Semi-supervised Cross-Spectral Face Recognition with Small Datasets

Anirudh Nanduri
University of Maryland
College Park, MD
snanduri@umd.edu

Rama Chellappa
Johns Hopkins University
Baltimore, MD
rchella4@jhu.edu

Abstract

While systems based on deep neural networks have produced remarkable performance on many tasks such as face/object detection and recognition, they also require large amounts of labeled training data. However, there are many applications where collecting a relatively large labeled training data may not be feasible due to time and/or financial constraints. Trying to train deep networks on these small datasets in the standard manner usually leads to serious over-fitting issues and poor generalization. In this work, we explore how a state-of-the-art deep learning pipeline for unconstrained visual face identification and verification can be adapted to domains with scarce data/label availability using semi-supervised learning. The rationale for system adaptation and experiments are set in the following context - given a pretrained network (that was trained on a large training dataset in the source domain), adapt it to generalize onto a target domain using a relatively small labeled (typically hundred to ten thousand times smaller) and an unlabeled training dataset. We present algorithms and results of extensive experiments with varying training dataset sizes and composition, and model architectures using the IARPA JANUS Benchmark Multi-domain Face dataset for training and evaluation with visible and short-wave infrared domains as the source and target domains respectively.

1. Introduction

There has been tremendous progress in the field of face recognition in the deep learning era, tackling even unconstrained, in-the-wild scenarios. But most of the work has been done on images/videos in the visible spectrum. Cross-spectral face recognition [13] refers to the class of problems where data collected from one part of the spectrum (or domain) is compared against data from another part of the spectrum. Some of the commonly used non-visible domains include near-infrared/NIR (750 *nm* - 1100 *nm*), short-wave infrared/SWIR (1100 *nm* - 2500 *nm*), medium-wave infrared/MWIR (3000 *nm* - 5000 *nm*), and long-wave infrared/LWIR (7000 *nm* - 14000 *nm*). Complementing the visible images with data from these domains can have many advantages. For example, under low-light conditions, NIR and SWIR images have higher SNR compared to visible images. They are also more robust to atmospheric conditions like rain, fog or smoke.

Examples of cross-spectral face datasets include Equinox [35], NVIE [37], LDHF-DB [23], CASIA NIR-VIS 2.0 [21], EURECOM [24], IJB-MDF [16], and ARL-VTF [30]. The IJB-MDF dataset comprises of images and videos captured using a variety of cameras: fixed and body-worn, capable of imaging at visible, short-wave, mid-wave and long-wave infrared wavelengths at distances up to 500m. Some sample images from the IJB-MDF dataset (after cropping and alignment) are shown in Figure 1. Domains-11, 12, 13 and 14 refer to SWIR - captured without a filter, captured at 1150nm, 1350nm, and 1550nm respectively.

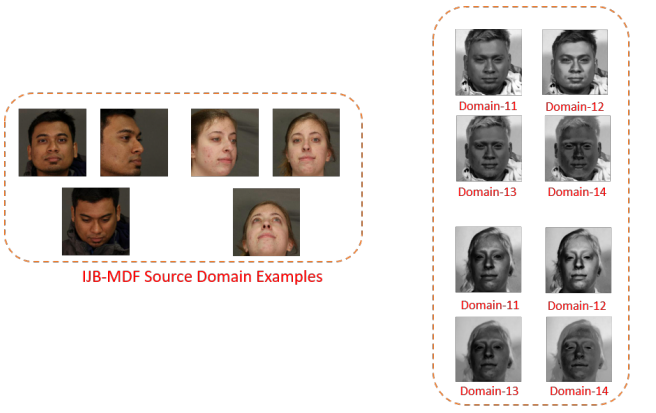

Figure 1. IJB-MDF dataset sample images. Source domain - visible. Target domains - SWIR at different wavelengths.

Based on the availability of labels in the training data, the following three learning paradigms are defined - supervised learning, semi-supervised learning and unsupervised

learning. Supervised learning assumes that all the training examples come with their corresponding labels. This is one of the most common learning paradigms and comparatively the easiest. In semi-supervised learning, labels for some of the training data are missing. Finally, unsupervised learning algorithms are trained without access to any labels. In this paper, we look at the problem of face recognition in a context that involves transfer learning, domain adaptation, and limited labeled data (semi-supervised learning). Although there has been a lot of work done in each of these areas separately, there has not been much research done in the area which lies at their intersection. The aim of this paper is to design interesting experiments that help shed light on more fundamental aspects of training with very limited labeled data in cross-spectral face recognition - how the composition of the training dataset, complexity of the network, and the domain gap between the training and test datasets affect the performance.

A natural way to deal with the lack of labeled data is data generation and data augmentation [5, 28, 38]. Generative Adversarial Networks (GANs) and Denoising Diffusion Probabilistic Models (DDPMs) have improved face synthesis significantly over the past few years. NVIDIA's StyleGAN [17] and StyleGAN2 [18] can generate images of remarkable quality. Image-to-image translation based methods for transferring the attributes of one image onto another such as Few Shot Unsupervised Image to Image translation (FUNIT) [22] and StarGAN v2 [4] can then be used to increase the diversity of the classes generated by StyleGAN. Recently, DDPMs [11] have been shown to outperform GANs [6]. They have been used for generating synthetic face datasets [19] and also for image-to-image translation from thermal to visible domain [26]. But the major concern with using these image generation approaches for augmenting small datasets is that these models do not generalize to domains that are very different from the domain of their training data. And since we do not have access to a lot of annotated data in the non-visible domains to train these models, data augmentation is not effective.

The following are the main contributions of this paper:

- We introduce the problem of semi-supervised cross-spectral face recognition in the context of small training datasets.
- We describe an end-to-end system for solving this problem.
- We present extensive experiments with different training datasets and network architectures to explore their impact on performance and gain useful insights for tackling this problem.

Collecting and labeling data is a very expensive process and we believe that the results of our experiments which analyze the effect of dataset composition (specifically, effects of adding unlabeled data from the same or similar domains), will be very helpful in efficient data collection.

The rest of the paper is organized as follows: Section 2 details some works related to cross-spectral face recognition; Section 3 describes the problem formulation, the pipeline and the evaluation protocol; in Section 4 we present the experiments and results; and finally Section 5 contains the conclusions and future work to extend this work.

2. Related Work

Most of the existing works on cross-spectral face recognition are in the supervised learning regime.

Bourlai et al. [3] published one of the first papers which looked into the problem of cross-spectral SWIR face recognition. They collected the WVU Multispectral dataset with 50 subjects, 1,250 VIS and 1,350 SWIR images, and presented cross-spectral matching results using classical face recognition methods like PCA with k-NN. Kalka et al. [15] extended the work in [3] to heterogenous face recognition in semi-controlled and uncontrolled environments. Nicolo et al. [27] proposed an algorithm for SWIR-VIS matching that encodes the magnitude and phase of images filtered with a Gabor filter bank using Simplified Weber Local Descriptor, Local Binary Pattern and Generalized Local Binary Pattern. Bourlai et al. [2] studied SWIR-VIS, MWIR-MWIR, MWIR-VIS and NIR-VIS matching and extended the work presented in [15] to more challenging scenarios (cross-distance matching) and other domains like MWIR and NIR.

Maeng et al. [23] collected the Long Distance Heterogeneous Face Database (LDHF-DB) with VIS and NIR images captured at short and long distances. They proposed Gauss-SIFT algorithm and reported results on both intra-spectral and cross-spectral cross-distance matching. Juefei-Xu et al. [14] proposed a dictionary learning approach to learn a mapping function between VIS and NIR domains, thus reducing the problem of cross-spectral matching to intra-spectral matching. Lezama et al. [20] proposed a deep learning-based approach which involves producing VIS images from NIR images by adapting a deep network pre-trained on VIS images to generate discriminative features from both VIS and NIR images. They also applied a low-rank embedding to the deep features which restores a low-rank structure for the cross-spectral features from the same subject. Song et al. [36] proposed a deep network with cross-spectral face hallucination and discriminative feature learning for VIS-NIR matching using a GAN, by employing an adversarial loss and a high-order variance discrepancy loss to measure the global and local discrepancy between the domains. He et al. [10] extended the work reported in [36] by performing cross-spectral face hallucination using inpainting of VIS image textures from NIR textures and,

pose correction to generate VIS images at frontal pose.

Fu et al. [7] proposed a Dual Variational Generation framework to learn the joint distribution of paired heterogeneous images, and then generated paired images from the two domains. These generated images are used to train a face recognition network using a contrastive learning mechanism. Peri et al. [29] proposed another synthesis based approach using GAN architectures for thermal-to-visible face verification. In contrast to generative models, Miao et al. [25] used a physically-based renderer to generate a large dataset of NIR-VIS image pairs. While all these works have focused on fully-supervised learning, we look into the problem of semi-supervised cross-spectral face recognition.

3. Problem Formulation

We set the problem in the following context: given a pre-trained network (that was trained on a large training dataset in source domain), adapt it to generalize onto a target domain using a relatively small training dataset (that is typically hundred to ten thousand times smaller). The training data of this semi-supervised (few shot) domain adaptation problem, consists of a small labeled source $D_s^l = \{(x_i^s, y_i^s)\}_{i=1}^{n_s}$, a small labeled target $D_t^l = \{(x_i^t, y_i^t)\}_{i=1}^{n_{tl}}$, and a large unlabeled target dataset $D_t^u = \{(x_i^t)\}_{i=1}^{n_{tu}}$. D_s^l and D_t^l are assumed to share the same label space with each other, but not necessarily with D_t^u. Typically $n_s \approx n_{tl} \ll n_{tu}$. In this context, the term "few shot" means that the models only have access to a very small amount of labeled samples per class. For example, a dataset we use in our work has 126 classes, $n_s = n_{tl} = 882$, and $n_{tu} = 3906$. So the total size of the dataset is just 5,669 images.

We now describe the face recognition pipeline as shown in figure 2.

3.1. Face Detection

Face detection is the first module in any face recognition pipeline. We employ the Deep Pyramid Single Shot Face Detector (DPSSD) algorithm presented in [32]. This uses a modified Single Shot Detector (SSD) algorithm so as to be able to detect extremely small faces also. This is achieved by adding additional convolutional layers at the end of the VGG-16 architecture of the SSD model to detect faces at different scales.

DPSSD is trained on the WIDER face [39] dataset. More details about the architecture and training can be found in [32].

3.2. Keypoint Detection and Alignment

Face keypoints include centers and corners of eyebrows, eyes, nose, mouth, earlobes and chin. The All-in-One Face framework [33] is used for keypoint localization. This method simultaneously does tasks such as face detection, face alignment, pose estimation, age estimation etc. The all-in-one model is trained using a multi-task learning framework which helps it to learn the different tasks synergistically.

Figure 3 shows some examples of the results of the face detector and keypoint detector on SWIR images. We can see that even though both the DPSSD face detector and All-in-One Face network were trained on VIS images, they perform well when applied on SWIR images.

3.3. Face Verification

For training our feature extractor in a semi-supervised manner, we use a combination of crystal loss [31] and entropy loss [34].

Crystal loss, given by (1) constrains all the features of the deep network to be on a hypersphere with radius α.

$$\begin{aligned} \text{minimize} \; & -\frac{1}{M}\sum_{i=1}^{M}\log\frac{e^{W_{y_i}^T f(\mathbf{x}_i)+b_{y_i}}}{\sum_{j=1}^{C} e^{W_j^T f(\mathbf{x}_i)+b_j}} \\ \text{subject to} \; & \|f(\mathbf{x}_i)\|_2 = \alpha, \quad \forall i = 1, 2, ..., M, \end{aligned} \tag{1}$$

where ($\mathbf{x}_i$, y_i) are the input image and its label, M is the mini-batch size, $f(\mathbf{x}_i)$ is the feature vector extracted from the penultimate layer of the deep network, C is the number of classes, W and b are the weights and bias for the classification layer of the network, and α represents the L_2-constraint on the norm of the feature vector.

Entropy loss, given by (2) is applied on the unlabeled data such that features are both class dicriminative and domain invariant. This is achieved by training the classifier to maximize the entropy loss and by training the feature extractor to minimize the entropy loss. The intuition behind entropy loss is that, maximizing the entropy of predictions on the target unlabeled data by the classifier brings the class-mean-features (class prototypes) away from the source domain and closer to the target domain, and minimizing the entropy by the feature extractor clusters all the class features around the class-mean-feature.

$$L_{entropy} = -\mathbb{E}_{(\mathbf{x},y)\in D_u}\sum_{i=1}^{C} p(y=i|\mathbf{x})\log p(y=i|\mathbf{x}) \tag{2}$$

where C is the number of classes, and $p(y=i|\mathbf{x})$ represents the probability that $\mathbf{x}$ is predicted to belong to class i.

During training, given a mini-batch consisting of both labeled (VIS and SWIR) and unlabeled examples (SWIR only), we apply the crystal loss on the labeled samples and entropy loss on the unlabeled samples, so that the final loss functions for the feature extractor and the classifier are given by (3)

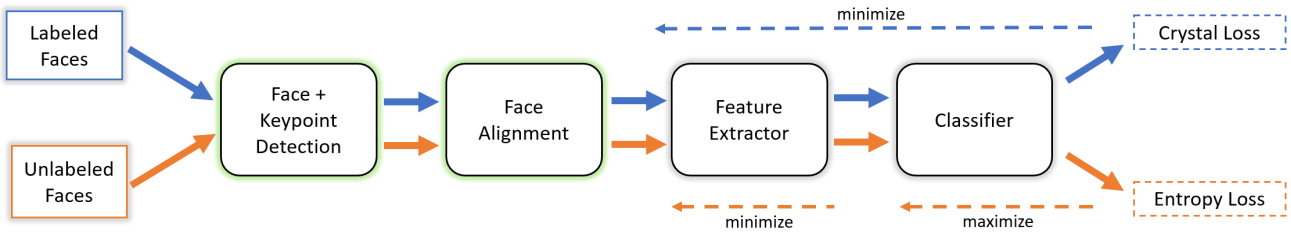

Figure 2. The face recognition pipeline: crystal loss is calculated on the labeled images, and entropy loss is calculated on the unlabeled images. Loss back-propagation is shown by the dotted backward-arrows.

Figure 3. Face and Keypoint Detection on SWIR images

$$L_{feature-extractor} = L_{crystal} + \lambda L_{entropy}$$
$$L_{classifier} = L_{crystal} - \lambda L_{entropy} \quad (3)$$

where λ is the hyperparameter which decides the weight given to entropy loss over the crystal loss.

3.4. Evaluation Protocol

The evaluation protocol is 1:N identification - query images from target domain are compared against gallery templates from source domain. The 875 visible enrollment images are folded into 125 templates (one for each subject) by averaging their features and form the gallery set. There are about 20,000 images from domains-11, 12, 13 and 14 in the query set. We compare the cosine similarity scores between the deep features of each query image and all the gallery templates to predict the label.

4. Experiments

4.1. Dataset

4.1.1 IJB-MDF

The IARPA JANUS Benchmark Multi-domain Face (IJB-MDF) [16] dataset consists of images and videos of 251 subjects captured using a variety of cameras corresponding to visible, short-, mid-, and long-wave infrared and long range surveillance domains. There are 1,757 visible enrollment images, 40,597 short-wave infrared (SWIR) enrollment images and over 800 videos spanning 161 hours. The dataset can be requested from the authors of [16] as stated in that paper.

We divide the 251 subjects into two disjoint sets of 126 and 125 to be used for training and testing respectively. The visible enrollment images form the source labeled dataset D_s^l with 882 images in total and seven images per subject. As for the target domain datasets D_t^l and D_t^u, we use short-wave infrared (SWIR) enrollment images from four sub-domains: Domain 11: SWIR (no filter), Domain 12: SWIR (captured at 1150 nm), Domain 13: SWIR (captured at 1350 nm), and Domain 14: SWIR (captured at 1550 nm).

We generate nine different training datasets with varying compositions of D_t^l and D_t^u as shown in Table 1. The first four datasets (trainset-v1-11, 12, 13, and 14) in the table contain data from only a single target domain (either domain-11, 12, 13 or 14). The next four datasets (trainset-v2-11, 12, 13 and 14) have unlabeled data added from all four target domains. Finally the last dataset trainset-v3 contains both labeled and unlabeled data from all the target domains. The numbers in Table 1 represent the number of images per subject in each of the source and target domains.

4.2. Training Details

We first train our network on a large source domain dataset D_{large}, using crystal Loss [31] and evaluate it using our evaluation protocol. This will form our baseline. We set the crystal loss α parameter to 50 for these baseline experiments.

We then further train these baseline models (pretrained on D_{large}) on D_{small} using crystal loss [31] and entropy loss [34] as described in section 3.3. The entropy loss is applied on the unlabeled data D_t^u, and is maximized by the classifier (W) and minimized by the feature extractor (F) of the network. Crystal loss is applied on the labeled source D_s^l and target D_t^l data as in the baseline network. The parameter α in crystal loss is set to 10 and λ in the entropy loss is set to 0.1 in most of these experiments.

We train our base networks on UniverseFaces dataset

Table 1. Composition of Different Training Datasets Used in the Experiments.

Dataset ↓	D_s^l (source-labeled) (imgs/sub)	D_t^l (target-labeled) (imgs/sub)				D_t^u (target-unlabeled) (imgs/sub)				**Total #imgs**
Domain →	***Vis***	***11***	***12***	***13***	***14***	***11***	***12***	***13***	***14***	
trainset-v1-11	7	7	-	-	-	31	-	-	-	5669
trainset-v1-12	7	-	7	-	-	-	31	-	-	5667
trainset-v1-13	7	-	-	7	-	-	-	31	-	5669
trainset-v1-14	7	-	-	-	7	-	-	-	31	5668
trainset-v2-11	7	7	-	-	-	31	41	41	41	20987
trainset-v2-12	7	-	7	-	-	41	31	41	41	20987
trainset-v2-13	7	-	-	7	-	41	41	31	41	20987
trainset-v2-14	7	-	-	-	7	41	41	41	31	20987
trainset-v3	7	7	7	7	7	34	34	34	34	20987

(which is a union of cleaned MS-Celeb-1M [8] and UMDFaces [1] datasets) with 40k subjects and 5M images. We conduct experiments with the following architectures: Resnet-50 [9], Resnet-101 [9], Resnet-152 [9], SENet-50 [12], SENet-101 [12], and SENet-152 [12]. The Resnet models are initialized from their ImageNet-pretrained weights while the Squeeze-and-Excitation nets (SENets) are trained from scratch on the UniverseFaces dataset (since the ImageNet-pretrained weights are not publicly available for SENet-101 and SENet-152).

We train all of these networks on each of the nine training datasets listed in Table 1. Each of the network training experiments are repeated 5 times and the average performance of the 5 runs is presented (along with the standard deviation wherever possible). Unless otherwise specified, the query-set in the test dataset consists of data from all 4 SWIR sub-domains (11, 12, 13 and 14).

4.3. Results

4.3.1 Effect of network architecture

We choose a set of six networks (Resnet-50, Resnet-101, Resnet-152, SENet-50, SENet-101, and SENet-152) to study the effect of network architecture on the performance. Since it is beyond the scope of this paper to conduct an exhaustive search for the best network architecture for this problem, we primarily focus on the effect of network complexity. Table 2 compares the performance of these networks trained on trainset v3 dataset and the performance of their corresponding baseline networks. Even though the absolute performance of Resnet-152 is the best, the relative improvement over the baseline is highest for the SENets.

Even when the training dataset is small, we observe that as we increase the network size/complexity there is no overfitting and the performance consistently improves.

Table 2. Rank 1 retrieval rates of different network architectures trained on trainset v3 and their corresponding baselines

Network	Baseline	Trained on trainset v3
Resnet-50	63.19	92.19 ± 0.53
Resnet-101	74.82	96.62 ± 0.34
Resnet-152	75.93	96.84 ± 0.21
SENet-50	65.85	96.15 ± 0.10
SENet-101	66.41	95.10 ± 0.33
SENet-152	70.79	96.41 ± 0.32

4.3.2 Effect of training dataset composition

Column 1 (Test domain: All) of table 3 compares the performance of Resnet-152 networks trained with various training datasets. Networks trained with trainsets v2(-11, 12, 13 and 14) perform better than corresponding networks trained with trainsets v1(-11, 12, 13 and 14), which shows that adding more unlabeled data of other domains helps. As we would expect, the network trained on trainset v3 (which has the most labeled data) outperforms all the other networks.

Within trainsets v1 and trainsets v2, we observe the following performance trends:

$$\text{v1-13} \approx \text{v1-14} > \text{v1-11} > \text{v1-12} \tag{4}$$

$$\text{v2-13} \approx \text{v2-14} \approx \text{v2-12} > \text{v2-11} \tag{5}$$

Since the test dataset has images from all 4 SWIR sub-domains, from (4) we can infer that, the data from domains-13 and 14 is richer in information compared to domain-11 and domain-12. And (5) shows that when the training set has unlabeled data from all the four sub-domains, the performance of the network is more or less the same if labeled data from either of domains-12, 13 or 14 is added. But adding labeled data from domain-11 does not add as much information. This can be explained by the fact that domain-11 is closest to the visible domain in the spectrum and since

Table 3. Rank 1 retrieval rates of Resnet-152 trained on different trainsets and tested on different domains

Training set	Test domain: All	Test domain: 11	Test domain: 12	Test domain: 13	Test domain: 14
baseline	75.93	86.73	92.33	76.36	48.29
v1-11	89.50 ± 0.88	97.52 ± 0.34	97.84 ± 0.12	93.27 ± 1.20	69.35 ± 2.29
v1-12	84.98 ± 0.66	95.88 ± 0.45	97.64 ± 0.39	88.76 ± 0.78	57.63 ± 1.77
v1-13	93.10 ± 0.30	97.47 ± 0.17	97.37 ± 0.23	96.60 ± 0.18	80.96 ± 0.86
v1-14	93.09 ± 0.67	93.66 ± 0.68	93.70 ± 1.00	93.37 ± 0.51	91.64 ± 0.90
v2-11	92.26 ± 4.48	97.00 ± 1.14	97.54 ± 0.76	94.09 ± 3.12	80.41 ± 12.97
v2-12	94.81 ± 2.42	97.57 ± 0.63	97.87 ± 0.33	95.85 ± 1.81	87.94 ± 7.11
v2-13	95.70 ± 0.92	97.66 ± 0.44	97.68 ± 0.32	96.77 ± 0.55	90.70 ± 2.45
v2-14	95.32 ± 0.22	97.09 ± 0.19	97.42 ± 0.23	95.53 ± 0.47	91.23 ± 0.38
v3	$\mathbf{96.84 \pm 0.21}$	$\mathbf{98.16 \pm 0.19}$	$\mathbf{98.27 \pm 0.20}$	$\mathbf{97.56 \pm 0.25}$	$\mathbf{93.35 \pm 0.36}$

the baseline network is already trained on a large visible dataset, finetuning on data from domain-11 adds little to the generalization capabilities of the network onto other domains.

4.3.3 Effect of test dataset domain

Now we look at the performance of the base networks on different subsets of the test data. As mentioned earlier, the test query images are from all the SWIR domains (11, 12 ,13 and 14). So we separate the test query data into four subsets each containing images from domains 11, 12, 13 and 14 respectively.

Row 1 of table 3 shows the performance of Resnet-152 base network on these four test subsets and the original complete test set. The increasing order of difficulty of the test domains is: domain-12 $<$ domain-11 $<$ domain-13 $<$ domain-14. This implies that the baseline network trained on VIS images finds it hardest to generalize to domain-14.

The remaining rows show the performance of Resnet-152 networks trained on trainsets v1- and v2-(11, 12, 13 and 14) on separate test domains. One common trend that we observe is that almost all the networks perform best on domains 11 and 12, followed by the domain which was predominant in the trainset. Networks trained on trainsets containing data that is predominantly from domains-11, 12 and 13 do not generalize to domain-14. Another interesting observation is the very large standard deviation in the networks trained on v2-11, 12 and 13 when tested on domain-14 compared to the network trained on trainset v2-14. This seems to indicate that when you add unlabeled data from domain-14 to any of the trainsets v1-11, 12 or 13, the improvement in performance of the resulting network on domain-14 has significant variance.

4.3.4 Effect of width and depth of training data

In table 4, we compare the performance of Resnet-152 networks trained on wide and deep training datasets. We observe that training with a wider dataset (trainset v1-11b: 3 labeled samples per subject and 126 subjects) yields better performance than training with a deep dataset (trainset v1-11d: 7 labeled samples per subject and 50 subjects) of similar overall size. We also observe that adding unlabeled examples from subjects not in the labeled data (as in trainset v1-11c) significantly degrades the performance of the network. This is an unfortunate limitation of entropy loss in this case, because in most practical scenarios, cleaning the unlabeled data of specific subjects essentially amounts to labeling the unlabeled data. To alleviate this issue, we trained the network without maximizing the entropy loss on the classifier and only back-propagating it on the feature extractor. The modified loss function is given by (6). With this change, rank 1 retrieval rate of this network increased to 82.43 ± 0.68 from 74.06 ± 7.52. Despite not quite reaching the same performance as the network trained on trainset v1-11d (unlabeled data taken only from subjects with labeled data), we observe a significant improvement when we remove the classifier entropy loss.

$$\begin{aligned} L_{feature-extractor} &= L_{crystal} + \lambda L_{entropy} \\ L_{classifier} &= L_{crystal} \end{aligned} \tag{6}$$

From table 4, we also observe that as we increase the depth of the training dataset from 1 image per subject to 7 images per subject, the performance of the network saturates at about 3 images per subject. This could be because the main variation in the enrollment images of a subject is their pose, and the baseline network has already learned pose-invariance during its pre-training. So adding more labeled images to the training data adds little overall information.

4.4. Ablation Study

In this section, we evaluate the impact of various components of our loss function. Specifically we try training the networks without the crystal loss, without the entropy

Table 4. Rank 1 retrieval rates of networks trained on wide and deep datasets

Dataset	D_s^l **(imgs/sub)**	D_t^l **(imgs/sub)**	D_t^u **(imgs/sub)**	**#subjects**			**#imgs**	**rank-1**
Domain	***Vis***	***11***	***11***	D_s^l	D_t^l	D_t^u		
v1-11	7	7	31	126	126	126	5669	89.50 ± 0.88
v1-11b	3	3	31	126	126	126	4665	88.88 ± 0.79
v1-11c	7	7	31	50	50	126	4609	74.06 ± 7.52
v1-11d	7	7	31	50	50	50	2253	84.82 ± 0.92
v1-11e	1	1	31	126	126	126	4161	83.39 ± 0.41
v1-11f	2	2	31	126	126	126	4413	86.97 ± 0.63
v1-11g	5	5	31	126	126	126	5169	89.23 ± 0.82

loss on the classifier and finally without the entropy loss. We also present results with different α values in the crystal loss.

From our experiments we see that crystal loss is crucial for the network to perform well. Replacing the crystal loss with cross entropy loss decreases the performance from 96.84 ± 0.21 to 40.26 ± 5.38. This can partly be attributed to the fact that the base network is trained with crystal loss and so finetuning that network without crystal loss is more difficult. But interestingly, we notice that even though the test performance is so poor, the validation accuracy of the networks was still around 99%.

Next, we see how much impact the entropy loss (unlabeled data) has on the performance when trained with trainset v1-11 and trainset v3. When we remove the entropy loss, the network trained on trainset v3 showed a drop of 0.63% in the performance; while the network trained with trainset v1-11 showed 1.15% drop. This shows that as the amount of labeled data increases, the impact of unlabeled data in the training set decreases.

Table 5 shows the affect of the hyperparameter α in the crystal loss on the performance of the network. For a classification problem with C classes, [31] provide a lower limit for α to achieve a classification probability p in equation 7

$$\alpha_{low} = \log \frac{p(C-2)}{1-p} \tag{7}$$

In our case, with $C = 126$ and $p = 0.99$, α_{low} comes out to be around 9.42. From table 5 we do observe that the best performance is achieved when $\alpha = 10$.

5. Conclusion

We discussed the problem of semi-supervised cross-spectral face recognition in the context of very small training datasets using large pretrained models. Through extensive experiments, we explored how different training dataset compositions impact the generalization capability of the trained network on different test domains. We confirmed that larger models perform better than smaller architectures

Table 5. Rank 1 retrieval rates of with different α values

α	rank-1
2	94.20 ± 0.92
5	95.90 ± 0.38
8	96.47 ± 0.38
10	96.84 ± 0.21
20	96.51 ± 0.61
30	95.78 ± 0.31
50	90.34 ± 7.76

even with very small training datasets without overfitting. When we have multiple target domains and there are constraints on the amount of data that can be collected, it may be better to prioritize collecting training data for the hardest domain (the domain farthest from the source domain). Wider training datasets (more classes and less samples per class) perform better than deep datasets (less classes and more samples per class). When the unlabeled data is noisy (some of them may belong to classes not represented in the labeled data), back-propagating the entropy loss only on the feature-extractor (and not the classifier), significantly boosts the performance. When designing a dataset for finetuning, trying to label more data without making sure that there are novel variations in the data (like pose or illumination) will not help improve performance consistently.

Acknowledgments

This research is based upon work supported by the Office of the Director of National Intelligence (ODNI), Intelligence Advanced Research Projects Activity (IARPA), via IARPA R&D Contract No. 2014-14071600012. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of the ODNI, IARPA, or the U.S. Government. The U.S. Government is authorized to reproduce and distribute reprints for Governmental purposes notwithstanding any copyright annotation thereon.

References

[1] Ankan Bansal, Anirudh Nanduri, Carlos D Castillo, Rajeev Ranjan, and Rama Chellappa. Umdfaces: An annotated face dataset for training deep networks. In *2017 IEEE International Joint Conference on Biometrics (IJCB)*, pages 464–473. IEEE, 2017. 5

[2] Thirimachos Bourlai and Bojan Cukic. Multi-spectral face recognition: Identification of people in difficult environments. In *2012 IEEE International Conference on Intelligence and Security Informatics*, pages 196–201. IEEE, 2012. 2

[3] Thirimachos Bourlai, Nathan Kalka, Arun Ross, Bojan Cukic, and Lawrence Hornak. Cross-spectral face verification in the short wave infrared (swir) band. In *2010 20th International Conference on Pattern Recognition*, pages 1343–1347. IEEE, 2010. 2

[4] Yunjey Choi, Youngjung Uh, Jaejun Yoo, and Jung-Woo Ha. Stargan v2: Diverse image synthesis for multiple domains. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8188–8197, 2020. 2

[5] Ekin D Cubuk, Barret Zoph, Dandelion Mane, Vijay Vasudevan, and Quoc V Le. Autoaugment: Learning augmentation strategies from data. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 113–123, 2019. 2

[6] Prafulla Dhariwal and Alexander Nichol. Diffusion models beat gans on image synthesis. *Advances in neural information processing systems*, 34:8780–8794, 2021. 2

[7] Chaoyou Fu, Xiang Wu, Yibo Hu, Huaibo Huang, and Ran He. Dvg-face: Dual variational generation for heterogeneous face recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2021. 3

[8] Yandong Guo, Lei Zhang, Yuxiao Hu, Xiaodong He, and Jianfeng Gao. Ms-celeb-1m: A dataset and benchmark for large-scale face recognition. In *European conference on computer vision*, pages 87–102. Springer, 2016. 5

[9] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 5

[10] Ran He, Jie Cao, Lingxiao Song, Zhenan Sun, and Tieniu Tan. Adversarial cross-spectral face completion for nir-vis face recognition. *IEEE transactions on pattern analysis and machine intelligence*, 42(5):1025–1037, 2019. 2

[11] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 6840–6851. Curran Associates, Inc., 2020. 2

[12] Jie Hu, Li Shen, and Gang Sun. Squeeze-and-excitation networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7132–7141, 2018. 5

[13] Shuowen Hu, Nathaniel Short, Benjamin S Riggan, Matthew Chasse, and M Saquib Sarfraz. Heterogeneous face recognition: Recent advances in infrared-to-visible matching. In *2017 12th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2017)*, pages 883–890. IEEE, 2017. 1

[14] Felix Juefei-Xu, Dipan K Pal, and Marios Savvides. Nir-vis heterogeneous face recognition via cross-spectral joint dictionary learning and reconstruction. In *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, pages 141–150, 2015. 2

[15] Nathan D Kalka, Thirimachos Bourlai, Bojan Cukic, and Lawrence Hornak. Cross-spectral face recognition in heterogeneous environments: A case study on matching visible to short-wave infrared imagery. In *2011 International Joint Conference on Biometrics (IJCB)*, pages 1–8. IEEE, 2011. 2

[16] Nathan D Kalka, James A Duncan, Jeremy Dawson, and Charles Otto. Iarpa janus benchmark multi-domain face. In *2019 IEEE 10th International Conference on Biometrics Theory, Applications and Systems (BTAS)*, pages 1–9. IEEE, 2019. 1, 4

[17] Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4401–4410, 2019. 2

[18] Tero Karras, Samuli Laine, Miika Aittala, Janne Hellsten, Jaakko Lehtinen, and Timo Aila. Analyzing and improving the image quality of stylegan. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8110–8119, 2020. 2

[19] Minchul Kim, Feng Liu, Anil Jain, and Xiaoming Liu. Dcface: Synthetic face generation with dual condition diffusion model. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12715–12725, 2023. 2

[20] José Lezama, Qiang Qiu, and Guillermo Sapiro. Not afraid of the dark: Nir-vis face recognition via cross-spectral hallucination and low-rank embedding. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 6628–6637, 2017. 2

[21] Stan Li, Dong Yi, Zhen Lei, and Shengcai Liao. The casia nir-vis 2.0 face database. In *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, pages 348–353, 2013. 1

[22] Ming-Yu Liu, Xun Huang, Arun Mallya, Tero Karras, Timo Aila, Jaakko Lehtinen, and Jan Kautz. Few-shot unsupervised image-to-image translation. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 10551–10560, 2019. 2

[23] Hyunju Maeng, Shengcai Liao, Dongoh Kang, Seong-Whan Lee, and Anil K Jain. Nighttime face recognition at long distance: Cross-distance and cross-spectral matching. In *Asian Conference on Computer Vision*, pages 708–721. Springer, 2012. 1, 2

[24] Khawla Mallat and Jean-Luc Dugelay. A benchmark database of visible and thermal paired face images across multiple variations. In *2018 International Conference of the Biometrics Special Interest Group (BIOSIG)*, pages 1–5. IEEE, 2018. 1

[25] Yunqi Miao, Alexandros Lattas, Jiankang Deng, Jungong Han, and Stefanos Zafeiriou. Physically-based face rendering for nir-vis face recognition. In *NeurIPS 2022*, 2022. 3

[26] Nithin Gopalakrishnan Nair and Vishal M Patel. T2v-ddpm: Thermal to visible face translation using denoising diffusion probabilistic models. In *2023 IEEE 17th International Conference on Automatic Face and Gesture Recognition (FG)*, pages 1–7. IEEE, 2023. 2

[27] Francesco Nicolo and Natalia A Schmid. Long range cross-spectral face recognition: matching swir against visible light images. *IEEE Transactions on Information Forensics and Security*, 7(6):1717–1726, 2012. 2

[28] Luis Perez and Jason Wang. The effectiveness of data augmentation in image classification using deep learning. *arXiv preprint arXiv:1712.04621*, 2017. 2

[29] Neehar Peri, Joshua Gleason, Carlos D Castillo, Thirimachos Bourlai, Vishal M Patel, and Rama Chellappa. A synthesis-based approach for thermal-to-visible face verification. In *2021 16th IEEE international conference on automatic face and gesture recognition (FG 2021)*, pages 01–08. IEEE, 2021. 3

[30] Domenick Poster, Matthew Thielke, Robert Nguyen, Srinivasan Rajaraman, Xing Di, Cedric Nimpa Fondje, Vishal M Patel, Nathaniel J Short, Benjamin S Riggan, Nasser M Nasrabadi, et al. A large-scale, time-synchronized visible and thermal face dataset. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 1559–1568, 2021. 1

[31] Rajeev Ranjan, Ankan Bansal, Hongyu Xu, Swami Sankaranarayanan, Jun-Cheng Chen, Carlos D Castillo, and Rama Chellappa. Crystal loss and quality pooling for unconstrained face verification and recognition. *arXiv preprint arXiv:1804.01159*, 2018. 3, 4, 7

[32] Rajeev Ranjan, Ankan Bansal, Jingxiao Zheng, Hongyu Xu, Joshua Gleason, Boyu Lu, Anirudh Nanduri, Jun-Cheng Chen, Carlos D Castillo, and Rama Chellappa. A fast and accurate system for face detection, identification, and verification. *IEEE Transactions on Biometrics, Behavior, and Identity Science*, 1(2):82–96, 2019. 3

[33] Rajeev Ranjan, Swami Sankaranarayanan, Carlos D Castillo, and Rama Chellappa. An all-in-one convolutional neural network for face analysis. In *2017 12th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2017)*, pages 17–24. IEEE, 2017. 3

[34] Kuniaki Saito, Donghyun Kim, Stan Sclaroff, Trevor Darrell, and Kate Saenko. Semi-supervised domain adaptation via minimax entropy. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 8050–8058, 2019. 3, 4

[35] Diego A Socolinsky, Lawrence B Wolff, Joshua D Neuheisel, and Christopher K Eveland. Illumination invariant face recognition using thermal infrared imagery. In *Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. CVPR 2001*, volume 1, pages I–I. IEEE, 2001. 1

[36] Lingxiao Song, Man Zhang, Xiang Wu, and Ran He. Adversarial discriminative heterogeneous face recognition. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32, 2018. 2

[37] Shangfei Wang, Zhilei Liu, Siliang Lv, Yanpeng Lv, Guobing Wu, Peng Peng, Fei Chen, and Xufa Wang. A natural visible and infrared facial expression database for expression recognition and emotion inference. *IEEE Transactions on Multimedia*, 12(7):682–691, 2010. 1

[38] Xiang Wang, Kai Wang, and Shiguo Lian. A survey on face data augmentation. *arXiv preprint arXiv:1904.11685*, 2019. 2

[39] Shuo Yang, Ping Luo, Chen-Change Loy, and Xiaoou Tang. Wider face: A face detection benchmark. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5525–5533, 2016. 3