IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 19 May 2023, accepted 13 June 2023, date of publication 27 June 2023, date of current version 6 July 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3289923

RESEARCH ARTICLE

Open-Set Profile-to-Frontal Face Recognition on a Very Limited Dataset

MUHAMMAD DJAMALUDDIN, (Graduate Student Member, IEEE), RINALDI MUNIR, NUGRAHA PRIYA UTAMA, AND ACHMAD IMAM KISTIJANTORO, (Member, IEEE)

Bandung Institute of Technology, Bandung 40132, Indonesia

Corresponding author: Muhammad Djamaluddin (33218027@mahasiswa.itb.ac.id)

The work of Muhammad Djamaluddin was supported by the Indonesia Endowment Fund for Education (LPDP).

ABSTRACT Open-set face recognition on a small dataset with limited image samples per individual poses a significant challenge and is a topic of active research. Therefore, this study investigated the problems of open-set face verification and face identification on a dataset known as the ITB Frontal Profile Limited Dataset (IFPLD), which included only one frontal and one profile image per individual. Various training procedures were used to obtain a more appropriate network embedding for feature representation on the dataset. Transfer learning was employed to improve the performance of the models by fine-tuning the networks using a dataset with properties similar to those of the IFPLD. The results showed that the SimCLR method generated the optimal network embedding for face verification on the Siamese network. The prototypical network with an N-way-k-shot learning scenario where k-1 came from data augmentation outperformed the Siamese network for face identification by a maximum 17.0% accuracy improvement. The transformation from 1-shot learning to k-shot learning is critical for achieving high performance.

INDEX TERMS Open set face recognition, Siamese network, SimCLR, ArcFace loss, prototypical network.

I. INTRODUCTION

Deep learning-based face recognition systems have gained a significant amount of praise owing to their high accuracy on several near frontal face datasets such as Labeled Faces In The Wild (LFW) [1], VGGFace [2], and MegaFace [3]. These public datasets share general characteristics such as near frontal poses, less than 45° yaw, individuals of European American or African descent, and no head cover attributes. It is well-known that many deep face recognition systems perform poorly when a dataset does not exhibit these characteristics. Unfortunately, most faces captured in the wild do not have this feature, leading to inaccuracies in the performance of face recognition systems when applied to real-life scenarios.

This study primarily focuses on face recognition problems, including face verification and face identification, when the system is trained using a small dataset with limited accessible poses and a restricted number of samples per person. Despite the advent of more extensive databases, many public institutions maintain this limited database, including a limited number of ID photographs for each individual and their personal information. The authors focused on frontal-to-profile face recognition, where models can choose the appropriate frontal image based on query data in profile images or determine whether a pair of frontal and profile faces belong to the same individual. However, a profile face is problematic during face recognition, primarily because of partial occlusions in the facial landmarks caused by yaw angle rotation.

A unique face dataset called the ITB Frontal Profile Limited Dataset (IFPLD) was created to address this issue, as shown in Figure 1a. This dataset is distinct from the others because each individual is represented by only two face images, where the first is for the frontal face and the second is for the profile face with a yaw angle of 90°. In addition to its size limitation, this database differs from others in that some individuals wear head coverings, which obscure some of their features. These constraints, including the number of

The associate editor coordinating the review of this manuscript and approving it for publication was Xianzhi Wang.

photos per individual, permissible posture types, and distinct attributes of each face, are significant obstacles to the face recognition process.

Using IFPLD, multiple models of deep learning systems based on Siamese network architecture for face verification were evaluated. Several models were trained using various methodologies to generate improved network embedding vectors. The ArcFace loss function is considered state-of-the-art for face recognition in some datasets [4]. According to this study, the Siamese network with a CNN network that has been fine-tuned with ArcFace loss and combined with other learning methods provides some of the top results on the IFPLD dataset. Additionally, the self-supervised learning technique called SimCLR [5] is an alternative superior way for generating an embedding network with a competitive face verification performance.

This study explores two separate settings for face identification. The first strategy employed Siamese network models specifically trained for face verification to accomplish an N-way-1-shot evaluation task for face identification in an open test set. The second option is to train a prototypical network [6] on the IFPLD training dataset using an N-way-1-shot learning scenario with modifications to the original algorithm to overcome the sample per-class limitation. Table 1 lists the symbols and acronyms used in this study as references.

The contributions of this research can be summarized as follows:

1. The dataset addresses the open-set face recognition issue, with a small number of samples per person and a small number of poses called the ITB Frontal Profile Limited dataset (IFPLD).

2. Comparative research and step-by-step techniques for generating an improved face image representation vector for open-set face verification, such as metric learning through Contrastive Loss or ArcFace loss and a self-supervised learning technique called SimCLR. The study showed that the Network Embedding produced by SimCLR outperforms metric learning and is appropriate for face verification using the Siamese network on the IFPLD.

3. A method for training a prototypical network for IFPLD that utilizes augmentation to overcome the support and query set limits during training and testing, thereby achieving optimal performance for face identification problems on a relatively small dataset. The research shows that the three augmented data of frontal and profile faces provide the optimal balance between processing speed and accuracy.

II. RELATED WORKS

Recent studies have extensively explored CNN-based feature extraction for face recognition. This is primarily due to the success of AlexNet, GoogleNet, ResNet, and other CNN-based image classification models that defeated rivals in ILSVR. This scenario causes deep learning-based solutions to gain momentum in all fields of computer vision. One of the state-of-the-art end-to-end deep learning-based face recognition systems, FaceNet [7], achieved an accuracy of 99.63% on the LFW dataset. However, on the more challenging Megaface dataset for the face identification task, its accuracy drops to 70.49%.

Triplet loss is a major contributor to the performance of FaceNet. It produces feature embeddings that are well-suited for face classification by reducing the intra-class and increasing inter-class variances. Researchers have produced several variants of loss functions to enhance the discriminative power of deep learning features, including Marginal Loss [8], SphereFace loss [9], ArcFace loss [4], and CosFace loss [10]. For example, Marginal Loss showed a 0.6% improvement in face verification performance on the LFW dataset and a 1.7% improvement on the YFT dataset, achieving 99.48% and 95.68% accuracy, respectively, compared to the default softmax loss.

Several studies have focused on developing deep-learning techniques for face frontalization to improve face recognition. The aim is to standardize a profile face to its corresponding frontal face. Cao et al. [11] developed a DREAM block to generate a frontal embedding vector from a profile face, which can be reconstructed into a frontal face image using a GAN model called Plug and Play Network [12]. On the IJB-A dataset, this approach achieved 94.60% accuracy; however, on the MS-Celeb-1M dataset, it achieved 94.40% accuracy.

Fariborz et al. introduced Coupled Conditional GAN [13], a technique for latent space face verification that employed twin GAN networks composed of Generator, Discriminator,

TABLE 1. Nomenclature.

Symbols	Description
IFPLD	ITB Frontal Profile Limited Dataset
CFP	Celebrity Frontal Profile Dataset
ResNet-18_IN	ResNet-18 pre-trained with ImageNet 1000 classes
ResNet-18_IN_CFP*	ResNet-18_IN fine-tuned with the CFP dataset
ResNet-18_IN_CFP* + M	ResNet-18_IN_CFP* trained with M scenario

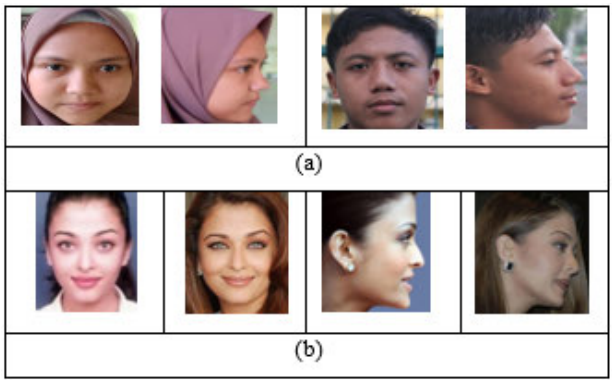


FIGURE 1. (a) Sample of IFPLD dataset where one person has one frontal face and one profile face and (b) Sample of CFP dataset where one person has ten frontal faces and four profile faces.

and Reconstruction networks to verify both frontal and profile faces. The Generator network, a UNet auto-encoder network, produces frontal and profile embedding vectors in a common embedding space after training with Contrastive Loss [14]. For the IJB-A dataset, this approach achieved a verification rate of 96.6%, and for CFP, it achieved a verification rate of 96.2%. In contrast to IFPLD, the profile images in the MS-Celeb-1M and IJB-A datasets rarely had a yaw angle greater than 45° and did not feature any head covers, making face frontalization more efficient. Another approach, Side Face Correction GAN [15], utilizes a GAN model to correct non-frontal face images to obtain frontal face images and then extracts facial features for face recognition. This method comprises two parts: generation and discrimination modules.

Since the feature-based model era, reinforcement learning (RL) has been used in face recognition to adapt to various variables and settings, enabling more robust and efficient techniques. To overcome the imbalance problem of the face dataset, the fair loss algorithm [16] employs Q Learning to modify the margin value. This approach achieved 99.57% accuracy on the LFW dataset, second to FaceNet among all investigated methods. Fast-FAR [17] used RL to improve the facial recognition speed without sacrificing accuracy by adjusting the depth of the inference layer based on a decision policy. Fast-FAR reduced the inference time by 14.22% for LFW and 7.84% for CFP.

In addition to IFPLD, only a few well-known public datasets contain frontal and profile faces that meet strict criteria, including a yaw angle near 90°. Among these is the Celebrity Frontal Profile (CFP) dataset [18], which serves as a benchmark for several state-of-the-art face recognition techniques. It contains 500 celebrities, each with ten frontal and four profile face images, for a total of 7000 face images. In contrast to CFP, IFPLD has only one image for each frontal and profile face and contains some faces with head covers not found in CFP, creating additional complexity. Figure 1 shows a comparison between the IFPLD and CFP.

The CFP dataset comprises two testing scenarios: Frontal to Profile (CFP-FP) and Frontal to Frontal (CFP-FF). The original implementation in this study showed that the face verification accuracy in the CFP-FF scenario was 11% higher than that in the CFP-FP scenario. This result showed a significant challenge in identifying profile-to-frontal faces compared with frontal-to-frontal scenarios. More recent research has improved the performance of the CFP-FP face verification. The ResNet model trained with the ArcFace Loss function [4] reported 95.56% accuracy for the CFP-FP scenario. This result represents a nearly 9% improvement over the first deep feature implementation, which had an accuracy of 84.91%.

Significant research has been conducted on few-shot learning for image categorization, but only a few studies have explored its application to face recognition problems. As previously mentioned, most face recognition research relies on a large face dataset, whereas few-shot learning focuses primarily on limited training data. The phrase ''N-way-k-shot learning'' refers to the use of k samples of data for each of the N classes of the subject, either as a query or support dataset. When k is 1, there is only one sample per class, which is suitable for the face recognition problem of IFPLD.

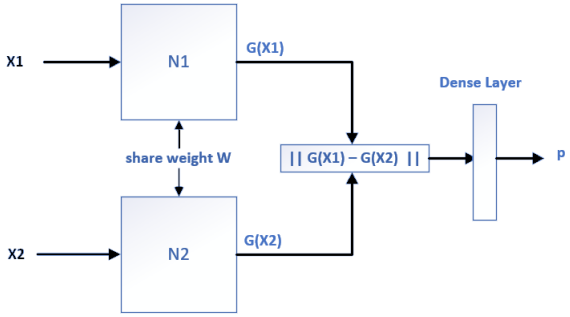


FIGURE 2. Siamese Network for face verification.

There are several few-shot-learning techniques based on a data-level or parameter-level approach. For example, Matching networks generate a weighted sum for each support image label as the prediction, using an attention mechanism to weight each support sample image based on its relevance to the query image [19]. On the other hand, the Prototypical network is one of the architectures for few-shot classification that maps query data into the closest point called prototype, representing a region of a particular class [6].

III. METHODS

The face verification problem is often solved using the Siamese network architecture, which compares two inputs and generates an output that indicates their similarity. The Siamese network consists of two identical base networks that share weights and process different images. Furthermore, a probability value was obtained by comparing the results of each network. Figure 2 shows the architecture of the Siamese network used for face verification, where N1 and N2 are twin CNN models with shared weights, and the output p represents the probability that the input pair (X1, X2) belongs to the same individual.

The use of the Siamese network for face verification and face identification as a one-shot classification problem was inspired by Koch et.al [20]. The architecture of the base network used in Koch's Siamese network served as a baseline model, referred to as Koch Model. Another baseline model was ResNet-18_IN, ResNet-18 pre-trained with the ImageNet dataset, a commonly used CNN base network for image classification. ResNet-18 was selected over other ResNet families because testing on IFPLD showed that a deeper version did not improve the performance.

Instead of using a Siamese network for face identification, a Prototypical network was developed, whose base network had been previously trained using a Siamese network. These base networks were trained using methods that yielded

the highest accuracy. To handle limited support and query images, the 1-shot learning problem was modified to K-shot learning, where $K > 1$. The K number of samples comprised the original frontal or profile image with $K - 1$ data augmentation. Five image operations were employed to create these additional samples: blurring, rotation, brightness adjustment, horizontal flipping, and translation. The impact and number of data augmentations required for the Prototypical networks to operate at their optimum level were then analyzed.

During training, the Adam optimizer was selected as the optimization method after comparing it with SGD (with momentum) and RMSProp. The step-wise decay approach was used to regulate the learning rate, which was adjusted to a specific value after a particular epoch. The Early Stopping mechanism was also used to prevent overtraining, which could lead to overfitting. The highest accuracy model for the validation data was saved for further use.

A. DATASET PREPARATION

IFPLD included 475 subjects, each with a frontal and profile image as shown in Figure (Figure 1a), for a total of 950 face images. Face images were cropped after detecting them with the MTCNN, removing irrelevant parts, and producing face image data. MTCNN is comprised of three-stage multi-task CNNs to propose region candidates, refine the candidate selections, and output bounded rectangles to represent facial regions and their facial landmarks.

The dataset was divided into a training and validation set of 380 subjects and a testing set of 95 subjects. This setup was considered ideal for ''Open set face recognition'' because the identity in the training data differed from the testing. Therefore, facial data in the testing stage were absent from the training data, and subject identities were different for face verification or identification purposes.

To augment the limited data in the IFPLD, five pieces of data were generated for each face image using translation, horizontal flip, brightness modification, rotation, or blurring operations. Consequently, each subject was represented by six frontal and six profile face images. The translation operation shifted an image by (20,20) pixels to the right and down, respectively. The rotation operation rotated an image by 10° in the clockwise direction, while the brightness modification operation modified the image's brightness and contrast by 0.7 gain and 0.2 bias coefficient. During the blurring process, a (10,10) kernel was applied to an image.

Two testing data preparation schemes were used, depending on the type of task conducted.

1. Face Verification

The testing data were randomly generated by selecting two pairs of faces from identical or different individuals, including their augmented versions from the original 95 individuals in the test class. Given a frontal image, a random number determines whether a profile image is selected from the same or a different class than the frontal image. This setup mirrored how training data were obtained, and there were 1140 testing data (95 × 12 faces per individual). Since 95 × 6 frontal faces must be compared, the length of the face verification test dataset is 570.

2. Face Identification

The testing data were randomly generated from the original frontal and profile face images of 95 individuals in the test class. The goal was to determine the matching frontal face from a subset of N individuals, where N denotes a subset of 95 testing profiles, given a sample of the profile face image. The generation of testing data per episode is outlined in greater depth in each approach employed, such as the Siamese Network or the Prototypical Network. Furthermore, the effect of online image augmentation was investigated in a Prototypical network to establish optimum performance.

B. NETWORK EMBEDDING

The Feature representation of a face image plays a significant role in the face identification or verification process. Typically, facial data features are generated by network embedding from pre-trained backbones CNN architectures such as ResNet, VggNet, or FaceNet. These architectures are typically pre-trained using the ImageNet dataset containing 1000 classes. In a transfer learning scenario, a model trained using data with characteristics similar to the problem data provided better generalization.

The network embedding baseline consisted of the Koch and ResNet-18_IN models. Specifically, the Koch Model was a CNN network architecture developed by Koch et al. [20] for addressing the one-shot learning problem using the Siamese network. It performed well on datasets such as Omniglot and MNIST. The second baseline was ResNet-18 trained using ImageNet 1000 classes and abbreviated as ResNet-18_IN for brevity.

The CFP dataset [18] was similar to IFPLD but had more individuals and faces per person, and each person in the dataset had ten frontal faces and four profile faces, compared to just one frontal face and one profile face per person in IFPLD, as shown in Figure 1.b). An ImageNet pre-trained ResNet-18 model was fine-tuned with the CFP dataset using two different training scenarios: multiclass and binary classification. In multiclass classification, the ResNet-18_IN model was trained to recognize an identity from 500 individuals. In binary classification, the ResNet-18_IN model determines whether a face image is a frontal (class 0) or profile (class 1) face. After multiclass classification training, this fine-tuned ResNet-18_IN model was called ResNet-18_IN_CFP1. After binary classification, it was named ResNet-18_IN_CFP2.

ResNet-18_IN was also fine-tuned using the ArcFace loss function in the multiclass classification mode and named ResNet-18_IN_CFP3. The Additive Angular Margin Loss, known as ArcFace Loss [4], is a loss function and process for generating highly discriminative features for face recognition considered state-of-the-art in specific face datasets. It is a modified softmax loss in angular space designed to circumvent the intra-class classification limitation of softmax loss.

ArcFace loss function was defined using the equation below.

$$L = -\frac{1}{N}\sum_{i}^{N}\log\frac{e^{s(\cos(\theta_{y_i}+m))}}{e^{s(\cos(\theta_{y_i}+m))}+\sum_{j=1,j\notin y_i}^{n}e^{s\cos\theta_j}} \quad (1)$$

The hyperparameters m and s, indicating the additive angular margin and feature scale, respectively, were set to 0.35 and 30 for the best performance. The parameter m indicates the distance between classes, while s modify the scale of the logits. All three CFP fined-tuned ResNet-18_IN models are called ResNet-18_IN_CFP* for easier reference.

While classification directly calculates and optimizes the classification loss, the metric learning approach focuses on optimizing the embedding loss before the classification layer; hence, the objective is to create a distinctive and unique embedding feature that accurately represents the input image.

The Siamese network is a commonly used architecture for metric learning because it allows for producing and comparing a set of embedding features generated by a twin network. Two common scenarios for training Siamese networks are training with Contrastive Loss [14] and Triple Loss [7]. The contrastive loss function is defined as follows:

$$L = \frac{1}{2}((1-Y)D_w^2 + Y\max(0,\alpha - D_w)^2) \quad (2)$$

where D_w is a distance function such as the Euclidean distance, Y is the output label (0 or 1), and α denotes a margin value > 0 indicating the radius in the embedding space. Two feature pairs of the same class contribute only to the loss function if distance D_w is within the margin value.

Triplet Loss is defined as:

$$L = \sum_{i}^{N}[\|f(x_i^a) - f(x_i^p)\|_2^2 - \|f(x_i^a) - f(x_i^n)\|_2^2 + \alpha]_+ \quad (3)$$

The variables x_i^a, x_i^p, and x_i^n represent the anchor, positive, and negative data, respectively, and α is the margin error between positive and negative pairs.

When training with Triplet Loss, the selection of triplet scenarios can significantly affect model performance. To train the IFPLD on the Siamese network based on the Koch and ResNet-18_IN_CFP* models, a simple random selection procedure was used as follows:

1. Divide 475 classes into 380 classes for training, 95 classes for testing.
2. For each class, frontal and profile face data were collected, including augmentation, which produced 2280 embedding vectors for each type.
3. Specify the maximum number of datasets.
4. A frontal face image was selected as anchor data for each triplet. Furthermore, a profile face image with the same class as the anchor was randomly chosen as the positive data, and any random profile face with a class label different from the anchor was selected as the negative data.

Another approach for training network embedding is Self-Supervised Learning (SSL). In SSL, unlabeled data are input to a network that finds its feature representation based on a particular loss function. Subsequently, the generated network embedding is used for downstream-specific tasks such as classification and segmentation.

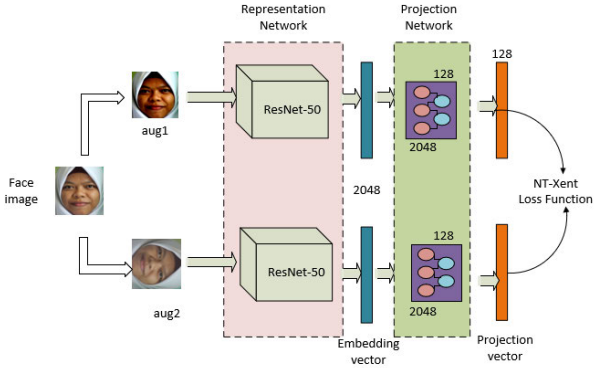


FIGURE 3. SimCLR training implementation in the IFPLD dataset.

A recently proposed SSL method is a Contrastive Learning method called SimCLR, a state-of-the-art method successfully implemented for various datasets, including CIFAR10, CIFAR100, and SUN397, none of which is a face dataset [5]. SimCLR was designed to maximize the feature similarity between two augmented images. This was achieved through learning with Contrastive Loss in the latent space. The loss function used is known as the Normalized Temperature-scaled Cross Entropy (NT-Xent).

Figure 3 shows the implementation of SimCLR training on IFPLD. To train with SimCLR, a dataset was generated by creating two different augmented versions of a face image in IFPLD, whether profile or frontal face. These augmentations were generated by passing an image through a series of random image transformations such as cropping, horizontal flip, rotation and resizing, changing brightness, contrast, and saturation (random jitter). Furthermore, the two images were inputted into a twin ResNet-18 network. Each embedding vector was projected onto a higher dimensional space by a network projector consisting of fully connected layers, producing a 128-d projection vector.

The NT-Xent loss function measures the similarity between pairs of projection vectors and is defined as follows:

$$l_{i,j} = -log(\frac{e^{(\frac{sim(z_i,z_j)}{\tau})}}{\sum_{k=1}^{2N}1_{[k\neq i]}e^{(\frac{sim(z_i,z_k)}{\tau})}}) \quad (4)$$

For each pair of augmentation data (z_1, z_2) considered as positive data, there were N-1 negative data, where N represents the batch size. Temperature τ has a value ranging from 0 to 1.

C. FACE VERIFICATION ON IFPLD DATASET

The Siamese network with a classification layer, as shown in Figure 2, produces the probability of input pairs from the same individual. If the probability $p \geq 0.5$, the input pair is from the same person, otherwise, they are different individuals.

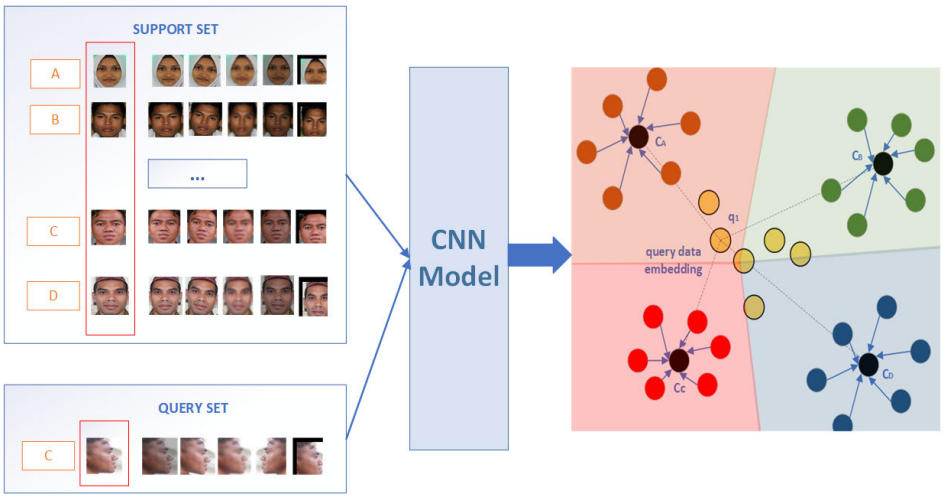


FIGURE 4. Prototypical Network for face identification on IFPLD dataset.

TABLE 2. Siamese network configuration for several scenarios.

Siamese Network based Model	Classification Layer Configuration
Koch	1. FC layer (in = 4096, out = 1)
ResNet-18	1. FC layer (in = 4096, out = 1)
ResNet-18_IN_CFP*	1. FC Layer (in = 2048, out = 128) 2. FC Layer (in = 128, out = 1)
ResNet-18_IN_CFP* Contrastive Loss and ResNet-18_IN_CFP* Triplet Loss	1. FC Layer (in = 2048, out = 512) 2. FC Layer (in =512, out = 128) 3. Sigmoid 4. Dropout 5. FC Layer (in =128, out = 1)
ResNet-18_IN_CFP* SimCLR	1. FC Layer (in = 2048, out = 512) 2. Sigmoid 3. FC Layer (in =512, out = 128) 4. Sigmoid 5. FC Layer (in =128, out = 1

Several base models of Siamese network have been investigated. These models were trained using the methods described in the previous section, such as the Koch Model based on the architecture proposed by Koch et al. [20], and the ResNet-18_IN and ResNet-18_IN_CFP* models described above.

Table 2 lists the configuration of the classification layers for each models.

D. FACE IDENTIFICATION ON IFPLD DATASET

Few-shot-learning is a machine learning problem that arises when the available data, denoted as E, consists of a limited number of labeled samples for a given Task T. In the case of IFPLD, each person was represented by one frontal and one profile face image, making it a suitable setup for implementing N-way-1-shot classification where N is the number of individuals to be compared that could be selected as needed. Instead of training a network using the entire concept of Few-Shot-Learning, a Siamese network for face verification can be used to identify an individual's identity because its output is the probability that the input pair comes from the same individual.

For example, from 95 individuals in the test class, 20 randomly selected individuals were denoted as N. One person was then randomly chosen from the 20 individuals to represent the class for the query data. Notably, the query data represented the profile face of the selected class. The query data were then compared with the frontal faces of the remaining 20 individuals to determine the highest probability of a match. The identity of the individual whose frontal face was used as the input was revealed by the highest probability. This process is repeated for a specified number of episodes or trials. Koch et al. [20] used 400 episodes with N=20 support classes for an N-way-1-shot evaluation in evaluating a Siamese network. In the experiment, the episode number was set to 400 or 1000, and N was varied with values of 5, 20, and 40.

Suppose a query profile face x belongs to one of the individuals, d in class C, where $d \in C$. Each person in C has a collection of frontal face datasets x_c or $\{x_i\}_{i=1}^{i=C}$ called support sets. To determine the identity of d based on similarity, a query is performed for pairs (x, x_i) where $i = 1, \ldots, C$. In the case of a Siamese network for binary classification, the identity of individual d was revealed by class i with the highest probability value p:

$$d = \arg\max_{c}(p^{(c)}) \tag{5}$$

It is preferable to train a network with the N-way-k-shot learning principle to directly learn class representations or distance measures for face identification rather than relying on the 1-shot-learning evaluation of the Siamese network [21]. A popular method for few-shot-learning is the Prototypical network, which has been used to achieve state-of-the-art results on standard few-shot-learning-oriented datasets such as Omniglot and Mini ImageNet [6].

The Prototypical network approach involves constructing prototypes and assessing the similarity between the query data and prototypes for classification purposes. The mean of a collection of embedding vectors of the same class in the CNN's feature space was recorded to determine the prototype point. The class to which the query data belonged was identified by measuring the distance between the query data and the class prototypes in the feature space.

Assuming that a CNN model is defined as an embedding function $f_\phi : R^D \rightarrow R^M$ where ϕ is a learnable parameter, D is the dimension of the input, and M denotes the dimension of the embedding vector, the prototype for class k or c_k is defined as follows:

$$c_k = \frac{1}{|S_k|} \sum_{(x_i, y_i) \in S_k} f_\phi(x_i) \tag{6}$$

Let S be the support set of N labeled examples, defined as $S = \{(x_1, y_1), (x_2, y_2), \ldots, (x_N, y_N)\}$, where $x_i \in R^D$ is the D-dimensional embedding vector.

To implement Prototypical networks in IFPLD, a CNN model was trained using methods to generate embedding vectors of Support and Query sets. The base model that exhibited the best performance in face recognition using the Siamese network was selected. Separating training and testing data into Support and Query sets is crucial for few-shot learning. N classes were randomly selected from 95 test samples, and up to five augmented images were generated on the fly for each frontal face, bringing the total number of images per class to six. Therefore, when N = 5, the Support set data contained 30 images. For each class in N, the profile face matching that class is selected as the query set data.

As shown in Figure 4, each episode included N_C randomly selected classes from the 95 available test classes, divided into Support and Query sets, each consisting of K images. For the Support set, the first image was the original frontal image, and the remaining K-1 images were generated from data augmentation. Query data were selected by matching N support classes with appropriate profile images and K-1 data augmentation.

The Support and Query sets are fed into a CNN model to produce embedding vectors in the same feature space. The prototypes for each class in the support set were then calculated as shown in Figure 4, where C_A and C_B represent class A and class B, respectively.

The yellow embedding query data in the figure minimize the distance between each sample and its respective class prototype. The loss function is defined as $J(\phi) = -log(p_\phi(y = k|x))$.

$$p_\phi(y = k|x) = \frac{e^{-d(f_\phi(x), c_k)}}{\sum_{k'} e^{-d(f_\phi(x), c_{k'})}} \tag{7}$$

The distance function between the embedding vector of the query data and prototype c_k is denoted as d. The class of query data is determined by the prototype closest to the embedding vector.

Algorithm 1 presents the training algorithm for the Prototypical network on IFPLD for embedding network f_ϕ given a training episode N_{eps}. Four distinct training scenarios were analyzed, each focusing on the number of instances of N_{eps} and N_C combinations to be used during a single training session. N_{eps} represents the number of episodes for training few-shot-learning, and values of 400 and 1000 were selected, as in the Siamese network evaluation. For a given

TABLE 3. Accuracy of siamese network models for face verification.

Siamese Network based Model	Accuracy
Koch	0.79
ResNet-18_IN	0.90
ResNet-18_IN_CFP1	0.89
ResNet-18_IN_CFP2	0.84
ResNet-18_IN_CFP3	0.92
ResNet-18_IN_CFP1 + Contrastive Loss	0.86
ResNet-18_IN_CFP2 + Contrastive Loss	0.85
ResNet-18_IN_CFP3 + Contrastive Loss	0.82
ResNet-18_IN_CFP1 + Triplet Loss	0.90
ResNet-18_IN_CFP2 + Triplet Loss	0.84
ResNet-18_IN_CFP3 + Triplet Loss	0.92
ResNet-18_IN_CFP1 + SimCLR	**0.93**
ResNet-18_IN_CFP2 + SimCLR	0.89
ResNet-18_IN_CFP3 + SimCLR	0.91

N_{eps} value, N_C was selected from the 5, 20, 40, and 95 values. For the Support set S_i and Query set Q_i, the total data per class was six, with one image representing the original frontal/profile and five representing augmented data. To facilitate the training, N-way-1-shot learning was transformed into N-way-6-shot learning. Model 1 corresponded to the training episode model when N = 5, Model 2 had N = 20, Model 3 corresponded to N = 40, and Model 4 corresponded to N = 95, equivalent to the number of IFPLD test classes.

All training models were saved for testing the unseen 95 test classes. During testing, the effect of various values of N and K of test data on the network performance of each model were examined. Similar procedures as in the Siamese network evaluation for face identification were used, where the number of episodes N_{eps} was (400, 1000), episode class number N_C was (5,20,40), and Support set S_i or Query set (Q_i) was $(1, \cdots, 6)$. Algorithm 2 shows the complete steps for testing the IFPLD with a trained Prototypical network. The input included the trained model f_ϕ, data testing D, N_Q and N_{eps}.

IV. RESULT AND DISCUSSION

A. FACE VERIFICATION

The experimental results for all Siamese network configurations for the open-set face verification problem on IFPLD, measured in terms of accuracy and F1 score, are listed in Tables 3 and 4, respectively. A Siamese network was fed with a randomly selected list of pairs of frontal and profile faces from individuals that were not included in the training dataset.

Table 3 shows the highest accuracy value (93%) marked in red. This value was obtained by the model with the backbone network ResNet-18_IN_CFP1 and then trained using the SimCLR method. This was followed by the ResNet-18_IN_CFP3 and ResNet-18_IN_CFP3 + Triplet Loss networks.

Among the models, the baseline Koch model Siamese network scored the lowest at 87 %. However, the baseline ResNet-18_IN Siamese network performed better than the Koch and some ResNet-18_IN_CFP* models, including the one trained with Contrastive loss. The base network configuration played a significant role in the performance,

Algorithm 1 The following are the training steps for prototypical networks used in face identification on IFPLD. Let N = the number of the training set samples, K is the number of training classes (380 training classes for IFPLD dataset), $N_C \geq K$ is the number of class per episode, N_S is the number of support example per class, N_Q is the number of query example per-class. N_{eps} refers to the number of training episodes. RandomSample(S, N) is a set of randomly chosen N elements from S without replacement. GenerateDataAug(S, A) is an operation to generate several data augmentations for each original IFPLD image in S. ComputeLoss(k) calculates the loss function of model k. UpdateModel(f_ϕ,J) is used to update model f_ϕ by the value of loss function J

Input : Initial model f_ϕ, Training set D = $\{(x_1, y_1), \cdot, (x_N, y_N)\}$, where each $y_i \in \{1, \cdots, K\}$, N_{eps}
Output: Models of all N_C option and selected N_{eps}

$N_S \leftarrow 1$
$N_Q \leftarrow N_S$
$N_C \leftarrow \{5, 20, 40, 95\}$; /* Set no class per-episode */
$N_{Aug} \leftarrow 5$; /* Set no of augmented data */
for *i in* N_C **do**
  for *e in {1,⋯,*N_{eps}*}* **do**
    $V \leftarrow RandomSamples(\{1, \cdots, K\}, i)$; /* Select random i classes for episode */
    for *j in {1,⋯,i}* **do**
      $S_j \leftarrow RandomSamples(D_{V_j}, N_S)$; /* Select random support data for class j */
      $Q_j \leftarrow RandomSamples(D_{V_j} \backslash S_j, N_Q)$; /* Select random query data for class j */
      $S_j \leftarrow S_j + GenerateDataAug(S_j, N_{Aug})$; /* Add data augmentation to S_j */
      $Q_j \leftarrow Q_j + GenerateDataAug(Q_j, N_{Aug})$; /* Add data augmentation to Q_j */
      $c_j = \frac{1}{|S_i|} \sum_{(x_i, y_i) \in S_j} f_\phi(x_i)$; /* Calculate prototype for class j */
      $d_j = d(f_\phi(Q_j), c_j)$; /* Determine distance of point Q_i to c_k */
    end for
    J = ComputeLoss(i)
    UpdateModel(f_ϕ, J)
  end for
end for

Algorithm 2 Testing steps for the prototypical network for face identification on IFPLD Dataset are outlined below. Let N = the number of the testing set samples, K is the number of testing classes, $N_C \geq K$ is the number of class per episode, N_S is the number of support example per class, N_Q is the number of query example per-class: $[2, \cdots, 6]$. RandomSample(S, N) is a set of randomly chosen N elements from S without replacement. GenerateDataAug(S, A) is an operation to generate Several data augmentations for each original IFPLD image in S. ComputeAccuracy(i, d_k) measures the accuracy based on distance query data of class i to c_k

Input : Trained model f_ϕ, Testing set D = $\{(x_1, y_1), \cdot, (x_N, y_N)\}$, where each $y_i \in \{1, \cdots, K\}$, N_Q and N_{eps}
Output: Performance Accuracy P

$N_S \leftarrow 6$
$N_C \leftarrow \{5, 20, 40\}$
for *e in {1,⋯,*N_{eps}*}* **do**
  $V \leftarrow RandomSamples(\{1, \cdots, K\}, N_C)$
  for *i in {1,⋯,*N_C*}* **do**
    $S_i \leftarrow RandomSamples(D_{V_i}, 1)$
    $Q_i \leftarrow RandomSamples(D_{V_i} \backslash S_i, 1)$
    $S_i \leftarrow S_i + GenerateDataAug(S_i, N_S - 1)$
    $Q_i \leftarrow Q_i + GenerateDataAug(Q_i, N_Q - 1)$
    $c_i = \frac{1}{|S_i|} \sum_{(x_i, y_i) \in S_i} f_\phi(x_i)$
    $d_i = d(f_\phi(Q_i), c_i)$
    P = ComputeAccuracy(d_k)
  end for
end for

and the ResNet-18_IN model performed better than Koch Model. ResNet-18 has deeper layers than the Koch model, but the number of trained parameters is much smaller. The number of parameters for the Koch-based Siamese network

TABLE 4. F1 Score of siamese network models for face verification.

Siamese Network based Model	Precision		Recall		f1-score	
	class 0	class 1	class 0	class 1	class 0	class 1
Koch	0.82	0.76	0.72	0.85	0.77	0.80
ResNet-18_IN	0.92	0.89	0.89	0.92	0.90	0.90
ResNet-18_IN_CFP1	0.89	0.90	0.90	0.88	0.89	0.89
ResNet-18_IN_CFP2	0.93	0.79	0.74	0.93	0.83	0.86
ResNet-18_IN_CFP3	0.95	0.91	0.91	0.95	**0.93**	**0.93**
ResNet-18_IN_CFP1 + Contrastive Loss	0.86	0.86	0.86	0.86	0.86	0.86
ResNet-18_IN_CFP2 + Contrastive Loss	0.86	0.86	0.86	0.86	0.86	0.86
ResNet-18_IN_CFP3 + Contrastive Loss	0.82	0.81	0.80	0.83	0.81	0.82
Resnet-18_IN_CFP1 + Triplet Loss	0.81	0.93	0.94	0.78	0.87	0.85
Resnet-18_IN_CFP2 + Triplet Loss	0.65	**0.94**	**0.96**	0.52	0.77	0.67
Resnet-18_IN_CFP3 + Triplet Loss	0.95	0.91	0.91	0.95	**0.93**	**0.93**
ResNet-18_IN_CFP1 + SimCLR	**0.96**	0.86	0.90	**0.96**	**0.93**	**0.93**
ResNet-18_IN_CFP2 + SimCLR	0.84	0.65	0.54	0.89	0.66	0.70
ResNet-18_IN_CFP3 + SimCLR	0.94	0.88	0.86	0.95	0.90	0.91

was 38 million, while ResNet-18 was only 11.7 million or 70% less.

As shown in Tables 3 and 4, the ResNet18_IN_CFP*-based Siamese network achieved higher performance metrics for face verification than the baseline Koch model, but in some scenarios, it was not better than ResNet-18_IN. In terms of accuracy, the network with the highest performance, such as the ResNet-18_IN_CFP1 + SimCLR model, was clearly superior to the baseline. The accuracy increased by 13% compared with the Koch-based model and by 2% compared with the ResNet-18_IN. This significant improvement demonstrated that training networks using SimCLR methods was superior to other training methods.

Moreover, the Koch model was not pre-trained, highlighting the advantages of transfer learning, where ResNet-18_IN_CFP* was ImageNet-trained ResNet-18 and fine-tuned with the CFP dataset. This pre-trained and fine-tuned scenario produced a model whose weights were initialized to understand the faces in IFPLD better.

The ResNet-18_IN_CFP* based Siamese network model with network embedding trained with Contrastive Loss or Triplet Loss exhibited better accuracy than the baseline Koch but less than ResNet-18_IN or ResNet-18_IN_CFP1 + SimCLR. ResNet-18_IN_CFP3 performed equally or slightly better than ResNet-18_IN_CFP1 + SimCLR did. The apparent limitation of IFPLD with a stark contrast of the yaw angle between the frontal and profile faces made it difficult for the network to learn using Contrastive Loss. This was because when the distance vectors of the images from the same profile were greater than the margin, the network did not learn. Regarding the Triplet loss-based network embedding, the simple method of generating a triplet dataset affects the generated network embedding. Moreover, ArcFace loss methods provided better separation for the inter-class embedding vectors and outperformed the triplet loss networks.

Among the models investigated for the Open Set Face Verification problem, which included four metrics and contrastive learning methods, the three ResNet-18_IN_CFP* + SimCLR models produced the best accuracy on average compared to the other training methods, which was 0.91. From the f1-scores in Table 3, ResNet-18_IN_CFP1 + SimCLR received the most top spots for each category, followed by ResNet-18_IN_CFP3 + Triplet Loss.

B. FACE IDENTIFICATION

As previously mentioned in Section III-D, the Siamese network was capable of performing the open set face verification task by outputting a probability value that could be used for face identification by matching a pair of frontal and profile face images using the N-way-1-shot learning principle, where N represents the number of individuals to be matched.

Table 5 shows the accuracy of various Siamese network models with the number of episode trials T as 400 and 1000, and the results indicated that the base models trained using ArcFace Loss or ResNet-18_IN_CFP3 in several training method categories, such as Contrastive Loss, Triple Loss, and SimCLR, provided the best performance. The highest accuracy was achieved by the ResNet-18_IN_CFP3 + Triplet Loss model, followed by the ResNet-18_IN_CFP1 + SimCLR and the ResNet-18_IN_CFP3 + SimCLR, which had a maximum difference of 10% from the top for different values of N. The network embedding generated by training the Siamese network using the ArcFace Loss Function on IFPLD provides a unique and distinctive embedding vector for face identification tasks in the N-way-1-shot scenario.

Several observations can be made from the Siamese network Face Verification and Face Identification results:

1. On average, ResNet-18_IN_CFP* + SimCLR demonstrated better accuracy than the other training methods for the same type of ResNet-18_IN_CFP*. SimCLR provided an advantage over other methods; for example, by comparing ResNet-18_IN_CFP1 + SimCLR with ResNet-18_IN_CFP1 + Triplet Network, there were 3.0% and 2.0% improvements in face verification and face identification accuracy, respectively. The only exception was ResNet-18_IN_CFP3 + Triplet Loss which performed better than the SimCLR version.

TABLE 5. Siamese network one shot evaluation.

Siamese Network based Model	T=400			T=1000		
	N			N		
	5	20	40	5	20	40
Koch	0.51	0.27	0.17	0.51	0.26	0.18
ResNet-18_IN	0.81	0.50	0.39	0.77	0.51	0.39
ResNet-18_IN_CFP1	0.71	0.45	0.33	0.68	0.47	0.29
ResNet-18_IN_CFP2	0.49	0.23	0.13	0.47	0.23	0.17
ResNet-18_IN_CFP3	0.63	0.43	0.3	0.67	0.42	0.31
ResNet-18_IN_CFP1 + Contrastive Loss	0.63	0.38	0.26	0.67	0.37	0.24
ResNet-18_IN_CFP2 + Contrastive Loss	0.68	0.4	0.23	0.73	0.39	0.31
ResNet-18_IN_CFP3 + Contrastive Loss	0.70	0.33	0.20	0.66	0.34	0.20
ResNet-18_IN_CFP1 + Triplet Loss	0.81	0.6	**0.54**	0.82	0.63	0.53
ResNet-18_IN_CFP2 + Triplet Loss	0.67	0.5	0.46	0.67	0.49	0.37
ResNet-18_IN_CFP3 + Triplet Loss	**0.89**	**0.72**	0.52	**0.89**	**0.69**	**0.58**
ResNet-18_IN_CFP1 + SimCLR	0.83	0.6	0.44	0.84	0.59	0.47
ResNet-18_IN_CFP2 + SimCLR	0.77	0.48	0.36	0.77	0.50	0.38
ResNet-18_IN_CFP3 + SimCLR	0.83	0.5	0.34	0.80	0.51	0.37

TABLE 6. Prototypical network performance on test class for T=400.

Training Scenario Model	N		
	5	20	40
Model 1 (N=5, K=6)	0.88	0.65	0.48
Model 2 (N=20, K=6)	0.94	0.77	0.65
Model 3 (N=40, K=6)	0.94	0.77	0.66
Model 4 (N=95, K=6)	0.94	0.81	0.71

2. Fine-tuning the ResNet-18_IN network with the ArcFace Loss Function on IFPLD generated a unique and distinguishable network embedding vector for N-way-1-shot face identification tasks. Using this fine-tuned base network in the Siamese network with relevant learning methods, such as SimCLR and Triplet Loss, led to an improved performance.

3. Face Identification is a multiclass classification problem. This makes the multiclass fine-tuning (CFP1) model more suitable for this task than the binary classification (CFP2) model. Consequently, the ResNet-18_IN_CFP1 models performed better than the ResNet-18_IN_CFP2 models in most cases.

The Prototypical network with a base CNN derived from network embedding within the Siamese network for face verification was trained in an N-way-6-shot learning scenario and used for face identification with 95 test classes that had not been seen before. As previously stated, the original method of the Prototypical network was modified to enable N-way-1-shot learning, where k > 1 and k-1 additional data were obtained through data augmentation. Table 6 shows the performance of the Prototypical networks using ResNet-18_IN_CFP1 + SimCLR base network, where the number of testing episodes T was set to 400, and the number of support images N_S and query image $N_Q = 6$. Comparing Tables 5 and 6, the Prototypical network was substantially more effective than the Siamese network for face identification. Training a Prototypical network in an episode containing $N_C = 95$ classes and $N_S = 6$ (Model 4) yielded the best results after evaluating testing data with varied N and K scenarios. The best-performing Siamese network model was improved

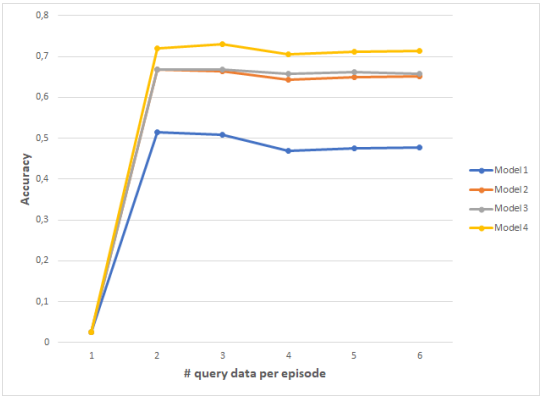


FIGURE 5. Effect of data augmentation in prototypical network.

by 6% for N=5, 9% for N=20, and 17% for N=40 by the Model 4 Prototypical network.

As shown in Figure 5, comparing the four models for the test data with N=40 demonstrated a significant impact of the data augmentation on the performance of the Prototypical network. When no data augmentation was applied or the original 40-way-1-shot learning procedure was implemented, the accuracy was less than 0.05. However, a significant improvement was observed with the introduction of data augmentation. The highest average performance was achieved when K=3 or two augmented data other than the original frontal/profile images were utilized. After this process, no further improvement in performance was observed. Therefore, it was concluded that the optimal number of augmented datasets is three.

V. CONCLUSION

This study investigated face recognition techniques relevant to IFPLD, a dataset comprising only one frontal and one profile face image per subject. The Siamese network was used for face verification, and it achieved an accuracy of 93% when the base network was fine-tuned using a face dataset similar to the target dataset. This approach enabled the acquisition

of the most suitable feature representation for the images in IFPLD. In some cases, fine-tuning the network using the ArcFace Loss function led to superior network embedding compared to the standard Cross entropy loss function. The accuracy and f1-score metrics of the Siamese network for Face Verification showed this. While the Siamese network can be used for face recognition in N-way-1-shot learning scenarios, the Prototypical network performance was significantly better with the same base network model. It achieved a maximum 17% accuracy improvement over Siamese network in the same testing scenario. The transformation of 1-shot-learning to k-shot-learning, where k-1 additional data were generated from data augmentation processes, was crucial for achieving a high-performance Prototypical network. It was observed that the Prototypical network achieved optimal performance at k=3 or with the addition of two data augmentations.

REFERENCES

[1] B. Gary Huang, M. Ramesh, T. Berg, and E. Learned-Miller, "Labeled faces in the wild: A database for studying face recognition in unconstrained environments," Univ. Massachusetts, Amherst, MA, USA, Tech. Rep. 07-49, Oct. 2007.

[2] Q. Cao, L. Shen, W. Xie, O. M. Parkhi, and A. Zisserman, "VGGFace2: A dataset for recognising faces across pose and age," in *Proc. 13th IEEE Int. Conf. Autom. Face Gesture Recognit. (FG)*, May 2018, pp. 67–74.

[3] 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. (CVPR)*, Jun. 2016, pp. 4873–4882.

[4] J. Deng, J. Guo, N. Xue, and S. Zafeiriou, "ArcFace: Additive angular margin loss for deep face recognition," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 4685–4694.

[5] T. Chen, S. Kornblith, M. Norouzi, and G. Hinton, "A simple framework for contrastive learning of visual representations," in *Proc. 37th Int. Conf. Mach. Learn. (ICML)*, 2020, pp. 1597–1607.

[6] J. Snell, K. Swersky, and R. Zemel, "Prototypical networks for few-shot learning," in *Advances in Neural Information Processing Systems*, vol. 30, I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, R. Garnett, Eds. Red Hook, NY, USA: Curran Associates, 2017.

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

[8] J. Deng, Y. Zhou, and S. Zafeiriou, "Marginal loss for deep face recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jul. 2017, pp. 2006–2014.

[9] W. Liu, Y. Wen, Z. Yu, M. Li, B. Raj, and L. Song, "SphereFace: Deep hypersphere embedding for face recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 6738–6746.

[10] H. Wang, Y. Wang, Z. Zhou, X. Ji, D. Gong, J. Zhou, Z. Li, and W. Liu, "CosFace: Large margin cosine loss for deep face recognition," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 5265–5274.

[11] K. Cao, Y. Rong, C. Li, X. Tang, and C. C. Loy, "Pose-robust face recognition via deep residual equivariant mapping," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 5187–5196.

[12] A. Nguyen, J. Clune, Y. Bengio, A. Dosovitskiy, and J. Yosinski, "Plug & play generative networks: Conditional iterative generation of images in latent space," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 3510–3520.

[13] F. Taherkhani, V. Talreja, J. Dawson, M. C. Valenti, and N. M. Nasrabadi, "Profile to frontal face recognition in the wild using coupled conditional GAN," 2021, *arXiv:2107.13742*.

[14] R. Hadsell, S. Chopra, and Y. LeCun, "Dimensionality reduction by learning an invariant mapping," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, vol. 2, Jun. 2006, pp. 1735–1742.

[15] H. Lin, H. Ma, W. Gong, and C. Wang, "Non-frontal face recognition method with a side-face-correction generative adversarial networks," in *Proc. 3rd Int. Conf. Comput. Vis., Image Deep Learn. Int. Conf. Comput. Eng. Appl. (CVIDL ICCEA)*, May 2022, pp. 563–567.

[16] B. Liu, W. Deng, Y. Zhong, M. Wang, J. Hu, X. Tao, and Y. Huang, "Fair loss: Margin-aware reinforcement learning for deep face recognition," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 10051–10060.

[17] J. Li, W. Jia, Y. Hu, S. Li, and X. Tu, "Learning to drop expensive layers for fast face recognition," *IEEE Access*, vol. 9, pp. 117880–117886, 2021.

[18] S. Sengupta, J. Chen, C. Castillo, V. M. Patel, R. Chellappa, and D. W. Jacobs, "Frontal to profile face verification in the wild," in *Proc. IEEE Winter Conf. Appl. Comput. Vis. (WACV)*, Mar. 2016, pp. 1–9.

[19] O. Vinyals, C. Blundell, T. Lillicrap, K. Kavukcuoglu, and D. Wierstra, "Matching networks for one shot learning," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 29, 2016, pp. 3637–3645.

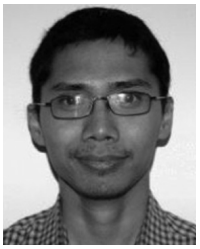
[20] G. Koch, R. Zemel, and R. Salakhutdinov, "Siamese neural networks for one-shot image recognition," in *Proc. ICML Deep Learn. Workshop*, vol. 2, Lille, France, 2015, pp. 1–30.

[21] X. Li, X. Yang, Z. Ma, and J.-H. Xue, "Deep metric learning for few-shot image classification: A review of recent developments," *Pattern Recognit.*, vol. 138, Jun. 2023, Art. no. 109381.

MUHAMMAD DJAMALUDDIN (Graduate Student Member, IEEE) is currently pursuing the Ph.D. degree with the School of Electrical Engineering and Informatics, Bandung Institute of Technology. His research interests include computer vision and machine learning.

RINALDI MUNIR received the bachelor's degree in informatics engineering and the M.Sc. degree in digital image compression from the Bandung Institute of Technology (ITB), Bandung, Indonesia, in 1992 and 1999, respectively, and the Ph.D. degree in image watermarking from the School of Electrical Engineering and Informatics, ITB, in 2010. In 1993, he started his academic career as a Lecturer with the Department of Informatics, ITB. He is currently an Associate Professor with the School of Electrical Engineering and Informatics, ITB, and the Informatics Research Group. His research interests include cryptography and steganography-related topics, digital image processing, fuzzy logic, and numerical computation.

NUGRAHA PRIYA UTAMA received the bachelor's degree in informatics from the Bandung Institute of Technology, Indonesia, in 2002, and the master's and Ph.D. degrees from the Tokyo Institute of Technology, in 2006 and 2009, respectively. His research interests include computer vision and neuroscience.



ACHMAD IMAM KISTIJANTORO (Member, IEEE) received the B.Eng. degree in informatics from the Institute of Technology Bandung (ITB), Bandung, Indonesia, the master's degree from the Delft University of Technology (TU Delft), Delft, The Netherlands, and the Ph.D. degree from the University of Newcastle upon Tyne, Newcastle upon Tyne, U.K. His research interests include distributed systems, parallel computation, and high performance computation.