Harnessing NNIA for Intelligent Triplet Generation in Fine-Tuned SENet-50 Based Face Recognition

S. Nebti *, K. Redjimi, M. Redjimi.****

** Department of Communication; Emir Abdelkader University; Algeria, Constantine, (nebtisalimaedu@gmail.com)*

***Department of computer Science; Université 20 Aoùt 1955; LICUS laboratory; Algeria, Skikda, (k.redjimi@univ-skikda.dz, m.redjimi@univ-skikda.dz)*

Abstract: FaceNet, founded on a triplet loss function, demonstrates strong face recognition performance. However, its effectiveness hinges significantly on the triplet selection strategy, which often fails to identify the most informative triplets, especially on largescale or complex datasets. To address this, we propose a training framework incorporating Multi-Objective Evolutionary Algorithms (MOEAs) into the triplet selection process. Our approach provides intelligent selection of triplets through a bi-objective optimization function that maximizes the anchor-negative distance (to enforce inter-class separability) while simultaneously optimizing the triplet-margin constraint (to facilitate active selection of hard and semi-hard negatives). In particular, we investigate the use of Fast Non-Dominated Sorting (FNDS), a Non-dominated Neighbor Immune Algorithm (NNIA), and a modified multi objective artificial bee colony algorithm (MOABC) for such selection. This offline, optimization-based approach incurs a large computational cost for triplet selection, but produces a filtered pool of high-quality triplets, enabling a more efficient learning of highly discriminative facial features. Extensive experiments on a range of benchmark datasets—including LFW, YTF, IJB-B, AgeDB, CFP-FP, CALFW, CPLFW, as well as masked and unmasked faces of the LFW dataset—demonstrate improvements over state-of-the-art methods in both recognition accuracy and training convergence.

Keywords: Deep Learning, Face Recognition, Triplet Loss, Multi-Objective Optimization, SENET-50, FNDS, NNIA, MOABC.

1. INTRODUCTION

Facial recognition is one of the most impactful applications of computer vision technology. This ubiquitous role is underscored by a broad set of applications, ranging from biometric authentication to surveillance applications, from digital identity in fintech, and e-government to automatic tagging and content management in multimedia, from face-based filters and friend-suggestion applications in social media to personalized recommendations.

The recent enhancements in the performance and popularity of face recognition systems have been due to the advances in deep learning. Specifically, convolutional neural network-based models that have drastically improved the way features were extracted and interpreted compared to conventional methods based on hand-designed features. These models are learnt at multiple levels in a hierarchical manner: edges and textures are firstly detected in a low level, and then, gradually, high-level facial traits are constructed through the successive convolutional layers. These hierarchal feature representations permit the learning models to achieve high accurate and reliable face verification and identification. With these innovations, deep learning is now the foundation of today's facial recognition systems, enabling them to meet the growing demands for scalability, ease, and security requirements across numerous sectors (Wang et al., 2018).

The newly risen paradigm of learning by transfer has also been substantial. It is as great for enhancing neural nets that learned on big data like ImageNet toward adjustment onto particular tasks by fine-tuning them. Hence, it goes a long way to-ward reducing the erstwhile demand for data labeled in abundance and computational power (Zhuang et al., 2019). The transfer learning approach has been particularly transformative in the design of facial recognition applications in which it effectively needs to cope with the dual challenge of limited availability of labeled data and strong generalization required for diverse facial features, expressions, and environmental conditions (Tan et al.,2018).

In spite of the efficacy of deep architectures and the transfer learning approach, the performance of metric learning based facial recognition architecture, namely, triplet-loss based networks like FaceNet, significantly hinges upon the choice of triplets (anchor, positive, negative), and the traditional triplet mining methods often suffer from random sampling or poor heuristics that can't guarantee the most informative triplets to be chosen. The situation gets even worse on face image data in the wild which is characterized by dynamics and variability.

In this area, multi-objective optimization (MOO) seems to be particularly promising framework to improve triplet selection. Instead of single criterion optimization or stochastic mini-batches, multi-objective evolutionary algorithms (MOEAs) can search the global triplet space by taking into account multiple, often conflicting criteria (e.g. maximizing inter-class separability and preservation of triplet loss margin) leading to generation of harder triplets able to push the model towards learning more robust face embeddings.

In this paper, we present a novel triplet selection method which incorporates MOEAs into a deep face recognition pipeline using transfer learning. In particular, our approach employs

state-of-the-art MOEAs (namely FNDS, NNIA, and MOABC, respectively) to find the hard or semi hard negative samples from the entire dataset in an offline pre-processing manner. Such negative samples not only lead to the formation of discriminating triplets, but also yield to progressive learning dynamics. The proposed framework is evaluated on standard domain datasets, where it shows significant improvements in recognition accuracy and neural model robustness over the traditional semi-hard mining approaches.

2. RELATED WORKS

Several recent advances in face analysis have used deep learning methods to automatically detect fine facial aspects, leading to very precise identification and verification. This part looks over some of the most impactful models and loss functions, each helping shape the progress of today's face recognition setups.

Early deep learning-based models had substantially improved the performance of face recognition system by learning Convolutional Neural Networks (ConvNets). DeepID (and DeepID2 as well as DeepID2+) introduced the use of dual supervision, learning face identification and face verification simultaneously to enhance the learned features' inter-class and intra-class feature dispersions. The authors also leveraged the desirable Convolutional Neural Activation's properties: sparsity, selectiveness and robustness. DeepID2+ model achieved state-of-the-art performance on multiple benchmark dataset: 99.47% on LFW and 93.2% on YTF (Wang et al., 2018).

A major milestone was finally reached with FaceNet in 2015 (Schroff et al., 2015). Instead of learning a classification layer, FaceNet directly learned to embed face images into a Euclidean space such that the distance in the space corresponded with face similarity. To learn such embeddings for verification and clustering tasks, FaceNet utilized a triplet loss function such that the distance between an anchor and a positive (same identity) was smaller than the distance between the anchor and a negative (different identity) by some margin. FaceNet achieved state-of-the-art with 99.63% on LFW and 95.1% on YTF.

Further improvements were made to loss functions and scaling up to larger datasets. ArcFace extended the triplet loss framework by inserting an angular margin into the softmax loss (Deng et al., 2019). Its effect was equivalent to maximizing the angular distance between feature vectors and class centers of the same class on the hypersphere, hence enhancing class separability. ArcFace's new approach toward embedding learning achieved a similar effect of high-class separability to triplet loss sampling, without additional computation incurred by triplet sampling. ArcFace advanced benchmark score significantly (99.83% on LFW, 98.01% on YTF) and remains a strong baseline method.

In order to deal with problems caused by high-identity-dimension in large-scale datasets, GroupFace and BroadFace were proposed. GroupFace constructs latent groups based on facial features that are extracted by a Group Decision Network (GDN), where initial group probabilities are calculated and a uniform distribution of samples among latent groups is ensured (Kim et al., 2020a). Regular classification loss and self-grouping loss are combined in the final loss function. Both the identity and group of a face are leveraged. BroadFace replaces typical mini-batch training with maintaining a large queue of embedding vectors from previous iterations, hence enabling optimal decision boundaries to be learned by the model across a broader set of identities. Another method used by BroadFace is compensation, which helps adjust errors between current and previous embeddings, thereby enhancing the model's generalization and, thus, its accuracy. GroupFace and BroadFace achieved excellent results (GroupFace: 99.85% on LFW, 97.8% on YTF; BroadFace: 99.85% on LFW, 98.0% on YTF) (Kim et al., 2020b).

Improvements in loss functions have also played a major role in enhancing the accuracy and robustness of such face verification systems. Softmax loss is one of the oldest deep learning methods; it computes the probability distribution over classes and penalizes the misclassification. It achieved 97.88% on the LFW dataset and 90.1% on YTF. While that is good, it is crucial to have a loss function to learn very discriminative face embeddings that showcase intra-class compactness and inter-class separability (Taigman et al., 2014). To achieve this, Center Loss was introduced in 2016. This way optimizes intra-class compactness by minimizing the Euclidean distance between feature embeddings and their respective class centers. The introduction of Center Loss greatly improved the performance to be 99.28% on LFW and 94.9% on YTF (Wen et al., 2019).

In 2017, techniques began to be angle-based. Angular Softmax Loss (A-Softmax) added an angular margin to decision boundaries, which effectively enhances discriminative power for learned embeddings. A-Softmax achieved the ability of about 99.26% on LFW and 94.1% on YTF. SphereFace made a step further from A-Softmax by introducing a multiplicative angular margin that better supported the distance between different classes in the embedding space. With such enhancement, the model's accuracy rose to 99.42% on LFW and 95.0% on YTF (Liu et al., 2017).

After SphereFace was introduced, one year later, in 2018, CosFace appeared. Though it also applied angular margins, CosFace used additive margin instead of multiplicative. This helped raise the LFW performance to 99.73% and YTF to 97.6% (Wang et al., 2018).

Finally, in 2019, RegularFace introduced a distinct regularization term to improve feature discriminability by increase inter-class separability and intra-class compactness in the angular space. Instead of either only using traditional softmax or margin-based losses, RegularFace added a regularization term to adjust the similarity between samples according to the relation between their intra-class distances and inter-class distances. By combining the A-Softmax and RegularFace, the Regular-Face-A achieved 99.61% accuracy on the LFW dataset as well as 96.7% on the YTF (Zhao et al., 2019).

Multi-objective optimization (MOO) methods have also been effective in improving facial-recognition tasks: a Pareto-based FaceNet approach, with advanced preprocessing techniques, and tested on large-scale datasets (CASIA-WebFace and

VGG-Face) achieved 94% accuracy on masked faces (compared with 91% for the FaceNet) (Akingbesote et al., 2023). Speed-constrained multi-objective particle swarm optimization (SMPSO) also reduced the dimensionality of features and succeeded in obtaining high accuracy results, such as 90.30% that was attained on the YALE dataset with only 11 selected features, doing better than both classical Eigenface and Fisherface methods (Larabi-Marie-Sainte et al., 2020). These advances would concern the transformative role of loss functions and optimization techniques in developing a face recognition system in different scenarios.

3. SYSTEM ARCHITECTURE: INTEGRATING MTCNN, SENET-50, AND MOO

Our face recognition pipeline includes a sequence of steps requiring robustness in each step as there is a cascading effect and each step critically affects the overall system performance. In particular, for face detection, the pipeline uses the MTCNN which is a very robust face detector method. This stage is followed by a robust feature extractor which is a SENet-50 previously trained on the VGG-Face2 dataset. Once features are extracted, the system applies effective MOO techniques to select the most challenging triplets for training. Moreover, a hard-mining filter is adopted during the training phase; to focus the model on learning the most complex samples, and hence improve its generalization capacity. Furthermore, various backpropagation methods were evaluated for the network' weights adjustment. The SGD optimizer has been found the most effective for our verification system. Although simple, the SGD's inherent stochasticity and slow convergence permitted better generalization and stronger learning compared to some more recent adaptive optimizers (e.g., Adam, and AdaBelief).

3.1 MTCNN Model for Face Detection

MTCNN refines face detection with high accuracy in a coarse-to-fine manner. It can robustly identify faces of varying scale and pose as well as partially occluded faces. It also accurately identifies key facial landmarks, such as the corners of the eyes, nose and mouth, the key points needed for face alignment. The detection pipeline consists of three steps (Zhang et al., 2016):

- P-Net (Proposal Network) generates candidate bounding boxes.
- R-Net (Refinement Network) eliminates the false examples and better bounding box position refinement.
- O-Net (Output Network): It generates final bounding box regression and landmark location.

Such a multi-stage design ensures a more accurate prediction of face bounding boxes and landmarks, making MTCNN a strong candidate for alignment and a pre-processing stage of face recognition systems.

3.2 SENet-50 for Feature Extraction

The backbone of our framework is SENet-50, a variant of the ResNet-50 model augmented with Squeeze-and-Excitation (SE) blocks. SE blocks use channel-wise attention mechanisms that adaptively recalibrate the importance of feature maps, enabling the network to focus its attention on the most informative features, which are highly effective in enhancing face recognition performance (Hu et al., 2018).

The SENet-50 network is pre-trained on the VGGFace2 dataset, introduced by Cao et al. (Cao et al., 2018), which contains approximately 3.3M images of 9131 persons. VGG-Face2 dataset has large variations in pose, age, illumination and ethnicity and thus enables learning of robust and generalizable facial representations.

From a technical perspective, the system uses SENet-50 to extract the 2048-dimension feature vectors from input face images. These embeddings are robust for representing the person's identity and maintaining the relevant identity-specific details for the task of recognition. The system uses the Keras-Vggface library for both model loading and pre-processing, which includes very specialized normalization and resizing procedures. Channel-wise attention mechanisms, combined with such powerful pre-training from VGGFace2, allow the obtained system to handle diverse facial variations with strong consistency in real world face images.

3.3 CNN Architecture Components:

The described CNN architecture is the embedding head for a FaceNet model (implemented in PyTorch). The task of this CNN is to produce the fixed-size embeddings from the input feature vectors (usually such feature vectors are pre-extracted from the input face images using the backbone network SENet-50).

The main part of this embedding head includes the following operations in order:

1. Input Feature Vector (2048 dimensions): The pre-trained model used to extract features is a SENet-50.
2. Fully Connected Layer (512 dimensions): This fully dense layer reduces the dimensionality from 2048 to 512.
3. A ReLU activation is added to introduce non-linearity, and for regularization Dropout is added to prevent overfitting.
4. A fully connected layer (128 dimensions) relating to the last embedding decreases dimensionality from 512 to 128.
5. L2 Normalization: the 128-dimensional embeddings are normalized using L2 normalization to put them on a unit hypersphere. This does an easy comparison between embeddings using distances like cosine similarity or Euclidean distance.

The result output is a 128-dimensional embedding vector, usable for tasks like face recognition or face verification.

3.4 Challenges in Triplet Loss Selection

First introduced in the FaceNet architecture, the Triplet Loss function is a metric learning objective designed to operate on image triplets—comprising an anchor image, a positive image from the same identity, and a negative image from a different identity. Its primary goal is to train the neural network to produce an embedding space in which the distance between the anchor and the positive is smaller than the distance between the anchor and the negative by some margin (Schroff et al., 2015).

The Triplet Loss function is defined as:

loss = max (||anchor - positive||² - ||anchor - negative||² + α, 0)

$$L_{triplet} = \max\left(0, d\left(f(x_a), f(x_p)\right) - d\left(f(x_a), f(x_n)\right) + \alpha\right) \quad (1)$$

Where:

$f(x)$: is the embedding function (CNN) that maps face images to the embedding space.

$(x_a), (x_p), and\ (x_n)$ are the anchor, positive, and negative images, respectively.

$d(.,.)$ is the distance metric (e.g., Euclidean distance)

α: acts as a margin that would enforce a minimum separation between the anchor-positive and anchor-negative pairs. α would ensure that the negative is at least α further from the anchor than the positive.

An important feature that Triplet loss focuses on is local comparisons in a mini-batch. This can be difficult to achieve informative triplets, especially for large datasets, wherein the number of potential triplets grows combinatorically and, thus, the computational cost is quite high. Randomly chosen triplets may lead to easy cases where there is little change in the gradient, and the training convergence will be slow. In order to address the issue of effective triplet generation, the CNNs are optimized for discriminative power using a multi-objective optimization approach.

4. MOO FOR ROBUST TRIPLET SELECTION

The core contribution of our work is a two-stage triplet selection strategy which provides the network with very informative training samples.

In the first stage, an offline multi-objective optimization of the negative candidates is conducted using a MOEA. For each given anchor-positive pair, the MOEA explores the entire set of all possible negative samples (from all other classes, based on the initial SENet-50 features) to find negatives which simultaneously maximize the anchor-negative distance, while bringing the triplet loss term d(a,p)-d(a,n)+α close to 0. This preselection step utilizes the global features characteristics to obtain a subset of negatives forming "critically positioned" and challenging triplets which are, therefore, potentially more valuable than the ones that could be obtained using only a random batch sampling strategy.

In the second stage, over the main training epochs, these MOEA-preselected triplets undergo an online hard triplet mining filter. As the FaceNet model head learns and its embedding space changes, this online filter re-evaluates the "hardness" of the pre-selected triplets based on the current model embeddings, discarding those triplets that no longer actively violate the triplet margin based on the current model embeddings (i.e., d(a,p)-d(a,n)+α>0). This way, only the MOEA-preselected triplets that continue to be "hard" relative to the current embeddings continue to contribute to the triplet loss and the back-propagation of the FaceNet model weights. Consequently, our face representation model is constantly striving to resolve the most challenging and hardness-guided samples by dynamically filtering the "hard" triplets; avoiding degrading triplets that were "hard" at some early point in the training phase.

The optimization is done on two contradictory objectives defined explicitly from the triplet loss constraint:

Maximize Anchor-Negative Distance

$$Obj\ 1\text{: } f_1(n_i) = d(a, n_i) \tag{2}$$

Where $d(a, n_i)$ is the Euclidean distance between the anchor "a" and the negative candidate "n_i".

This objective promotes the selection of negative samples that are dissimilar to the anchor in the feature space, thus maximizing the discriminative power learnt by the network.

Minimize Margin Proximity

$$Obj\ 2\text{: } f_2(n_i) = \left|\left(d(a,p) - d(a,n_i)\right) - \alpha\right| \tag{3}$$

Where: $d(a,p)$ is the Euclidean distance between the anchor"a" and the positive sample "p". α a margin =0.2

This objective focuses on "hard" or "semi-hard" negatives, i.e. whose distance to the anchor is close to the distance between the anchor and positive adding the margin. Minimizing this difference implies selecting negatives lying inside the decision boundary created by the margin. Therefore, hard or semi-hard negatives are difficult examples that are important for learning the metric efficiently.

The overall multi-objective problem can be stated as follows:

$$f(n_i) = \begin{cases} \text{Maximize} & f_1(n_i) = d(a, n_i) \\ \text{Minimize} & f_2(n_i) = \left|\left(d(a,p) - d(a,n_i)\right) - \alpha\right| \end{cases} \tag{4}$$

The Dominance Concept enables us to find solutions which are "Pareto-optimal", i.e., no other solution is strictly better with respect to all objectives.

Given two solutions Sol_1 and Sol_2, Sol_1 dominates Sol_2 (denoted $Sol_1 \succ Sol_2$) if and only if: Sol_1 is at least as good as Sol_2 in all objectives, that is:

$f_1(Sol_1) \geq f_1(Sol_2)$
$f_2(Sol_1) \leq f_2(Sol_2)$, and
Sol_1 is strictly better than Sol_2 in at least one objective, that is:
$f_1(Sol_1) > f_1(Sol_2)$ or $f_2(Sol_1) < f_2(Sol_2)$

In practice, the Pareto set optimization algorithm achieves a trade-off between the two conflicting objectives and finds solutions such that no other solution could improve any one of the objectives without causing a worsening of another (Ruppert et al., 2022). The MOEA algorithm iteratively evolves a population of negative candidates, using non-dominated sorting and evolutionary operators (crossover, mutation, and/or cloning...) to find a set of solutions representing the optimal trade-offs (Pareto front) between these two objectives. From this final optimized set, the "best" negative candidate (in this implementation, the top-ranked solution from the final Pareto front) is selected to form the triplet (a, p, n). This entire optimization process is repeated multiple times for each class to generate a comprehensive dataset of challenging triplets (Figure 1.).

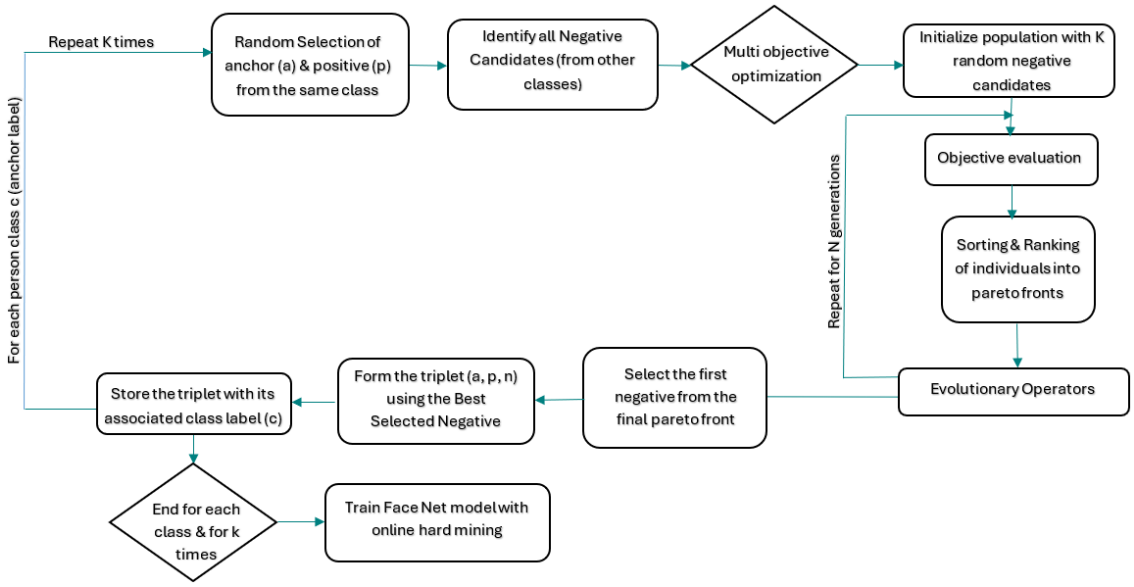

Fig. 1. Triplet Selection via MOEAs.

4.1 FNDS Based Model

A MOEA incorporating a selection scheme inspired by Fast Non-Dominated Sorting (FNDS) principles, which is the selection mechanism of NSGA-II (Zhang et al., 2020), is adopted to find negative samples n exhibiting optimal trade-offs between multiple competing objectives such that the selected triplets are discriminative enough for training the model.

For each class C in the training set, for an anchor a and positive p (randomly sampled from the same class C), the algorithm derives appropriate negative samples from within the negative candidates' pool (all samples not belonging to class C). The selection mechanism uses the 2 objectives as in (4). to compare negative candidates:

The algorithm steps:

Initialization:
- A population P is created by randomly sampling from the negative candidate's pool.

For a Fixed Number of Generations:
- **Evaluation:** Evaluate each individual in the population by computing its $f1$ and $f2$ scores.
- **Non-Dominated Sorting:**
  - Sort the evaluated population into different non domination levels (fronts): front [0], front [1], Members of the first front (front [0]) are individuals that are Pareto-optimal within the current population P.
- **Selection for Next Generation:**
  - A new population is formed by sequentially adding higher ranked individuals: (front[0], front[1], etc.)

Final Selection:
- After the final generation, the top K (e.g., 5) individuals from the final population are returned as the best negative candidates.

Repetition of the Process:
- This offline process is repeated several times (e.g., 5 times per class) for each given anchor-positive pair to produce a set of preselected triplets (a, p, n), with n being the top-ranked negative found by the FNDS.

4.2 NNIA based Model

NNIA algorithm provides major improvements over existing multi-objective evolutionary algorithms (MOEAs) like NSGA-II, SPEA2 and PESA-II by introducing a new mechanism of selection that targets the isolated non-Dominated individuals according to the values of their crowding distances (Li et al., 2021). It hybridizes immune-inspired principles—such as cloning and mutation—with multi-criteria optimization, ensuring a more effective search process. The main steps of the algorithm are outlined as follows:

Algorithm Steps:

This algorithm describes steps to select a negative sample n for given randomly sampled anchor and positive pair (a,p) belonging to the same class C.

Initialization step:
- An initial population P is randomly selected from the Negative candidates belonging to classes other than class C.

Evolutionary Multi-objective Optimization Loop:
- **For a number of generations do:**
  - **Objective Evaluation:**
    - Evaluate each individual in the population by computing its $f1$ and $f2$ scores.
  - **Non-Dominated Sorting:**
    - Rank individuals using non-dominated sorting to form the current Pareto fronts.
  - **Diversity preservation:**
    - Calculate crowding distances for individuals within the non-dominated fronts.
    - Select the Non-Dominated Set (ND) containing the non-dominated individuals

sorted by crowding distance in descending order (favoring less crowded solutions).

- **Selection:**
  - Select the active subset (with population_size /2) of the Non-Dominated Set ND.
- **Generation of New Offspring:**
  - Generate clone_size (e.g. 5) clones CT of the active population A. Individuals with higher crowding distance (or better rank) are proportionally cloned more often.
  - **Crossover**: through crossover, the negative indices of the cloned population CT and the active population A will be crossed to create Ct_indices, a temporary offspring set.
  - **Mutation**: the Ct_indices will be mutated and part replaced by another random index picked from the original negative_candidates pool.
- **Population Update for Next Generation:**
  - Construct new population P by combining mutated_indices (the off-spring) and the indices of the non-dominated set ND

Final Selection:

- **Final Evaluation**: Evaluate the objectives (f1, f2) for all individuals in the final population P.
- **Final Ranking**: Perform final non-dominated sorting on this final population to obtain the final Pareto front.
- **Select K Best Negatives**: Select the best K (e.g. 5) individuals from this final front based on rank and crowding distance.

Output :

- Return the list of best negative indices containing the indices of the K selected "best" negative samples for the initial anchor and positive pair.

4.3 MOABC based model

This algorithm introduces a new adaptation of the Multi-Objective Artificial Bee Colony algorithm (MOABC) with some key variations from the basic model. (Ye et al., 2023). Instead of using a grid-based archive structure and a single global leader selection, our approach implements a Fast Non-Dominated Sorting in conjunction with a global archive management system. Furthermore, our approach differs from the original MOABC in its final selection phase. Rather than promoting whole Pareto front diversity preservation, we make explicit selection of k top solutions from the Pareto-optimal set ranked according to their multi-objective scores.

Our modified MOABC also retains the three-phase optimization approach based on employed bees, onlooker bees, and scout bees to effectively balance exploration and exploitation within the solution space (Wang et al., 2020). This core mechanism is strengthened by a robust Pareto-optimal solution management system that retains all non-dominated solutions in a single external repository.

Algorithm Steps:

Initialization:

- A population of NP food sources is initialized with randomly selected indices from the set of negative candidates.
- Calculate the Multi-objective fitness of each initial food source:$(f_1 \& f_2)$
- Initialize the external archive (GlobalRep) to store the non-dominated solutions obtained so far.
- Initialize abandonment counter (C1) for each food source to zeroes.

For a number of iterations do:

- **Employed Bees Phase:**
  - **For each food source p do:**
    - Generate a new candidate negative in the neighborhood of population[p]: newbee.
    - Evaluate the multi-objective fitness of newbee using (let's call this newbeeCost).
    - Compare newbeeCost with the fitness of the current food source (Fitness[p]).
      - If newbeeCost dominates Finess[p] then replace the current food source with newbee and reset its counter (C1) to 0. Otherwise Increment its counter.
- **Update the Non-Dominated Archive (GlobalRep):**
  - Combine the current population ([population, Fitness]) with the solutions stored in the external archive GlobalRep.
  - Apply Fast Non-Dominated Sorting to the combined set.
  - Update GlobalRep to store only the individuals lying on the first (best) Pareto front of the combined set.
  - Update main population with the best NP individuals from the combined set based on its Pareto front ranking (also recompute Fitness for the updated population).
- **Onlooker Bee Phase:**
  - **For each Onlooker bee do:**
    - A food source I from the current population is selected using roulette wheel selection.
    - Pick a neighboring food source k of population[I], and produce a new candidate solution : newbee.
    - Calculate the fitness of the newbee (newbeeCost).
      - If newbeeCost dominates Finess[I] then replace the current food source I with newbee and reset its counter to 0. Otherwise Increment its counter.
- **Scout Bee Phase:**
  - **For each food source j do:**
    - If its counter C1[j] exceeds L limit then:
      - Replace the corresponding food source with a scout (i.e., with a random negative candidate index from the initial negative candidates' pool).
      - Update the position population[j], Fitness[j], and counter C1[j] for this scout bee accordingly.

Final Selection (After the final iteration):

- A final update of the GlobalRep and of the main population is performed by non-dominated sorting of their concatenated set.
- The algorithm returns the indices of the top K individuals from the final refined population, which is the best found

set of negative candidates found by the MOABC procedure according to the Pareto ranking.

This offline MOABC procedure is iteratively repeated for different anchor-positive pairs in order to obtain the complete set of pre-selected triplets (a, p, n) used to train the Face-Net Model.

6. EXPERIMENTAL DETAILS

Numerous datasets exist for face recognition research and evaluation. Most of them are available for download in the two following github repositories: which is associated with the work of Deng (Deng et al., 2021) and face.evoLVe by (Zhao, 2021). LFW, YTF, IJBB, AgeDB, CFP-FP, CALFW and CPLFW datasets are used. This suite has been selected to test performance over a large range of real-world challenge and to guarantee a direct comparison with state-of-the-art works.

Fig. 2. Examples from the used face recognition Benchmarks.

We used a variety of benchmark datasets to comprehensively test and validate the performance of our models on different real-world settings. We adopted the LFW (Labeled Faces in the Wild) dataset as a standard benchmark for unconstrained face verification and extended the evaluation further to the video-based recognition with the YTF (YouTube Faces) dataset. In order to test robustness in large-scale scenarios with increased variability, we used the IJB-B (IARPA Janus Benchmark B) dataset. The IJB-B dataset features a considerable number of variations in pose, occlusion, and illumination. For further robustness against specific covariates, we adopted several more challenging datasets, namely AgeDB and CALFW (Cross-Age LFW) for larger age gaps, and CFP-FP (Cross Pose Frontal-Profile) and CPLFW (Cross-Pose LFW) for extreme poses. Moreover, as a very relevant real-world challenge, we performed a dedicated test on the LFW dataset to evaluate performance on masked and subsequently unmasked faces (see Appendix A for details). Overall, these widely-used, difficult and complementary benchmarks supply a rigorous and multi-faceted validation for our approach, demonstrating our model's effectiveness and generalizability against the main challenges in face recognition.

We also improved the process of learning by adopting hard mining techniques, where we pick out difficult examples during the training session. Such an approach majorly improves the general robustness of the model as it emphasizes some of the more challenging scenarios, such as images with very similar appearances or parts of the face that are covered (Xu et al., 2020). Hard mining in face recognition with triplet loss means that hard mining selects triplets where the distance between the anchor and the negative example is close or the distance between the anchor and the positive example is large in embedding space (Schroff et al., 2015). Hard mining has an initial convergence slower than standard training but eventually has better performance on harder, realistic tasks because it helps reduce the dominance of easy examples, and the model has the opportunity to learn from its mistakes.

Moreover, the example balance across classes impacts model convergence and generalization ability very significantly in the task of face recognition. If there is an imbalance in class distributions—that is, some classes have significantly more samples than others—then this might result in a model that is biased and even excellent on the majority of classes, performing poorly on the minority ones. This is especially an important issue within face recognition systems (Leng et al., 2017). In this work, we choose to do data augmentation upfront, at the time of training, rather than online augmentation.

Furthermore, we performed two types of evaluation on our models. The first involved processing triplets individually, where each triplet was pre-selected using a hard triplet mining strategy to serve as a filter during training. The second approach was based on batch training, where the model was trained using randomly sampled triplets and optimized by computing the triplet loss over each batch. This latter method does not explicitly rely on hard mining strategies, but rather benefits from the diversity within the mini-batches. It is also worth noting that we did not include evaluations of the MOABC-based model in this phase, as its training process is significantly longer and its results are comparable to those of the FNDS-based model.

To ensure a consistent evaluation framework across both strategies, we divided the data into three parts in the ratio of 70%, 15%, and 15% for training, validation, and testing, respectively. At the end of the training process, the weight of the model that gives the highest AUC on the validation set would be saved since AUC is the best criterion to test how effective the model is at generalizing. With this, we can make sure that, out of all feasible choices of models, it will be the one better suited for making such distinctions.

5.1 Parameter settings:

For the FaceNet model training we used an SGD optimiser with a 0.001 learning rate, 0.9 momentum and 1e-4 weight decay for 30 epochs, and the triplet loss margin $\alpha = 0.2$

For the offline (MOEA) stage (our FNDS-based, NNIA and modified MOABC for negative triplet selection) the population size of 10 individuals and 5 evolving generations were typically used for each anchor-positive pair search. In particular, for our NNIA implementation, the clone size was set to 5, and the mutation rate was set to 0.1. The margin used in the MOEA's bi-objective evaluation was also 0.2. For the modified MOABC, the key parameters include a limit counter (C1_limit) of 5, and an acceleration coefficient (a) of 6. The search bounds for negative candidate indices were dynamically set from 0 to len(negative_candidates) - 1.

To comprehensively assess model performance, well-established metrics like AUC, EER, FAR, FRR, Test_Accuracy, FPR, and TPR, which characterize different aspects of the classification performance, are used. In essence, these metrics tell how well such a model can differentiate between positive and negative pairs. Joining these measures offers us a full look at how good the model works in various situations (De Mel et al., 2023).

Table 1. Test results on LFW.

Model	Training	Accuracy	AUC	EER	FAR	FRR
FNDS	triplet	99.76%	0.9961	0.0249	0.0185	0.0285
MOABC	triplet	99.84%	0.9971	0.0215	0.0143	0.0250
NNIA	triplet	99.90%	0.9974	0.0211	0.0135	0.0247
FNDS	batch	99.87%	0.9972	0.0203	0.0114	0.0241
NNIA	batch	99.91%	0.9973	0.0188	0.0111	0.0241

Table 1 shows the performance of FNDS, MOABC, and NNIA models under the two proposed training strategies: individual triplet selection and batch-based. Overall, across the triplet-based approach NNIA has the best performance across all metrics, achieving the best accuracy, 99.90%, the best AUC, 0.9974, the lowest EER, 0.0211, the lowest FAR, 0.0135, followed by MOABC and then FNDS. For the batch-based training strategy, NNIA attains the best results across all metrics. The batch-based NNIA achieves the best accuracy, 99.91%, the best AUC, 0.9973, and the lowest EER, 0.0188, and FAR, 0.0111. The batch-based training strategy appears to increase the FRR (False Rejection Rate) compared to the triplet-based strategy, but the batch-trained NNIA model achieves significantly lower FRR compared to the other models, demonstrating its efficacy and making it the best performing configuration across all models and strategies for face recognition. While MOABC achieves competitive results within the triplet-based training strategy, due to the expensive computational cost and the similar performance to FNDS within the batch-based training strategy, experimental validation was not continued for MOABC within batch-based.

Table 2. Test results on YTF dataset.

Model	Training	Accuracy	AUC	EER	FAR	FRR
FNDS	triplet	98.97%	0.9884	0.0515	0.0362	0.0641
MOABC	triplet	99.20%	0.9898	0.0489	0.0510	0.0459
NNIA	triplet	99.34%	0.9926	0.0411	0.0317	0.0483
FNDS	batch	99.43%	0.9922	0.0428	0.0361	0.0474
NNIA	batch	99.38%	0.9923	0.0419	0.0358	0.0465

From Table 2, we can conclude that NNIA is the most stable and overall best performing model in most criteria. Although the accuracy of the batch trained FNDS is marginally higher, accuracy alone is not a sufficient metric to assess the overall quality of the model, since it is computed at one decision threshold, and this threshold is usually chosen post hoc and does not generalize well. Threshold-independent metrics like AUC and EER are a more robust anchor point in model comparison, and while in batch training the FNDS model achieves a marginally higher accuracy, the NNIA model achieves a higher AUC (0.9923 vs 0.9922), and a lower, more balanced EER (0.0419 vs 0.0428, with the best/worst case at EER being furthest away from the ideal 0.04) and hence threshold and recognition-metric wide, NNIA is the most consistently good and well-balanced model.

To further evaluate the model's performance, we analyzed its separability by examining the distribution of distances between anchor-positive (dap in blue) and anchor-negative (dan in red) pairs. An ideal high-performing model should show very little or no intersection between these two distributions.

Table 3. Embedding Space Separability on YTF.

Model	d_ap Mean	d_ap Std	d_an Mean	d_an Std	d_an - d_ap
NNIA	0.4864	0.2223	1.3670	0.2143	0.8806
MOABC	0.4972	0.2356	1.3725	0.2180	0.8753
FNDS	0.4987	0.2365	1.3502	0.2196	0.8516

The analysis of the separability of the embedding space, presented in Table 3, shows that the most successful approach is the NNIA model. It has the largest separation gap (0.8806) owing to having the most compact intra-class clusters (mean d_ap = 0.4864). Furthermore, its improved robustness is evident in having the smallest standard deviations of both positive (0.2223) and negative (0.2143) distance distributions. Since the NNIA model produces the largest and the most stable margin between classes and its intra-class spread is the smallest among other methods, we can conclude that it outputs the most discriminative and structured embedding space.

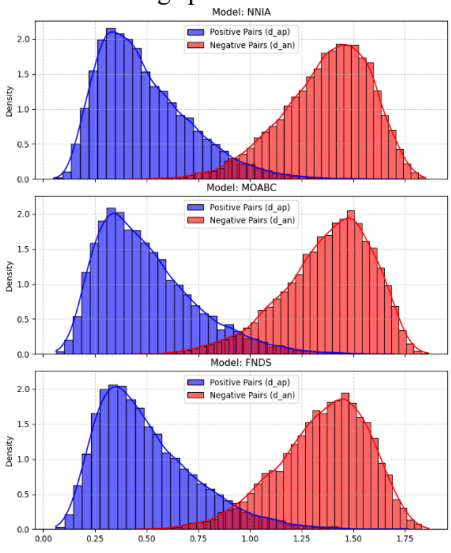

Fig. 3. Embedding Space Separability by Model on YTF.

Table 4. Results on AgeDB.

Model	Training	Accuracy	AUC	EER	FAR	FRR
FNDS	triplet	97.92%	0.9744	0.0864	0.0935	0.0776
MOABC	triplet	97.60%	0.9688	0.0935	0.0931	0.0935
NNIA	triplet	98.06%	0.9774	0.0778	0.0608	0.0922
FNDS	batch	98.59%	0.9796	0.0727	0.0741	0.0700
NNIA	batch	98.62%	0.9819	0.0716	0.0661	0.0746

Table 5. Results on CFP.

Model	Training	Accuracy	AUC	EER	FAR	FRR
FNDS	triplet	99.32%	0.9935	0.0381	0.0324	0.0392
MOABC	triplet	99.60%	0.9936	0.0369	0.0236	0.0420
NNIA	triplet	99.66%	0.9952	0.0314	0.0334	0.0278
FNDS	batch	99.78%	0.9978	0.0234	0.0236	0.0214
NNIA	batch	99.84%	0.9977	0.0214	0.0164	0.0232

Table 6. Results on CALFW.

Model	Training	Accuracy	AUC	EER	FAR	FRR
FNDS	triplet	99.21%	0.9905	0.0499	0.0460	0.0524
MOABC	triplet	99.41%	0.9911	0.0488	0.0423	0.0533
NNIA	triplet	99.65%	0.9943	0.0369	0.0393	0.0324
FNDS	batch	99.67%	0.9949	0.0355	0.0376	0.0325
NNIA	batch	99.71%	0.9952	0.0351	0.0421	0.0261

Table 7. Results on CPLFW.

Model	Training	Accuracy	AUC	EER	FAR	FRR
FNDS	triplet	97.77%	0.9741	0.0833	0.0854	0.0802
MOABC	triplet	98.01%	0.9749	0.0822	0.0693	0.0932
NNIA	triplet	98.35%	0.9786	0.0744	0.0820	0.0651
FNDS	batch	98.01%	0.9744	0.0849	0.0729	0.0955
NNIA	batch	98.23%	0.9774	0.0796	0.0813	0.0771

Table 8. Results on IJBB.

Model	Training	Accuracy	AUC	EER	FAR	FRR
FNDS	batch	97.45%	0.9774	0.0759	0.0457	0.0977
NNIA	batch	97.57%	0.9782	0.0730	0.0525	0.0890

The exhaustive evaluation on five challenging benchmarks (AgeDB, CFP-FP, CALFW, CPLFW and the largescale IJB-B) shows a very strong ranking consistency (Table 4 to Table 8), and shows the robustness of the proposed models. Across the datasets, which test for invariance to age, pose, and general unconstrained conditions, it is clear that the NNIA model shows the most stable performance. It almost always reaches the highest accuracy and AUC, and the lowest EER. On the cross-pose dataset CFP-FP for example, the batch-trained NNIA model reaches a near perfect accuracy of 99.84% which decisively outperforms all of the other variants.

Figure 4 depicted the ROC curves of the three models. It can be seen that NNIA is the best among the three models; the verification model of the NNIA is the best with AUC = 0.9943, which indicates that this model always obtains the best within-security accuracies as it can maintain the best TPRs at all FPRs, which means that it has the best trade-off between the security and recognition accuracy among the three models. On the other hand, although the other two models MOABC (AUC = 0.9911) and FNDS (AUC = 0.9905) perform well in the verification performance (ROCs), they seem to be slightly inferior to the NNIA model in terms of reliability and robustness.

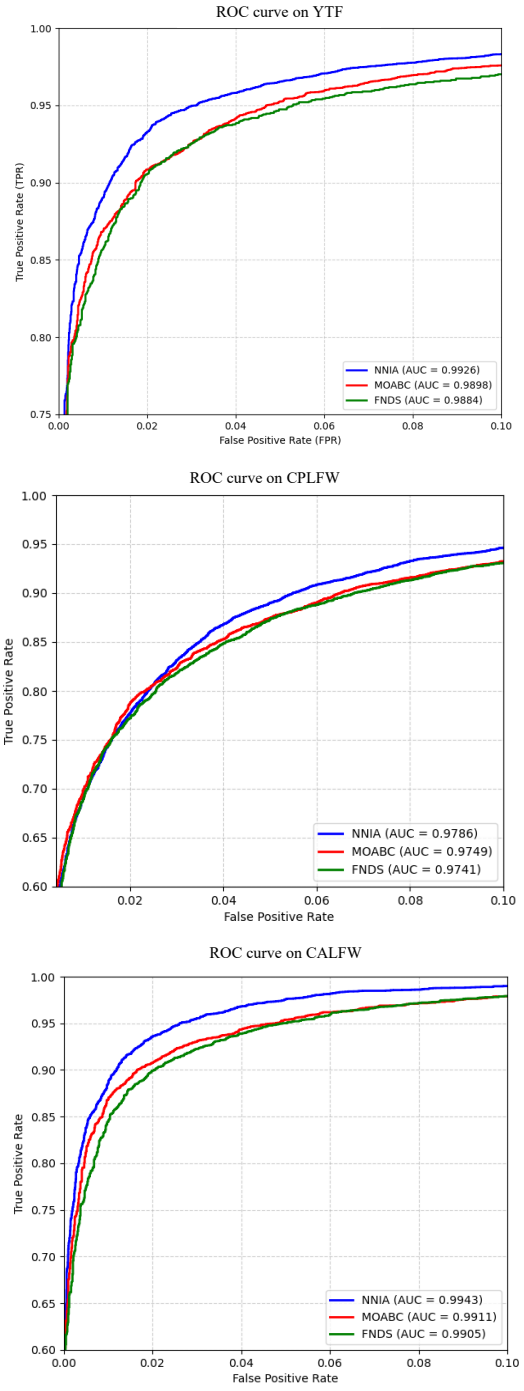

Fig. 4. ROC Curve Analysis.

All considered MOO-based triplet selection strategies (FNDS, NNIA, and the modified MOABC) were implemented with a small population size of (10) individuals and for a few number of generations (5) per search instance. Keeping a smaller number of population size and generations not only yield a similar computational cost to conventional random sampling techniques as that in FaceNet for both proposed strategies FNDS and NNIA, but also lead to a more optimized and efficient computational load, as both these strategies are based on a few numbers of Euclidean distance evaluations, and such

calculations are applied to the efficiently pre-extracted SENet-50 feature space. In essence, the FNDS and the NNIA-based triplet sampling provide a more refined and optimized triplet selection without any additional cost. On the contrary, the general procedures of the modified MOABC strategy comes with an additional cost compared to conventional sampling strategies, as the procedure involves certain number of iterative neighborhood search and frequent update of a higher number of archive contents for both sorting and ranking. In summary, the FNDS and the NNIA-based triplet sampling can provide better triplet quality with no computational overhead.

Table 9. Summary of the results.

Dataset	NNIA-based	Arcface (Deng et al., 2019)
LFW	99.91%	99.83 %
YTF	99.38%	98.01%
IJB-B	97.57%	96.02%
AgeDB	98.62%	98.38%
CFP-FP	99.84%	98.87%
CALFW	99.71%	96.10%
CPLFW	98.23%	93.43%

The results in table 9 show that the proposed NNIA-based method performs competitively against the state-of-the-art ArcFace method over the seven benchmark evaluations. Particularly, on the LFW dataset, our NNIA-based method reaches a nearly perfect accuracy of 99.91%, slightly outperforming ArcFace. The improvement margin quickly increases on more challenging benchmarks that aim to assess the robustness of the compared models to real-world variations. On the cross-age CALFW dataset for instance, our method achieves an excellent accuracy of 99.71%, largely surpassing the 96.10% accuracy obtained by ArcFace. Likewise, on the cross-pose CPLFW benchmark, our method reaches an accuracy of 98.23% which is almost five points higher than that of ArcFace (93.43%). The consistent and in many cases large margin improvements, notably on challenging datasets including YTF, IJB-B, CALFW and CPLFW, demonstrate the improved generalization power and the higher robustness of the proposed NNIA-based method.

6. CONCLUSION

This paper introduced and validated a novel framework incorporating MOO for boosting the deep face recognition, with a focus on the mining of informative triplet. Particularly, we have shown that this hard negative mining task can be more efficiently addressed by MOO algorithms—specifically an FNDS-based selection, a Non-dominated Neighbor Immune Algorithm (NNIA), and a modified Multi-Objective Artificial Bee Colony (MOABC)—which systematically guide the algorithm through the entire dataset and provide the model with hard and semi-hard negative samples that most optimally trade-off inter-class separability (maximizing anchor-negative distance) and triplet margin constraint. This principled and global selection of triplets overcomes the inability of conventional random sampling or batch-based semi-hard negative mining to learn decision boundaries since the majority of samples are non-informative easy triplets, whereas few or none are hard.

Our comprehensive empirical studies on popular LFW and YTF benchmarks show that all the three MOO-enhanced strategies outperformed the baseline FaceNet model. Notably, the NNIA-guided triplet selection yielded the most superior results, achieving 99.90% of accuracy on LFW and 99.34% on YTF as well as strong AUC and EER scores. Such state-of-the-art results underscores the potential of NNIA's immune-inspired principles in finding the best trade-offs when searching for the challenging triplets. Although the modified MOABC exhibited even both effective learning and separability and the FNDS-based approach provided a computationally efficient alternative, NNIA has been proven the most successful at delivering the most discriminative facial embeddings.

In summary, our research provides affirmative evidence for the positive effects of exploiting MOO for intelligent triplet mining in face recognition. Compared to the conventional heuristic selection strategies, our MOO-enabled framework promotes a stronger learning and more discriminative feature representations and therefore improved the verification accuracy. Particularly, the efficacy demonstrated by the NNIA variant brings an attractive opportunity to build higher-accuracy and more reliable face recognition systems for real-word application context, where a careful need to balance security and user convenience.

For the future research direction, instead of hard mining over a static triplet pool (as in strategy 1) or combining the MOO-based pre-selection and the batch-based triplet loss in training (as in strategy 2), we propose a hybrid training strategy for= effective batch-hard triplet mining via triplet selection under the MOO scheme. In this framework, a MOO algorithm (i.e., NNIA) generates a diverse set of candidate triplets by simultaneously optimizing multiple objectives such as the distance margin, class balance, and difficulty level of the triplets. The triplets are then selected as input in batch-hard mining algorithm to select the hardest anchor-positive-negative triplets within a mini-batch during training. This hybrid approach of mining triplets would combine the global selectivity of MOO and the local discriminative capability of batch-hard mining, potentially improving training efficiency and the embedding space's overall quality.

REFERENCES

Akingbesote, D., Zhan, Y., Maskeliūnas, R., & Damaševičius, R. (2023). Improving Accuracy of Face Recognition in the Era of Mask-Wearing: An Evaluation of a Pareto-Optimized FaceNet Model with Data Preprocessing Techniques. *Algorithms*, 16(6), 292.

Cao, Q., Shen, L., Xie, W., Parkhi, O. M., & Zisserman, A. (2018). Vggface2: A dataset for recognizing faces across pose and age. In 2018 *13th IEEE international conference on automatic face & gesture recognition* (FG 2018) (pp. 67-74).

De Mel, V. L. B. (2023). Survey of Evaluation Metrics in Facial Recognition Systems. 10.13140/RG.2.2.10974.20805.

DeepInsight (2021), InsightFace: Recognition datasets, GitHub repository, [Online]. Available:

*https://github.com/deepinsight/insightface/tree/master/recognition/_datasets_*

Deng J., "WebFace260M: A Benchmark Unveiling the Power of Million-Scale Deep Face Recognition," (2021), *CVPR*, pp. 10492-10502.

Deng, J., Guo, J., Niannan, L., & Zafeiriou, S. (2019). ArcFace: Additive Angular Margin Loss for Deep Face Recognition. *IEEE/CVF Conference on Computer Vision and Pattern Recognition* (CVPR).

Din, N. U., Javed, K., Bae, S., & Yi, J. (2020). A novel GAN-based network for unmasking of masked face. *IEEE Access,* 8, 44276-44287.Face-Masking-Unmasking [Code source]. GitHub. https://github.com/aniketshukla9/Face-Masking-Unmasking

Hu, J., Shen, L., & Sun, G. (2018). Squeeze-and-excitation networks. *In Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 7132-7141).

Huang G. B., Mattar M., Berg T., and Learned-Miller E., (2008). Labeled faces in the wild: A database for studying face recognition in unconstrained environments. In *Workshop on faces in 'Real-Life' Images: detection, alignment, and recognition*. inria-00321923f

Jin M., A Study of Face Alignment Methods in Unmasked and Masked Face Recognition. (*Master's thesis*, 2023). Uppsala University, Uppsala.

Kim, S., Kim, D., Cho, M., & Kwak, S. (2020). "BroadFace: Looking at Tens of Thousands of People at Once for Face Recognition." In *European Conference on Computer Vision* (ECCV).

Kim, Y., Park, W., Roh, M. C., & Shin, J. (2020). GroupFace: Learning Latent Groups and Constructing Group-based Representations for Face Recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (CVPR), pp. 5621-5630.

Larabi-Marie-Sainte, S., & Ghouzali, S. (2020). Multi-objective particle swarm optimization-based feature selection for face recognition. *Studies in Informatics and Control*, 29(1), 99-109.).

Leng, B., Yu, K., & Jingyan, Q. I. N. (2017). Data augmentation for unbalanced face recognition training sets. *Neurocomputing*, 235, 10-14.

Li, L., Lin, Q., Li, K., & Ming, Z. (2021). Vertical distance-based clonal selection mechanism for the multiobjective immune algorithm. *Swarm Evol. Comput*., 63, 100886. https://doi.org/10.1016/J.SWEVO.2021.100886.

Liu, W., Wen, Y., Yu, Z., Li, M., Raj, B., & Song, L. (2017). Sphereface: Deep hypersphere embedding for face recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 212-220).

Ruppert J., Aleksandrova M., and Engel T., k-Pareto Optimality-Based Sorting with Maximization of Choice and Its Application to Genetic Optimization. *Algorithms*, 15(11) (2022). pp. 420.

Schroff, F., Kalenichenko, D., & Philbin, J. (2015). FaceNet: A unified embedding for face recognition and clustering. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (pp. 815-823).

Taigman, Y., Yang, M., Ranzato, M. A., & Wolf, L. (2014). Deepface: Closing the gap to human-level performance in face verification. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 1701-1708).

Tan, C., Sun, F., Kong, T., Zhang, W., Yang, C., & Liu, C. (2018). A survey on deep transfer learning. In *International Conference on Artificial Neural Networks* (pp. 270-279). Springer, Cham.

Wang, H., Wang, W., Xiao, S., Cui, Z., Xu, M., & Zhou, X. (2020). Improving artificial Bee colony algorithm using a new neighborhood selection mechanism. Inf. Sci., 527, 227-240. *https://doi.org/10.1016/j.ins.2020.03.064*.

Wang, H., Wang, Y., Zhou, Z., Ji, X., Gong, D., Zhou, J., ... & Liu, W. (2018). Cosface: Large margin cosine loss for deep face recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 5265-5274).

Wang, M., & Deng, W. (2018). Deep Face Recognition: A Survey. ArXiv, abs/1804.06655. *https://doi.org/10.1016/j.neucom.2020.10.081.*

Wen, Y., Zhang, K., Li, Z., & Qiao, Y. (2019). A Comprehensive Study on Center Loss for Deep Face Recognition. *International Journal of Computer Vision*, 127, 668 - 683. https://doi.org/10.1007/s11263-018-01142-4.

Xu, C., Sun, X., Chen, Z., & Tan, S. (2020). Exhaustive hard triplet mining loss for Person Re-Identification. *Turkish Journal of Electrical Engineering and Computer Sciences*, 28(5), 3052-3067.

Ye, T., Wang, H., Zeng, T., Omran, M., Wang, F., Cui, Z., & Zhao, J. (2023). An improved two-archive artificial bee colony algorithm for many-objective optimization. *Expert Syst. Appl*., 236, 121281. https://doi.org/10.1016/j.eswa.2023.121281.

Zhang, K., Zhang, Z., Li, Z., & Qiao, Y. (2016). Joint face detection and alignment using multitask cascaded convolutional networks. *IEEE signal processing letters*, 23(10), 1499-1503.

Zhang, M., Wang, L., Cui, Z., Liu, J., Du, D., & Guo, W. (2020). Fast Nondominated Sorting Genetic Algorithm II with Lévy Distribution for Network Topology Optimization. *Mathematical Problems in Engineering*, 2020, 1-12.

Zhao, J. (2021). face.evoLVe: High performance face recognition library based on PaddlePaddle & PyTorch. GitHub. *https://github.com/ZhaoJ9014/face.evoLVe*

Zhao, K., Xu, J., & Cheng, M. M. (2019). Regularface: Deep face recognition via exclusive regularization. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (pp. 1136-1144).

Zhuang, F., Qi, Z., Duan, K., Xi, D., Zhu, Y., Zhu, H., Xiong, H., & He, Q. (2019). A Comprehensive Survey on Transfer Learning. *Proceedings of the IEEE*, 109, 43-76. https://doi.org/10.1109/JPROC.2020.3004555.

Appendix A. EVALUATION ON MASKED FACES VIA GAN-BASED RECONSTRUCTION

To further investigate the robustness of our pre-trained face recognition models in practical application settings, we conducted an evaluation on masked and unmasked faces. More specifically, we used the MaskTheFace application (Din et al., 2020) to systematically produce masked variants of the LFW dataset and for Mask Removal and Face Reconstruction, we used a U-Net based Generative Adversarial Network (GAN) with a dual-module architecture This GAN uses a mapping module to locate masks and generate a binary segmentation mask, followed by an editing module that removes the mask and reasonably fills the occluded part of the face.

Fig. 5. Illustrative Examples of Challenges in Reconstructing Unmasked Faces.

The proposed face recognition models (i.e. NNIA, FNDS, and MOABC) are then evaluated under two scenarios:

1. Training and testing are both performed on GAN-reconstructed (unmasked) images, in order to simulate a fully masked-aware pipeline.
2. Training on original unmasked LFW, and testing on GAN-reconstructed masked images to simulate real-world application cases.

Table A.1. Training and Testing on GAN-Reconstructed LFW.

Model	Accuracy	AUC	EER	FAR	FRR
FNDS	99.79%	0.9964	0.0275	0.0238	0.0307
MOABC	99.73%	0.9961	0.0287	0.0205	0.0342
NNIA	99.82%	0.9974	0.0232	0.0229	0.0234

Table A.2. Training on Original LFW and Testing on GAN-Reconstructed LFW.

Method	Accuracy	AUC	EER
FNDS	98.99%	0.9863	0.0549
MOABC	98.69%	0.9830	0.0638
NNIA	98.90%	0.9858	0.0574

The results for the case when the training and the testing are both performed on GAN-reconstructed LFW images are presented in Table A.1. NNIA achieves the best performance with the overall high accuracy of all three models (Accuracy: 99.82%, AUC: 0.9974, EER: 0.0232). This indicates that despite the fact that the unpainted images are synthetic, they still contain enough identity-specific information to allow very accurate recognition.

When training is performed on original LFW faces & testing on GAN based unmasked & reconstructed faces (Table A.2), FNDS based model attained slightly better performance: acc. 98.99%, best AUC 0.9863, lowest EER 0.0549. It means that, some triplets chosen by FNDS have the model capture more distinguishing information from the mixture of both original unmasked region and GAN-unpainted part (face contours, synthesized mouth or nose, ...), while on the other hand, NNIA's optimization trajectory (inherently suiting unmasked faces) implicitly constructed an embedding space which is less amenable to classify this kind of GAN reconstructed facial samples (Figure 5).

When comparing our results with the most relevant work (Jin et al. 2023) investigating the masked and unmasked face recognition on LFW dataset, it can be seen that our findings outperformed them. The best AUC reached in that study has been succeeded by ArcFace (0.9831), then VGGFace (0.9712) and FaceNet (0.9690). Whereas both of our models (NNIA and FNDS) tested on the GAN reconstructed images have achieved higher AUC of 0.9858 and 0.9863, respectively. This suggests that our MOO-based triplet selection strategies and GAN-based facial reconstruction provide better robustness to occluded (masked) input. The results also highlight the difficulty of masked face recognition and the requirement for more adaptive strategies to address these specific challenges.