors as depth increases [35]. Yosinski *et al.* [39] and Li *et al.* [48] demonstrated that feature detectors that are closer to the input signal (called low level features) are base features that resemble Gabor features, color blobs, edge detectors, etc. On the other hand, features that are closer to the end of the neural network (called high level features) are considered to be more task specific and carry more discriminative power.

In the last section we observed that the feature detectors from $\mathcal{D}^s$ (VIS) have some discriminative power over all three target domains we have tested; with VIS-NIR being the "easiest" ones and the VIS-Thermal being the most challenging ones. With such experimental observations, we can draw the following hypothesis:

Hypothesis 1: Given $X_s = \{x_1, x_2, \ldots, x_n\}$ and $X_t = \{x_1, x_2, \ldots, x_n\}$ being a set of samples from $\mathcal{D}^s$ and $\mathcal{D}^t$, respectively, with their corresponding shared set of labels $Y = \{y_1, y_2, \ldots, y_n\}$ and Θ being all set of DCNN feature detectors from $\mathcal{D}^s$ (already learnt), there are two consecutive subsets: one that is domain **dependent**, θ_t, and one that is domain **independent**, θ_s, where $P(Y|X_s, \Theta) = P(Y|X_t, [\theta_s, \theta_t])$. Such θ_t, that can be learnt via back-propagation, is so called **Domain Specific Units**

A possible assumption one can make is that θ_t is part of the set of low level features, directly connected to the input signal. In this paper we test this assumption. Figure 3 presents

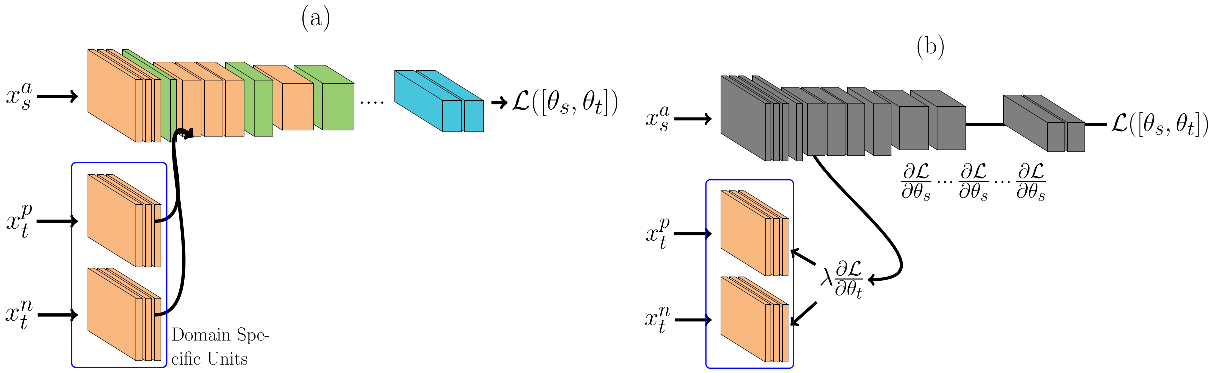
(a)
x_s^a
x_t^p
x_t^n
$\mathcal{L}([\theta_s, \theta_t])$
Domain Specific Units
(b)
x_s^a
x_t^p
x_t^n
$\mathcal{L}([\theta_s, \theta_t])$
$\frac{\partial \mathcal{L}}{\partial \theta_s} \cdots \frac{\partial \mathcal{L}}{\partial \theta_s} \cdots \frac{\partial \mathcal{L}}{\partial \theta_s}$
$\lambda \frac{\partial \mathcal{L}}{\partial \theta_t}$

Fig. 4. Domain specific units learnt with triplet neural networks given a triplet of samples: x_s^a from $\mathcal{D}_s$, and x_t^p and x_t^n from $\mathcal{D}_t$. (a) Forward pass behaviour. (b) Backward pass behaviour.

a general schematic of our proposed approach. It is possible to observe that each image domain has its own specific set of feature detectors (low level features) and they share the same face space (high level features) that was previously learnt using VIS.

Our approach consists in learning θ_t, for each target domain, jointly with the DCNN from the source domain. In order to jointly learn θ_t with D_s we propose two different architectural arrangements described in the Figures 2 and 4.

In the architecture described in Figure 2, θ_t is learnt using Siamese Neural Networks [40]. During the forward pass, Figure 2 (a), a pair of face images, one for each domain (either sharing the same identity or not), is passed through the DCNN. The image from the source domain is passed through the main network [the one at the top in Figure 2 (a)] and the image from the target domain is passed first to its domain specific set of feature detectors and then amended to the main network. During the backward pass, Figure 2 (b), errors are backpropagated only for θ^t. With such structure only a small subset of feature detectors are learnt, reducing the capacity of the joint model. The loss $\mathcal{L}$ is defined as:

$$\mathcal{L}(\Theta) = 0.5\Big[(1-Y)D(x_s, x_t) + Y\max(0, m - D(x_s, x_t))\Big], \tag{2}$$

where m is the contrastive margin, Y is the label (1 when x_s and x_t belong to the same subject and 0 otherwise) and D is defined as:

$$D(x_s, x_t) = ||\phi(x_s) - \phi(x_t)||_2^2, \tag{3}$$

where ϕ are the embeddings from the jointly trained DCNN.

In the architecture described in Figure 4, θ_t is learnt using Triplet Neural Networks [37]. During the forward pass, Figure 4 (a), a triplet of face images are presented as inputs to the network. In its figure, x_s^a consist of face images sensed in the source domain, and x_t^p and x_t^n are images sensed in the target domain, where x_s^a and x_t^p are from the same identity and x_s^a and x_t^n are from different identities. As before, face images from the source domain are passed through the main network [the one at the top in Figure 4 (a)] in and face images from the target domain are passed first to its domain specific set of feature detectors and then amended to the main network. During the backward pass, Figure 4 (b), errors are backpropagated only for θ^t, that is shared between the inputs x_t^p and x_t^n. With such structure only a small subset of features are learnt, reducing the capacity of the model. The loss $\mathcal{L}$ is defined as:

$$\mathcal{L}(\theta) = ||\phi(x_s^a) - \phi(x_t^p)||_2^2 - ||\phi(x_s^a) - \phi(x_t^n)||_2^2 + \lambda, \tag{4}$$

where λ is the triplet margin and ϕ are the embeddings from the DCNN.

Algorithm 1 presents a generic pseudo-code of the training procedure that is independent of architectural arrangements. It is worth noting that only the Domain Specific Units (θ_t) are updated.

Algorithm 1 Training Strategy Given a Pretrained DCNN Θ_s, Loss Function $\mathcal{L}$ and the Number of Layers to be Adapted n_layers. θ_t Is Split Between the Convolutional Kernels W and the Biases β

```
Data: $\Theta_s$, $\mathcal{L}$, $n\_layers$
Result: $\theta_t$
$\theta_t = \Theta_s[1 : n\_layers]$ ;   // Domain Spec. Units
$\theta_s = \Theta_s[n\_layers :]$ ;   // Domain Indep. Units
while $has\_data$ do
    batch = get_batch();
    $[\frac{\partial \mathcal{L}}{\partial \theta_s}, \frac{\partial \mathcal{L}}{\partial \theta_t}]$ = forward_backward(batch, $\theta_s$, $\theta_t$, $\mathcal{L}$);
    $\theta_t[\beta] = \theta_t[\beta] + \lambda \frac{\partial \mathcal{L}}{\partial \theta_t}[\beta]$;
    if $adapt\_kernels$ then
        $\theta_t[W] = \theta_t[W] + \lambda \frac{\partial \mathcal{L}}{\partial \theta_t}[W]$
    end
end
```

For our experiments, one DCNN is chosen for D_s: the Inception Resnet v2. Such network presented one of the highest recognition rates under different image domains. Since our target domains are one channel only, we selected the gray scaled version of it. Details of such architecture is presented in the Supplementary Material.

Our task is to find the set of low level feature detectors, θ_t, that maximizes the recognition rates for each image domain.

In order to find such set, we exhaustively try, layer by layer (increasing the DCNN depth), adapting both Siamese and Triplet Networks. Five possible θ_t sets are analysed and they are called $\theta_{t[1-1]}$, $\theta_{t[1-2]}$, $\theta_{t[1-4]}$, $\theta_{t[1-5]}$ and $\theta_{t[1-6]}$. A full description of which layers compose θ_t is presented in the Supplementary material of the paper. The Inception Resnet v2 architecture batch normalize [41] the forward signal for every layer. For convolutions, such batch normalization step is defined, for each layer i, as the following:

$$h(x) = \beta_i + \frac{g(W_i * x) + \mu_i}{\sigma_i}, \tag{5}$$

where β is the batch normalization offset (role of the biases), W are the convolutional kernels, g is the non-linear function applied to the convolution (ReLU activation), μ is the accumulated mean of the batch and σ is the accumulated standard deviation of the batch.

In the Equation 5, two variables are updated via back-propagation, the values of the kernel (W) and the offset (β). With these two variables, two possible scenarios for $\theta_{t[1-n]}$ are defined. In the first scenario, we consider that $\theta_{t[1-n]}$ is composed by the set of batch normalization offsets (β) only and the convolutional kernels W are shared between $\mathcal{D}_s$ and $\mathcal{D}_t$. We may hypothesize that, since the target object that we are trying to model has the same structure among domains (frontal faces with neutral expression most of the time), the feature detectors for $\mathcal{D}_s$ and $\mathcal{D}_t$, encoded in W, are the same and just offsets need to be domain specific. In this work such models are represented as $\theta_{t[1-n]}(\beta)$. In the second scenario, both W and β are made domain specific (updated via back-propagation) and they are represented as $\theta_{t[1-n]}(\beta + W)$.

V. Results and Discussion

In this section we discuss the results of our proposed approach under the four different image databases covering three different domains. As mentioned in Section III, the input of Inception Resnet v2 is 160 × 160 × 1 (width, height and number channels). All the DCNNs are trained using Stochastic Gradient Descend for 100 epochs. The learning rate update strategy and the batch size are the same as in [51]. The learning rate is 0.1 for 75 epochs, then it goes to 0.01 for 15 epochs and finally runs for 10 epochs at 0.001. The size of the batch is 90 (pairs or triplets). More implementation details and how this DCNN was trained for VIS can be found in the supplementary material. Furthermore, the source code of this paper is available.[10] Once those DCNNs are trained, the same comparison procedure applied in Section IV (Equation 1) is applied.

The following subsections describe the experiments for each database. For the sake of brevity, we present the Cumulative Match Characteristic plots (CMC) for the best performed system.

A. CUHK CUFS (VIS-Sketch)

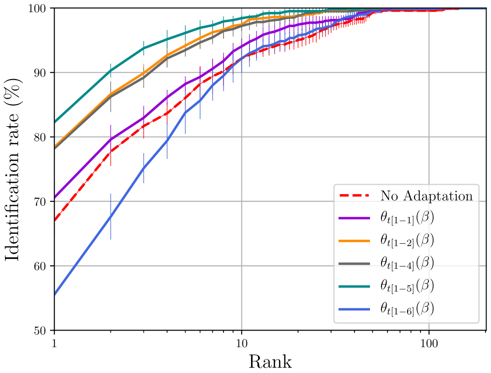

Fig. 5. CUFS - Average CMC curves (with error bars) for the adaptation of biases only - Siamese networks with Inception Resnet v2.

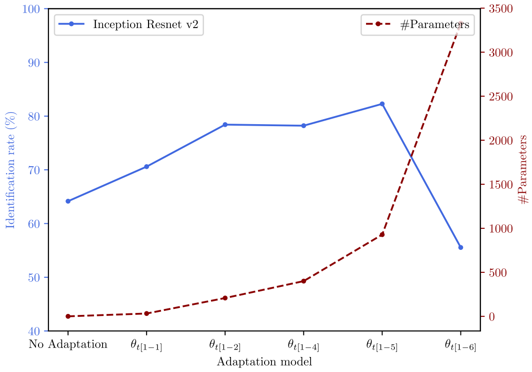

Fig. 6. Average rank one recognition rate vs number of parameters learnt.

Figure 5 presents the CMC curves with adaptation of the biases only for the Inception Resnet v2 using the Siamese Networks. Such DCNN, with no adaptation, has an average rank one recognition rate of 67.03%. Adapting only the biases (β in Equation 5) of the first layer ($\theta_{t[1-1]}(\beta)$ in the plots) it was possible to get this benchmark improved to ≈ 70%. The biases adaptation for $\theta_{t[1-2]}$ and $\theta_{t[1-4]}$ improved the average rank one recognition rate to ≈ 78% for both. Experiments with $\theta_{t[1-5]}$ get its best average rank one recognition rate with 82.2%. For $\theta_{t[1-6]}$ the average rank one recognition rate drops drastically to ≈ 55%. A possible overfitting can be suggested for $\theta_{t[1-6]}$. Figure 6 shows the plot of the average rank one recognition rates and the number of parameters learnt as a function of $\theta_{t[1-n]}$ for the Siamese net using the Inception Resnet v2 as a basis. We can observe the drop, in terms of average rank one recognition rate, from $\theta_{t[1-5]}$ to $\theta_{t[1-6]}$ when the number of parameters learnt drastically grows (from 928 to 3328). More details about the number of parameters for all θ_t can be checked in Table VI.

We also observed the same trends using Triplet Networks as a base trainer. Adapting $\theta_{t[1-1]}$ the average rank one recognition rates is improved ≈ 72%. For $\theta_{t[1-2]}$ and $\theta_{t[1-4]}$ the improvement are ≈ 78% and ≈ 79% respectively. For $\theta_{t[1-5]}$ the average rank one recognition rate improved to 82.9% and then drastically drops to ≈ 59% for $\theta_{t[1-6]}$.

With this set of experiments it was possible to observe that the adaptation of batch normalization offsets only improve the recognition rates. This could naturally imply that $\mathcal{D}_s$ and $\mathcal{D}_t$ for VIS-Sketch share the same set of feature detectors and the

[10] https://gitlab.idiap.ch/bob/bob.bio.face_ongoing

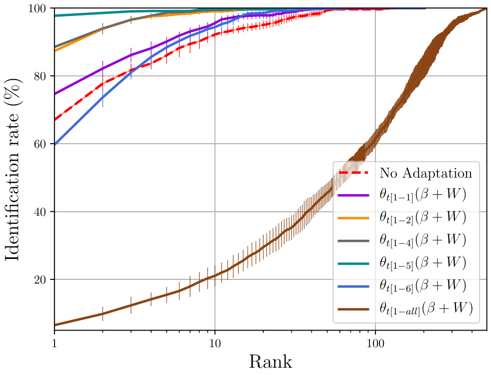

Fig. 7. CUFS - Average CMC curves (with error bars) for the adaptation of kernel and biases - Siamese networks with Inception Resnet v2.

difference is a matter of bias shifting. In order to investigate if there are domain specific feature detectors, the next set of experiments we perform the same experimental procedure, but instead of adapting only β we do adapt β and W (Equation 5).

Figure 7 presents the CMC curves with adaptation of convolutional kernels and biases for the Inception Resnet v2 using the Siamese Networks. Such DCNN, with no adaptation, presents an average rank one recognition rate of 67.03%. Adapting both, biases and kernels (β and W in Equation 5), of the first layer ($\theta_{t[1-1]}(\beta + W)$ in the plots) it was possible to improve this benchmark to $\approx$ 74%. The adaptation for $\theta_{t[1-2]}$ and $\theta_{t[1-4]}$ improved the average rank one recognition rates to $\approx$ 87% and $\approx$ 89% respectively. Experiments with $\theta_{t[1-5]}$ get its best average rank one recognition rate with 97.7%. For $\theta_{t[1-6]}$ the average rank one recognition rate drops drastically to $\approx$ 60%. The same aforementioned overfitting can be suggested for $\theta_{t[1-6]}$.

As before, with the Siamese Networks, we also observed the same trend using Triplet Networks as training strategy. Adapting $\theta_{t[1-1]}$ the average rank one recognition rates improved to $\approx$ 75%. For $\theta_{t[1-2]}$ and $\theta_{t[1-4]}$ the improvements are $\approx$ 78% and $\approx$ 81% respectively. For $\theta_{t[1-5]}$ the average rank one recognition rate improved to 81.5% and then drastically drops to $\approx$ 51% for $\theta_{t[1-6]}$. For the sake of completeness, in this experiment we also adapted all layers. In the Figure 7 this is represented as $\theta_{t[1-all]}$. It is possible to observe a severe degradation in terms of rank one recognition rate. Compared with $\theta_{t[1-6]}$ it dropped from $\approx$ 51% to $\approx$ 6.5%, confirming our assumption about overfitting.

With these set of experiments it was possible to observe that, despite the adaptation of only the $\beta's$ increase the recognition rates, the joint adaptation of β and W increases even more this benchmark. We can suggest that there are domain specific feature detectors and such feature detectors need to be taken in to account for the Heterogeneous Face Recognition task.

From the experiments above, the best average rank one recognition rate is achieved with using Siamese Networks as a base trainer. The model $\theta_{t[1-5]}$ achieved an average recognition rate of 97.72%(0.6).

Table II shows the average rank one recognition rate comparing different configuration of our proposed approach (the best ones for each setup) with with five reference systems from the literature. The first two are from [1] (P-RS and FaceVACS). Unfortunately, the source code of those approaches are not available for reproducibility. Hence, for these we only report the performance. For the other three references, the source code was made available and it can be checked in the corresponding publications.

TABLE II

CUHK-CUFS - Average Rank One Recognition Rate

Method	**Mean (Std. Dev.)**	**Info**
P-RS as in [1] (section 7.2)	99.% (not informed)	Paper baselines
Face VACS in [1] (section 7.2)	89.6% (not informed)	
ISV in [27]	96.9% (1.3)	Reproducible baselines
GFK in [28]	93.3% (1.4)	
MLBPs + DoG features in [17]	62.3% (3.8)	
Siam. Incep. Res. v2 $\theta_{t[1-5]}$	82.2% (1.7)	Adapt β
Trip. Incep. Res. v2 $\theta_{t[1-5]}$	82.9% (2.3)	
Siam. Incep. Res. v2 $\theta_{t[1-5]}$	**97.7% (0.6)**	Adapt $\beta + W$
Trip. Incep. Res. v2 $\theta_{t[1-5]}$	81.5% (2.9)	

Comparing with P-RS, in terms of average rank one, the difference is $\approx$ 1%, which represents $\approx$ 2 miss classifications. The HFR approach implemented in P-RS is composed by a score a fusion of 180 different face recognition systems (6 systems with 30 bags each). In the approach each face image is geometric normalized with 250×200 pixels keeping an inter-pupil distance of 75 pixels. Three preprocessing strategies are applied: Difference of Gaussian Filter (DoG) [18], Center Surround Divisive Normalization (CSDN) [42] and a Gaussian Filter. For each preprocessed image two different features are extracted: MLBP features [19] (uniform pattern with 59 bins) with 4 different radius (1, 3, 5, 7) and SIFT features [50] (128 features). Each of these features are extracted in patches of 32×32 pixels with a patch overlap of 16×16. Summing up, all these features combined with the preprocessing mechanisms leads to more than 40,000 feature descriptors. Compared with our approach, which is composed by only one system instead of 180 complex systems (several bags, different types of feature, different image processing algorithms), the difference of 2 miss classifications doesn't look an enormous gap. Furthermore, our proposed approach performs better than the ISV ($\approx$ 97%) and the GFK ($\approx$ 93%).

B. CASIA (VIS-NIR)

Figure 8 presents the CMC curves with adaptation of the biases only for the Inception Resnet v2 using the Siamese Networks. Such DCNN, with no adaptation, presents an average rank one recognition rate of 73.80%. Adapting only the biases (β in Equation 5) of the first layer ($\theta_{t[1-1]}(\beta)$ in the plots) it was possible to improve this benchmark to $\approx$ 77%. The biases adaptation for $\theta_{t[1-2]}$ and $\theta_{t[1-4]}$ improved the average rank one recognition rates to $\approx$ 83% and $\approx$ 86% respectively. Experiments with $\theta_{t[1-5]}$ get its best average rank one recognition rate with 88.5%. For $\theta_{t[1-6]}$ the average rank one recognition rate drops drastically to $\approx$ 35%. The same overfitting hypothesis suggested before can be applied for $\theta_{t[1-6]}$.

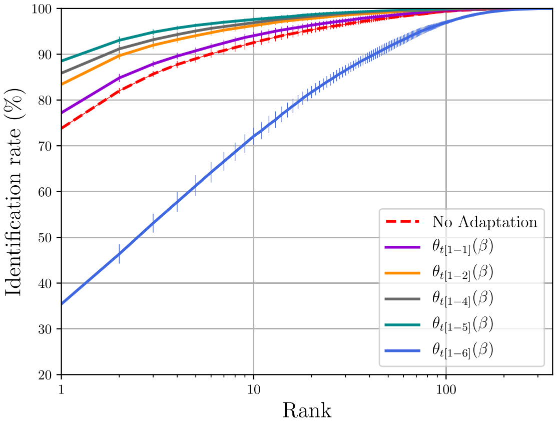

Fig. 8. CASIA - Average CMC curves (with error bars) for the adaptation of biases only - Siamese networks with Inception Resnet v2.

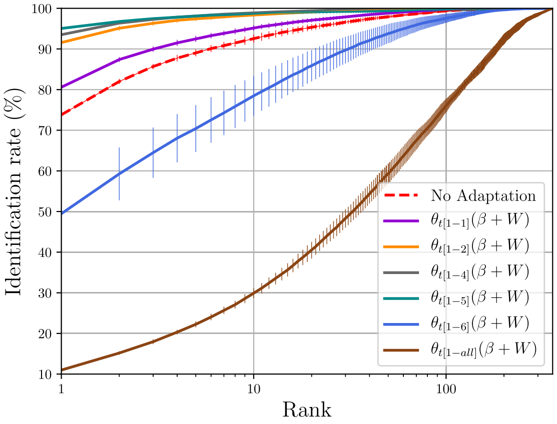

Fig. 9. CASIA - Average CMC curves (with error bars) for the adaptation of kernel and biases - Siamese networks with Inception Resnet v2.

TABLE III

CASIA - Average Rank One Recognition Rate and Verification Rate at False Acceptance Rate of 0.1%

Method	**Rank-1(Std. Dev.)**	**VR@FAR=0.1**	**Info**
Baseline [7] (Table 2)	23.70% (1.89)	-	Paper lines
CDFL in [26] (Table I)	71.5% (1.4)	55.10%	
DSIFT in [44] (Table II)	73.28% (1.10)	-	
FaceVACS in [44] (Table I)	58.56% (1.19)	-	
Gabor+RBM [26] (Table I)	86.1% (0.1)	81.29%(1.82)	
TRIVET in [43]	95.74% (0.5)	91.03% (1.3)	
IDR in [54]	95.82% (0.76)	94.03% (1.06)	
CDL in [55]	98.62% (0.2)	98.32% (0.05)	
WCNN in [56]	98.7% (0.3)	98.4% (0.4)	
LBPs+DoG feat. in [17]	70.33% (1.2)	74.1% (0.5)	Reproducible baselines
ISV in [27]	72.39% (1.35)	72.4% (0.5)	
GFK in [28]	26.98% (0.9)	32.9% (1.)	
Siam. Incep. Res.v2 $\theta_{t[1-5]}$	88.5% (1.1)	87.3% (0.5)	Adapt β
Trip. Incep. Res.v2 $\theta_{t[1-1]}$	73.8% (2.0)	75.9% (0.3)	
Siam. Incep. Res.v2 $\theta_{t[1-5]}$	**96.3% (0.4)**	**98.4%(0.12)**	Adapt $\beta + W$
Trip. Incep. Res.v2 $\theta_{t[1-5]}$	90.1% (2.9)	97.8%(0.2)	

We also observed the same trends using Triplet Networks as a base trainer. Adapting $\theta_{t[1-1]}$ the average rank one recognition rates is improved $\approx 74\%$. For $\theta_{t[1-2]}$ and $\theta_{t[1-4]}$ the improvement are $\approx 74\%$ and $\approx 75\%$ respectively. For $\theta_{t[1-5]}$ the average rank one recognition slightly drops to $\approx 68\%$ and then drastically drops to $\approx 15\%$ for $\theta_{t[1-6]}$.

The same trend observed in Section V-A, with VIS-Sketeches, was observed in VIS-NIR. The adaptation of the batch normalization offsets only improve the recognition rates. In the next set of experiments we investigate if there are domain specific feature detectors by adapting β and W (Equation 5)

Figure 9 presents the CMC curves with adaptation of convolutional kernels and biases for the Inception Resnet v2 using Siamese Networks. Such DCNN, with no adaptation, presents an average rank one recognition rate of 73.8%. Adapting both, biases and kernels (β and W in Equation 5), of the first layer ($\theta_{t[1-1]}$ in the plots) it was possible to get this benchmark improved to $\approx 80\%$. The adaptation for $\theta_{t[1-2]}$ and $\theta_{t[1-4]}$ improved the average rank one recognition rates to $\approx 91\%$ and $\approx 93\%$ respectively. Experiments with $\theta_{t[1-5]}$ get its best average rank one recognition rate with 96.3%. For $\theta_{t[1-6]}$ the average rank one recognition rate drops drastically to $\approx 49\%$. A possible overfitting can be suggested for $\theta_{t[1-6]}$. In this experiment we also adapted all layers. In the Figure 9 this is represented as $\theta_{t[1-all]}$. It is possible to observe a severe degradation in terms of rank one recognition rate. Compared with $\theta_{t[1-6]}$ it dropped from $\approx 49\%$ to $\approx 10.9\%$ confirming our assumption about overfitting for another image modality.

As before, with Siamese Networks, we also observed the same trends using Triplet Networks as training strategy. Adapting $\theta_{t[1-1]}$ the average rank one recognition rates are improved to $\approx 76\%$. For $\theta_{t[1-2]}$ and $\theta_{t[1-4]}$ the improvements are $\approx 79\%$ and $\approx 89\%$ respectively. For Inception Resnet v2 the average rank one recognition rate improved to 90.1% for $\theta_{t[1-5]}$ and it drastically drops to $\approx 51\%$ for $\theta_{t[1-6]}$.

With this set of experiments it was possible to observe that, despite the adaptation of only βs increase the recognition rates, the joint adaptation of β and W slightly increases such figure of merit. We can argue that there are domain specific feature detectors and such feature detectors need to be taken in to account for the HFR task.

Table III shows the average rank one recognition rate comparing different configurations of our proposed approach with twelve reference systems from the literature. We also report the Verification Rate at False Acceptance Rate of 0.1%, since this is a common evaluation metric used in the literature for this particular database. Unfortunately, the source code for the first nine approaches is not available for reproducibility. Hence, for these we only report the performance. For the other three references, the source code is made available and it can be checked in the corresponding publications.

Using the average rank one recognition rate as reference (closed-set identification task), different setups of our proposed approach are substantially better than the most of the state-of-the-art results. Our best setup (96.3% with the model $\theta_{t[1-5](\beta+W)}$ trained with Siamese Neural Networks and Inception Resnet v2), presents a slightly better recognition performance compared with the TRIVET system in [43] (95.74%) and the IDR system in [54] (95.85%). The CDL system in [55] and WCNN system in [56] presented slightly better average rank one recognition rates; respectively 97.8% and 98.8%. With respect to the Verification Rate at False

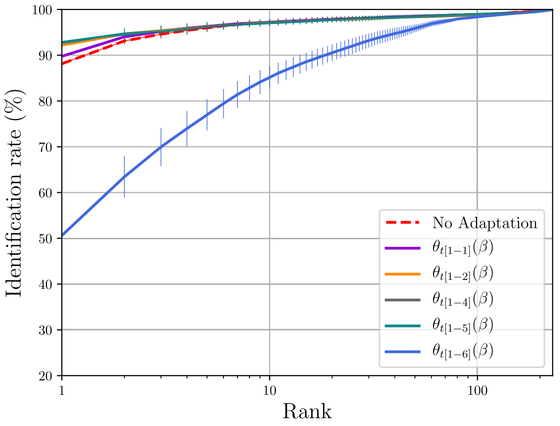

Fig. 10. NIVL - Average CMC curves (with error bars) for the adaptation of biases only - Siamese networks with Inception Resnet v2.

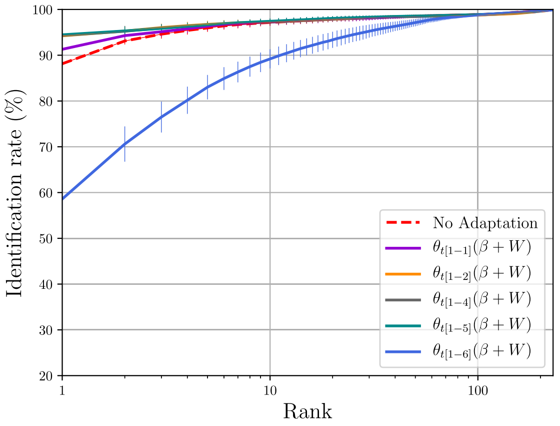

Fig. 11. NIVL - Average CMC curves (with error bars) for the adaptation of kernel and biases - Siamese networks with Inception Resnet v2.

Acceptance Rate of 0.1% (verification task), our best setup (98.4% with the model $\theta_{t[1-5](\beta+W)}$ trained with Siamese Neural Networks and Inception Resnet v2) presents equivalent recognition performance compared with CDL system in [55] and WCNN system in [56]; respectively 98.6% and 98.7%.

C. NIVL (VIS-NIR)

Figure 10 presents the CMC curves with adaptation of the biases only for the Inception Resnet v2 using the Siamese Networks. Such DCNN, with no adaptation, has an average rank one recognition rate of 88.14%. Adapting only the biases (β in Equation 5) of the first layer ($\theta_{t[1-1]}(\beta)$ in the plots) it was possible to improve this benchmark to $\approx$ 89%. The biases adaptation for $\theta_{t[1-2]}$ improved the average rank one recognition to $\approx$ 92%. Adapting $\theta_{t[1-4]}$ and $\theta_{t[1-5]}$ improved this benchmark to 92.7% and 92.8% respectively. For $\theta_{t[1-6]}$ the average rank one recognition rate drops drastically to $\approx$ 51%. The same overfitting hypothesis suggested before can be verified for $\theta_{t[1-6]}$.

Training with Triplet Networks the same trends are observed. Adapting $\theta_{t[1-1]}$ and $\theta_{t[1-2]}$ the average rank one recognition rates of both get improved to $\approx$ 91%. For $\theta_{t[1-4]}$ the average rank one recognition rate improved to $\approx$ 92%. Then, slightly decreased to $\approx$ 90% for $\theta_{t[1-5]}$ and it drastically drops to $\approx$ 30% for $\theta_{t[1-6]}$.

The same trends observed in Section V-A and V-B was observed for this database. The adaptation of the batch normalization offsets only do improve the recognition rates. In the next set of experiments we investigate if there are domain specific feature detectors by adapting β and W (Equation 5).

Figure 11 presents the CMC curves with adaptation of convolutional kernels and biases for the Inception Resnet v2 using the Siamese Networks. Such DCNN, with no adaptation, has an average rank one recognition rate of 88.14%. Adapting both, biases and kernels (β and W in Equation 5), of the first layer ($\theta_{t[1-1]}(\beta + W)$ in the plots) it was possible to get this benchmark improved to $\approx$ 91%. The adaptation for $\theta_{t[1-2]}$ and $\theta_{t[1-4]}$ improved the average rank one recognition rates to $\approx$ 94% and $\approx$ 94% respectively. Experiments with $\theta_{t[1-5]}$ get its best average rank one recognition rate with 94.5%. For $\theta_{t[1-6]}$ the average rank one recognition rate drops drastically to $\approx$ 59%. A possible overfitting can be suggested for $\theta_{t[1-6]}$.

TABLE IV

NIVL - Average Rank One Recognition Rate

Method	Mean (Std. Dev.)	Info
LBPs + DoG features in [17]	19.8% (1.3)	Reproducible baselines
ISV in [27]	89.3% (0.8)	
GFK in [28]	17.2% (1.7)	
Siam. Inception Resnet v2 $\theta_{t[1-5]}$	92.8%(1.2)	Adapt β
Triplet Inception Resnet v2 $\theta_{t[1-4]}$	91.9%(1.8)	
Siam. Inception Resnet v2 $\theta_{t[1-5]}$	**94.5%(1.2)**	Adapt $\beta + W$
Triplet Inception Resnet v2 $\theta_{t[1-5]}$	92.2%(1.4)	

As before, with Siamese Networks, we also observed the same trends using Triplet Networks as training strategy. Adapting $\theta_{t[1-1]}$ and $\theta_{t[1-2]}$ the average rank one recognition rates of both get improved to $\approx$ 90%. Then, the average rank one recognition rates are improved to $\approx$ 92% for $\theta_{t[1-4]}$ and $\theta_{t[1-5]}$ and it drastically drops to $\approx$ 54% for $\theta_{t[1-6]}$.

With this set of experiments it was possible to observe that, despite the adaptation of only βs increase the recognition rates, the joint adaptation of β and W slightly increased such figure of merit. We can suggest that there are domain specific feature detectors and such feature detectors need to be taken in to account for the HFR task.

Table IV shows the average rank one recognition rate comparing different configurations of our proposed approach with three reference systems from the literature whose source code is available. As mentioned in Section II-E.3, there is no official evaluation protocol for this database. In terms of average rank one recognition rate our proposed approach is substantially better than the rest of the state-of-the-art results. The best setup is the model $\theta_{t[1-4]}$ trained with Siamese Neural Networks using the Inception Resnet v2 as a basis and achieved a recognition rate of 94.9%.

D. Pola Thermal (VIS-Thermal)

Figure 12 presents the CMC curves with adaptation of the biases only for the Inception Resnet v2 using the Siamese Networks. Such DCNN, with no adaptation, has an average

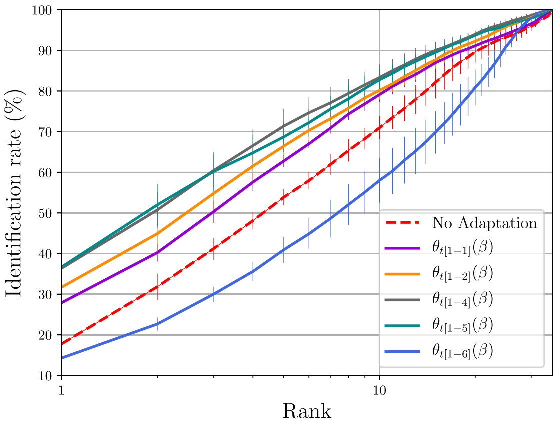

Fig. 12. Pola Thermal - Average CMC curves (with error bars) for the adaptation of biases only - Siamese networks with Inception Resnet v2.

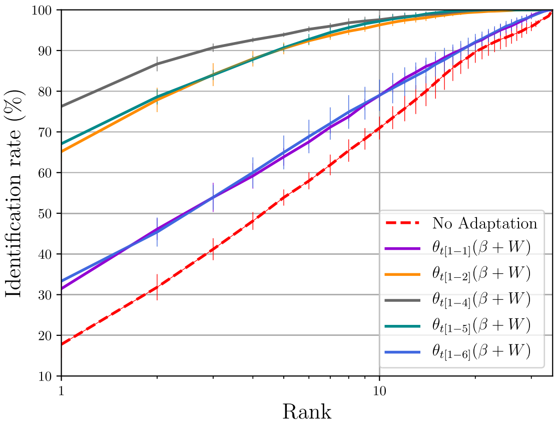

Fig. 13. Pola Thermal - Average CMC curves (with error bars) for the adaptation of kernel and biases - Siamese networks with Inception Resnet v2.

rank one recognition rate of 17.8 %. Adapting only the biases (β in Equation 5) of the first layer ($\theta_{t[1-1]}(\beta)$ in the plots) it was possible to improve this benchmark to $\approx$ 28%. The biases adaptation for $\theta_{t[1-2]}$ achieved an average rank one recognition to $\approx$ 31%. Adapting $\theta_{t[1-4]}$ and $\theta_{t[1-5]}$ the average rank one recognition rates increased to $\approx$ 35% and 36.7% respectively. For $\theta_{t[1-6]}$ the average rank one recognition rate drops drastically to $\approx$ 14%.

Training with Triplet Networks the same trends are observed. Adapting $\theta_{t[1-1]}$ and $\theta_{t[1-2]}$ the average rank one recognition rate get improved to $\approx$ 26% and $\approx$ 34% respectively. For $\theta_{t[1-4]}$ the average rank one recognition rate improved to $\approx$ 39%. For Inception Resnet v2 the average rank one recognition rate increased to 41.3% for $\theta_{t[1-5]}$ and it drastically drops to $\approx$ 16% for $\theta_{t[1-6]}$.

The same trends observed in the previous subsections were observed in this database. The adaptation of the batch normalization offsets only improve the recognition rates. In the next set of experiments we investigate if there are domain specific feature detectors by adapting β and W (Equation 5)

Figure 13 presents the CMC curves with adaptation of convolutional kernels and biases for the Inception Resnet v2 using Siamese Networks. Such DCNN, with no adaptation, presents an average rank one recognition rate of 17.7%. Adapting both, biases and kernels (β and W in Equation 5), of the first layer ($\theta_{t[1-1]}(\beta + W)$ in the plots) it was possible to get this benchmark improved to $\approx$ 31%. The adaptation for $\theta_{t[1-2]}$ and $\theta_{t[1-4]}$ improved the average rank one recognition rate to $\approx$ 65% and 76.3% (its best) respectively. With $\theta_{t[1-5]}$ the average rank one recognition rate drops to $\approx$ 67%. For $\theta_{t[1-6]}$ the average rank one recognition rate drops drastically to $\approx$ 33%.

TABLE V

POLA THERMAL - AVERAGE RANK ONE RECOGNITION RATE

Method	Mean (Std. Dev.)	Info
DPM in [14] (Table II)	75.31 % (-)	Paper Baseines
CpNN in [14] (Table II)	78.72 % (-)	
PLS in [14] (Table II)	53.05% (-)	
LBPs + DoG features in [17]	36.8% (3.5)	Reproducible baselines
ISV in [27]	23.5% (1.1)	
GFK in [28]	34.1% (2.9)	
Siam. Incep. Resnet v2 $\theta_{t[1-5]}$	36.7% (5.3)	Adapt β
Triplet Incep. Resnet v2 $\theta_{t[1-5]}$	41.3% (4.6)	
Siam. Incep. Resnet v2 $\theta_{t[1-4]}$	**76.3% (2.1)**	Adapt $\beta + W$
Triplet Incep. Resnet v2 $\theta_{t[1-5]}$	50.9% (2.0)	

As before, with Siamese Networks, we also observed the same trends using Triplet Networks as training strategy. Adapting $\theta_{t[1-1]}$ and $\theta_{t[1-2]}$ the average rank one recognition rate improved to $\approx$ 28% and $\approx$ 42% respectively. For $\theta_{t[1-4]}$, the average rank one recognition rate improved to $\approx$ 48% and to $\approx$ 51% for $\theta_{t[1-5]}$. Finally for $\theta_{t[1-6]}$ the average rank one recognition rates drastically drops to $\approx$ 27%.

With this set of experiments it was possible to observe that, despite the adaptation of only the βs increase the recognition rates, the joint adaptation of β and W drastically increased even more such figure of merit. We can suggest that there are domain specific feature detectors and such feature detectors need to be taken in to account for the HFR task.

Table V shows the average rank one recognition rate comparing different configurations of our proposed approach with six reference systems from the literature. Unfortunately, the source code for the first three approaches is not available for reproducibility. Hence, for these we only report the recognition rates. For the other three references, the source code was made available alongside the corresponding articles.

Our best setup (Inception Resnet v2 model $\theta_{t[1-4]}$ trained with Siamese Networks) presented an average rank one recognition rate of 76.3%. Such recognition rate surpass all the **Reproducible Baselines** average rank one recognition rates. For the **Paper Baselines**, the results are competitive when compared with DPM ($\approx$ 75%) and CpNN ($\approx$ 78%), although the evaluation protocols are not exactly same.

E. Discussion

Compared to a DCNN with no adaptation, our approach using Domain Specific Units systematically improved the HFR recognition rates for all tested image domains. This reinforces **Hypothesis 1** where we argue that for a given set of DCNN feature detectors Θ we can split them in two consecutive subsets, one that is **domain dependent**, θ_t, and one that is

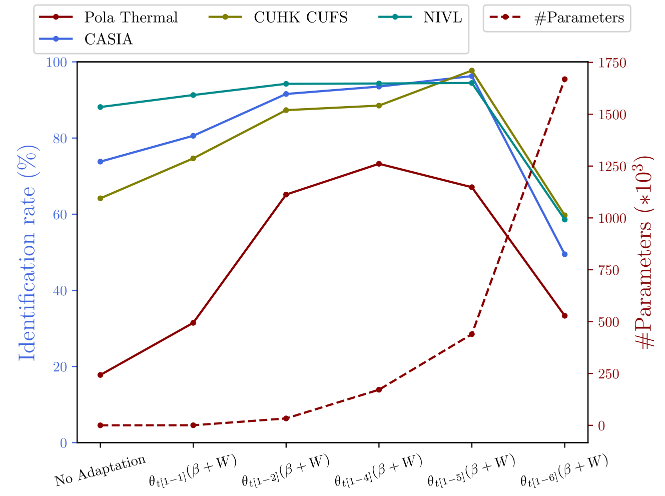

Fig. 14. Average rank one recognition rate vs number of parameters learnt using Inception Resnet v2 as a base DCNN and Siamese Neural Networks as a training method.

domain independent, θ_s. Moreover, such improvements were observed independently of training method (Siamese or Triplet Neural Networks) and the way that the Domain Specific Units were encoded [$\theta_{t[1-n]}(\beta)$ or $\theta_{t[1-n]}(\beta+W)$].

With respect to the training method, we observed that the pair Siamese Neural Networks and Inception Resnet v2 presented the highest average rank one recognition rates overall. Hence, for following analysis we consider this particular setup.

By incrementally applying our proposed approach layer by layer ($\theta_{t[1-n]}$), we could observe improvements, in terms of rank one recognition rate, until certain point. Overall, such improvements could be observed until the layer set $\theta_{t[1-5]}$. In this case, the recognition rate started to decrease concomitantly when the number of free parameters started to exponentially grow. Figure 14 presents the average rank one recognition rates for all databases as function of $\theta_{t[1-n]}$ in parallel with the number of free parameters for each one (dashed line). It is possible to observe that the average rank one recognition rates for all databases drastically drops in $\theta_{t[1-5]}$ when the number of free parameters is increased by a factor of ≈ 4. Such models are possibly overfitted. Table VI presents the number of free parameters that need to be learnt for each $\theta_{t[1-n]}$.

Two configurations were considered for $\theta_{t[1-n]}$. Either $\theta_{t[1-n]}$ was composed by adaptation of the offsets ($\theta_{t[1-n]}(\beta)$) or was composed by the adaptation of the pair: convolutional kernels W and their corresponding offsets ($\theta_{t[1-n]}(\beta+W)$). Substantial improvements were observed adapting only $\theta_{t[1-n]}(\beta)$ via back-propagation for all image domains. For the VIS-NIR domain, which is our "less challenging" task, the adaption of $\theta_{t[1-5]}(\beta)$ provided high recognition rates; it improved from $\approx 73\%$ to $\approx 89\%$ (see Table III) and from $\approx 88\%$ to $\approx 93\%$ (see Table IV) for CASIA and NIVL databases respectively. It is worth noting that such Domain Specific Unit consists in the adaptation of only 928 free parameters. The adaptation of the convolutional kernels W and their corresponding βs, in this domain, provided higher recognition rates overall, but in comparison with the adaption of only βs the improvements were slight. For instance, for the model $\theta_{t[1-5]}(\beta+W)$ the recognition rates were improved from $\approx 73\%$ to $\approx 96\%$ (difference of 7% in comparison with the adaptation of only βs) and from $\approx 88\%$ to $\approx 94\%$ (difference of 1% in comparison with the adaptation of only βs) for CASIA and NIVL databases respectively. Such Domain Specific Unit is more complex and consists in the adaptation of $439,488$ free parameters.

TABLE VI

Inception Resnet v2 - Number of Free Parameters Learnt Adapting Either β or $\beta+W$

	$\theta_{t[1-1]}$	$\theta_{t[1-2]}$	$\theta_{t[1-4]}$	$\theta_{t[1-5]}$	$\theta_{t[1-6]}$
Adapt β	32	208	400	928	3,328
Adapt $\beta+W$	320	33,264	171,696	439,488	1,668,768

Comparing with the VIS-Thermal task, which is our most challenging one, the adaption of $\theta_{t[1-5]}(\beta)$ improved the recognition rates from $\approx 17\%$ to $\approx 37\%$ (see Table V) only, with the same 928 parameters. On the other hand, the adaption of $\theta_{t[1-5]}(\beta+W)$ improved this figure of merit from $\approx 17\%$ to $\approx 76\%$ (difference of 39% in comparison with the adaptation of only βs). This analysis provide an evidence that tasks such as VIS-Thermal are more challenging than VIS-NIR or VIS-Sketch and more complex adaptations are required.

We used in our analysis only the Inception Resnet v2 as a base DCNN architecture. For the sake of page constraints, we provide, as **Supplementary Material**, the same analysis using the Inception Resnet v1 as base architecture.

VI. Conclusion and Future Work

With this work we first showed that DCNN high level features trained with VIS face images provide discriminative power in the Heterogeneous Face Recognition task. Tests carried out in three different image domains have showed that such DCNNs are very accurate for VIS-NIR task. The VIS-Sketch task they are less accurate, but still better than a random guess and better than some baselines in this dataset. The VIS-Thermal task is the most challenging one, but these DCNNs are still better than a random guess.

In order to improve these recognition rates using the discriminative capabilities of such DCNNs already trained for VIS, we introduced a method for HFR called **Domain Specific Units**. Such units learn low level feature detectors that are domain specific and share the same set of high level features from the source domain without re-train them.

Using two different methods to train them (Siamese and Triplet Neural Networks) and two ways to encode such Domain Specific Units ($\theta_{t[1-n]}(\beta)$ and $\theta_{t[1-n]}(\beta+W)$) we showed recognition rates improvements in all image domains that are comparable or better than the state-of-the-art.

For reproducibility purposes of the work, all the source code, trained models and recognition scores are made publicly available.[11]

[11] https://gitlab.idiap.ch/bob/bob.paper.tifs2018_dsu

Future work will focus on the analysis on what such feature detectors are learning for each image domain.

REFERENCES

[1] B. F. Klare and A. K. Jain, "Heterogeneous face recognition using kernel prototype similarities," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 6, pp. 1410–1422, Jun. 2013.

[2] W. Zhao, R. Chellappa, P. J. Phillips, and A. Rosenfeld, "Face recognition: A literature survey," *ACM Comput. Surv.*, vol. 35, no. 4, pp. 399–458, 2003.

[3] H. S. Bhatt, S. Bharadwaj, R. Singh, and M. Vatsa, "Memetically optimized MCWLD for matching sketches with digital face images," *IEEE Trans. Inf. Forensics Security*, vol. 7, no. 5, pp. 1522–1535, Oct. 2012.

[4] H. S. Bhatt, S. Bharadwaj, R. Singh, and M. Vatsa, "On matching sketches with digital face images," in *Proc. 4th IEEE Int. Conf. Biometrics, Theory Appl. Syst. (BTAS)*, Sep. 2010, pp. 1–7.

[5] D. Yi, Z. Lei, and S. Z. Li , "Shared representation learning for heterogenous face recognition," in *Proc. 11th IEEE Int. Conf. Workshops Autom. Face Gesture Recognit. (FG)* vol. 1, May 2015, pp. 1–7.

[6] H. Roy and D. Bhattacharjee, "Local-gravity-face (LG-face) for illumination-invariant and heterogeneous face recognition," *IEEE Trans. Inf. Forensics Security*, vol. 11, no. 7, pp. 1412–1424, Jul. 2016.

[7] S. Z. Li, D. Yi, Z. Lei, and S. Liao, "The CASIA NIR-VIS 2.0 face database," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshops*, Jun. 2013, pp. 348–353.

[8] S. Z. Li, Z. Lei, and M. Ao, "The HFB face database for heterogeneous face biometrics research," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. Workshops (CVPR Workshops)*, Jun. 2009, pp. 1–8.

[9] X. Wang and X. Tang, "Face photo-sketch synthesis and recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 31, no. 11, pp. 1955–1967, Nov. 2009.

[10] W. Zhang, X. Wang, and X. Tang, "Coupled information-theoretic encoding for face photo-sketch recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2011, pp. 513–520.

[11] Z. Lei, S. Liao, A. K. Jain, and S. Z. Li, "Coupled discriminant analysis for heterogeneous face recognition," *IEEE Trans. Inf. Forensics Security*, vol. 7, no. 6, pp. 1707–1716, Dec. 2012.

[12] K. Weiss, T. M. Khoshgoftaar, and D. Wang, "A survey of transfer learning," *J. Big Data*, vol. 3, no. 1, p. 9, 2016.

[13] H. Zhang, V. M. Patel, B. S. Riggan, and S. Hu. (Aug. 2017). "Generative adversarial network-based synthesis of visible faces from polarimetric thermal faces." [Online]. Available: https://arxiv.org/abs/1708.02681

[14] S. Hu *et al.*, "A polarimetric thermal database for face recognition research," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognition Workshops*, Jun./Jul. 2016, pp. 187–194.

[15] O. M. Parkhi, A. Vedaldi, and A. Zisserman, "Deep face recognition," in *Proc. BMVC*. vol. 1, no. 3, 2015, pp. 1–6.

[16] T. Zhang, A. Wiliem, S. Yang, and B. C. Lovell. (Dec. 2017). "TV-GAN: Generative adversarial network based thermal to visible face recognition." [Online]. Available: https://arxiv.org/abs/1712.02514

[17] S. Liao, D. Yi, Z. Lei, R. Qin, and S. Z. Li, "Heterogeneous face recognition from local structures of normalized appearance," in *Proc. Int. Conf. Biometrics*. Berlin, Germany: Springer, 2009.

[18] X. Tan and B. Triggs, "Enhanced local texture feature sets for face recognition under difficult lighting conditions," *IEEE Trans. Image Process.*, vol. 19, no. 6, pp. 1635–1650, Jun. 2010.

[19] M. Pietikäinen, *Computer Vision Using Local Binary Patterns*. vol. 40. Springer, 2011.

[20] S. Liu, D. Yi, Z. Lei, S. Z. Li, "Heterogeneous face image matching using multi-scale features," in *Proc. 5th IAPR Int. Conf. IEEE Biometrics (ICB)*, Mar./Apr. 2012, pp. 79–84.

[21] J. Lu, V. E. Liong, and J. Zhou, "Simultaneous local binary feature learning and encoding for homogeneous and heterogeneous face recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 40, no. 8, pp. 1979–1993, Aug. 2018.

[22] J. Bernhard, J. Barr, K. W. Bowyer, and P. Flynn, "Near-IR to visible light face matching: Effectiveness of pre-processing options for commercial matchers," in *Proc. IEEE 7th Int. Conf. Biometrics Theory, Appl. Syst. (BTAS)*, Sep. 2015, pp. 1–8.

[23] X. Wang and X. Tang, "Dual-space linear discriminant analysis for face recognition," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. (CVPR)* vol. 2, Jun./Jul. 2004, p. 2.

[24] X. Wang and X. Tang, "Random sampling LDA for face recognition," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. (CVPR)* vol. 2, Jun./Jul. 2004, p. 2.

[25] Q. Liu, X. Tang, H. Jin, H. Lu, and S. Ma, "A nonlinear approach for face sketch synthesis and recognition," in *Proc. IEEE Comput. Society Conf. Comput. Vis. Pattern Recognit. (CVPR)*, vol. 1, Jun. 2005, pp. 1005–1010.

[26] Y. Jin, J. Lu, and Q. Ruan, "Coupled discriminative feature learning for heterogeneous face recognition," *IEEE Trans. Inf. Forensics Security*, vol. 10, no. 3, pp. 640–652, Mar. 2015.

[27] T. de Freitas Pereira and S. Marcel, "Heterogeneous face recognition using inter-session variability modelling," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshops*, Jun./Jul. 2016, pp. 179–186.

[28] A. F. Sequeira *et al.*, "Cross-eyed 2017: Cross-spectral IRIS/periocular recognition competition," in *Proc. IEEE/IAPR Int. Joint Conf. Biometrics*, Oct. 2017, pp. 725–732.

[29] D. Yi, Z. Lei, S. Liao, and S. Z. Li. (Nov. 2014). "Learning face representation from scratch." [Online]. Available: https://arxiv.org/abs/1411.7923

[30] Y. Guo, L. Zhang, Y. Hu, X. He, and J. Gao, "MS-Celeb-1M: A dataset and benchmark for large-scale face recognition," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, 2016, pp. 87–102.

[31] I. Kemelmacher-Shlizerman, S. M. Seitz, D. Miller, and E. Brossard, "The megaface benchmark: 1 million faces for recognition at scale," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2016, pp. 4873–4882.

[32] C. Szegedy *et al.*, "Going deeper with convolutions," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2015, pp. 1–9.

[33] X. Glorot and Y. Bengio, "Understanding the difficulty of training deep feedforward neural networks," in *Proc. 13th Int. Conf. Artif. Intell. Statist.*, 2010, pp. 249–256.

[34] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2016, pp. 770–778.

[35] S. Mallat, "Understanding deep convolutional networks," *Philos. Trans. Roy. Soc. London A, Math. Phys. Sci.*, vol. 374, no. 2065, p. 20150203, 2016.

[36] X. Wu, R. He, Z. Sun, and T. Tan. (Nov. 2015). "A light CNN for deep face representation with noisy labels." [Online]. Available: https://arxiv.org/abs/1511.02683

[37] F. Schroff, D. Kalenichenko, and J. Philbin, "FaceNet: A unified embedding for face recognition and clustering," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2015, pp. 815–823.

[38] C. Szegedy, S. Ioffe, V. Vanhoucke, and A. A. Alemi, "Inception-V4, inception-resnet and the impact of residual connections on learning," in *Proc. AAAI*, vol. 4, 2017, p. 12.

[39] J. Yosinski, J. Clune, Y. Bengio, and H. Lipson, "How transferable are features in deep neural networks?" in *Proc. Adv. Neural Inf. Process. Syst.*, 2014, pp. 3320–3328.

[40] S. Chopra, R. Hadsell, and Y. LeCun, "Learning a similarity metric discriminatively, with application to face verification," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, vol. 1, Jun. 2005, pp. 539–546.

[41] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *Proc. 32nd Int. Conf. Mach. Learn. (ICML)*, vol. 37, F. Bach and D. Blei, Eds., 2015, pp. 448–456.

[42] E. Meyers and L. Wolf, "Using biologically inspired features for face processing," *Int. J. Comput. Vis.*, vol. 76, no. 1, pp. 93–104, 2008.

[43] X. Liu, L. Song, X. Wu, and T. Tan, "Transferring deep representation for NIR-VIS heterogeneous face recognition," in *Proc. IEEE Int. Conf. Biometrics (ICB)*, Jun. 2016, pp. 1–8.

[44] T. I. Dhamecha, P. Sharma, R. Singh, and M. Vatsa, "On effectiveness of histogram of oriented gradient features for visible to near infrared face matching," in *Proc. 22nd Int. Conf. IEEE Pattern Recognit. (ICPR)*, Aug. 2014, pp. 1788–1793.

[45] K. Simonyan and A. Zisserman. (Sep. 2014). "Very deep convolutional networks for large-scale image recognition." [Online]. Available: https://arxiv.org/abs/1409.1556

[46] L. Wolf, T. Hassner, and Y. Taigman, "Descriptor based methods in the wild," in *Proc. Faces Real-Life Images Workshop Eur. Conf. Comput. Vis. (ECCV)*, Marseille, France, 2008.

[47] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2012, pp. 1097–1105.

[48] H. Li, H. Lu, Z. Lin, X. Shen, and B. Price. (Aug. 2015). "LCNN: Low-level feature embedded CNN for salient object detection." [Online]. Available: https://arxiv.org/abs/1508.03928

[49] M. Shao and Y. Fu, "Cross-modality feature learning through generic hierarchical hyperlingual-words," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 28, no. 2, pp. 451–463, Feb. 2016.

[50] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *Int. J. Comput. Vis.*, vol. 60, no. 2, pp. 91–110, 2004.

[51] D. Sandberg. *FaceNet: Face Recognition Using TensorFlow*. Accessed: Mar. 27, 2018. [Online]. Available: https://github.com/davidsandberg/facenet,

[52] K. Zhang, Z. Zhang, Z. Li, and Y. Qiao, "Joint face detection and alignment using multitask cascaded convolutional networks," *IEEE Signal Process. Lett.*, vol. 23, no. 10, pp. 1499–1503, Oct. 2016.

[53] Y. Wen, K. Zhang, Z. Li, and Y. Qiao, "A discriminative feature learning approach for deep face recognition," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, 2016, pp. 499–515.

[54] R. He, X. Wu, Z. Sun, and T. Tan, "Learning invariant deep representation for NIR-VIS face recognition," in *Proc. AAAI*, vol. 4, 2017, p. 7.

[55] X. Wu, L. Song, R. He, and T. Tan, "Coupled deep learning for heterogeneous face recognition," in *Proc. AAAI*, 2018.

[56] R. He, X. Wu, Z. Sun, and T. Tan, "Wasserstein CNN: Learning invariant features for NIR-VIS face recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, doi: 10.1109/TPAMI.2018.2842770.

Authors' photographs and biographies not available at the time of publication.