Research on the effectiveness of concatenated embeddings in facial verification

Denys Khanin[1,*,†], Viktor Otenko[1,†] and Volodymyr Khoma[1,†]

[1] Lviv Polytechnic National University, Information Security Department, 12 Stepana Bandery str., 79000 Lviv, Ukraine

Abstract

In the era of digital authentication, facial verification systems have become a cornerstone of security protocols across various applications. This study explores the performance synergy from concatenated embeddings in enhancing biometric authentication accuracy. By leveraging the Celebrities in Frontal-Profile dataset (CFP), we investigate whether the fusion of embeddings generated by models such as VGG-Face, Facenet, OpenFace, ArcFace, and SFace can result in a more robust authentication process. Our approach is rooted in the hypothesis that the diverse strengths of these models, when combined, can address the limitations inherent in single-model systems, thus providing a more comprehensive solution to facial verification. The approach involves computing the L2 distance between normalized concatenated embeddings of an input face image and an anchor, thereby determining the authenticity of the individual. Experiments are designed to compare the performance of singular model embeddings against concatenated embeddings, employing metrics such as accuracy, False Acceptance Rate (FAR), and False Rejection Rate (FRR). One of the critical aspects of our research is the implementation of Z-Score normalization and L2 normalization processes to standardize the embeddings from different models. These normalization techniques are vital in ensuring that the diverse outputs from various models are effectively combined, maintaining balance and consistency in the feature vectors. Additionally, our methodology includes a comprehensive evaluation framework that meticulously analyses the trade-offs between computational efficiency and performance gains achieved through model concatenation. The findings of this research could significantly contribute to the development of more secure and reliable facial verification systems by using multiple existing models without the need for new model research, designing, and training. This approach not only optimizes resource utilization but also provides a scalable solution that can be readily adapted to existing systems, enhancing their security measures without extensive overhauls. Furthermore, the study's insights into the integration of model outputs could pave the way for future innovations in biometric authentication, encouraging the development of hybrid systems that combine the best attributes of various neural network architectures. This research underscores the potential of concatenated embeddings in revolutionizing facial verification technology. By harnessing the power of multiple neural network models, we can create a system that delivers superior accuracy and robustness, addressing the pressing need for advanced security solutions. This study sets the stage for further exploration into multi-model integration, offering a promising direction for future advancements in biometric authentication.

Keywords

facial verification, biometric authentication, neural networks, concatenated embeddings, machine learning, deep learning, model fusion, facial recognition, verification accuracy, security systems

1. Introduction

In today's digital landscape, facial verification [1] systems have become pivotal in ensuring the security and authenticity of individual identities across various applications, from mobile device security to access controls in sensitive environments. The adoption of facial recognition technology is driven by its non-intrusive nature and the unique, hard-to-replicate characteristics of the human face, positioning it as a front-runner in biometric authentication methods. Furthermore, the integration of socio-cyber-physical systems security frameworks provides a comprehensive approach to enhancing cybersecurity measures, as highlighted by Yevseiev et al. in their detailed monograph on socio-cyber-physical systems security [2].

This research investigates the potential of enhancing facial verification accuracy through concatenated embeddings from multiple neural network [3] models. Utilizing the CFP dataset [4], we aim to determine whether the integration of various model embeddings can produce a more robust and secure biometric authentication system. By examining the performance synergy of these concatenated embeddings in comparison to singular model outputs, this study aims to contribute to the development of more advanced and reliable facial verification techniques with the existing set of models for facial verification.

CSDP-2024: Cyber Security and Data Protection, June 30, 2024, Lviv, Ukraine

*Corresponding author.
†These authors contributed equally.
denys.o.khanin@lpnu.ua (D. Khanin); viktor.i.otenko@lpnu.ua (V. Otenko); volodymyr.v.khoma@lpnu.ua (V. Khoma)

0009-0001-4009-0202 (D. Khanin); 0000-0003-4781-7766 (V. Otenko); 0000-0001-9391-6525 (V. Khoma)

1.1. Background

The evolution of biometric authentication technologies has been significantly influenced by advancements in machine learning and deep learning [5], particularly in the domain of facial recognition. Neural network models, such as VGG-Face, Facenet, OpenFace, ArcFace, and SFace, represent the forefront of research and development in this field. These models are designed to extract and analyze facial features [6] from images, transforming them into numerical representations known as embeddings. These embeddings capture the unique aspects of an individual's facial structure, enabling systems to perform verification tasks with high degrees of accuracy. The success of these models is predicated on their ability to learn complex patterns and variations in facial features across diverse datasets, under various conditions of lighting, pose, and expression.

In facial verification technology, using just one neural network model comes with certain limitations. Different models excel in various aspects, such as accuracy, speed of processing, and their ability to handle changes in lighting or facial features [7]. The drive for better performance and reliability in these systems often requires large and varied datasets for training, which can be resource-intensive. Additionally, there is a constant need to develop and test new model architectures that can effectively transform facial images into useful numerical data, known as embeddings. This scenario suggests that combining several neural network models might offer a more efficient solution. By leveraging the unique strengths of multiple models, such an approach could potentially overcome the common challenges in facial verification. This sets the stage for investigating how the integration of outputs from different models could lead to improvements in system performance.

1.2. Problem statement

The hypothesis driving this research emerges from a critical challenge within the realm of facial verification systems: the limitations of using single-model architectures in achieving consistently high accuracy across diverse conditions. This issue underscores the necessity of exploring alternative strategies that can leverage the strengths of existing technologies without the need for constant model retraining, dataset updates, or the development of new architectures [8]. Moreover, an exploratory survey by Hlushchenko and Dudykevych on access control paradigms highlights the evolving landscape of policy management and its implications for biometric security systems [9].

Current facial verification systems often rely on a singular neural network model, which may excel under specific conditions but fall short in others. This reliance poses a significant problem as it demands continuous updates to the model and its underlying dataset to address emerging challenges and maintain system performance. Such an iterative cycle of development is resource-intensive, requiring substantial investments in data collection, processing, and computational power. Additionally, the creation of new model architectures to improve feature extraction and classification accuracy further complicates the process, making it unsustainable in the long run. The hypothesis presented in this study arises from these challenges, proposing the use of concatenated embeddings [10] from multiple models as a means to bypass the constraints of singular model dependency. This approach aims to explore whether integrating the diverse capabilities of established models can offer a more robust and accurate solution for facial verification, thus addressing the core issues associated with the current methodologies.

1.3. Objectives of the research

This research focuses on key goals designed to explore improvements in facial verification systems:

- To create a system to test both individual and combined embeddings from models such as VGG-Face, Facenet, OpenFace, ArcFace, and SFace, leveraging the CFP dataset for comprehensive analysis.
- To measure the effectiveness of each model and their combinations using accuracy, FAR, and FRR [11].
- To analyze system performance across single and combined model embeddings to identify the most effective strategies for facial verification.
- To extract insights for potential system enhancements and recognize any associated challenges with multi-model embeddings.

2. Related works

The problem of enhancing the accuracy and robustness of facial verification systems has been a focal point in numerous studies due to ongoing challenges such as spoofing, adversarial attacks, and varying conditions of image capture. Ding and Tao (2018) addressed the limitations of traditional face recognition approaches by introducing Trunk-Branch Ensemble Convolutional Neural Networks for video-based face recognition, which improved recognition accuracy but still faced challenges in handling dynamic and complex environments [12]. Nagrath et al. (2021) highlighted the need for lightweight and efficient neural networks like MobileNetV2 for real-time applications, but their study also pointed out the difficulties in maintaining high accuracy under real-time constraints [13].

Li et al. (2018) focused on enhancing deep learning features with facial texture features for improved recognition performance, but the integration of different feature extraction techniques remained complex and computationally intensive [14].

Moon et al. (2016) developed a face recognition system based on convolutional neural networks using multiple distance faces, which further emphasized the necessity of integrating various models to enhance system robustness, yet it also indicated the increased computational demands [15].

Yang et al. (2019) explored federated machine learning for face verification, addressing privacy and security concerns while maintaining high verification accuracy. Their research underscored the challenge of managing decentralized data and the need for efficient data integration techniques [16].

Bhuiyan et al. (2017) presented a noise-resistant network for face recognition under noisy conditions, which highlighted the ongoing challenge of achieving robust performance in diverse real-world scenarios [17].

Recent advancements have shown that despite significant improvements in facial verification technologies, several unresolved issues persist. Gao et al. (2018) discussed privacy-preserving techniques in face recognition, which remain a critical concern in the deployment of these systems [18]. Furthermore, the study by Hanmandlu et al. (2013) on Elastic Bunch Graph Matching for face recognition identified the need for better handling of pose and illumination variations [19].

The integration of multiple models to leverage their unique strengths and mitigate individual weaknesses is a promising approach, as highlighted by recent research on hybrid and ensemble methods. However, this integration introduces new challenges, such as increased computational complexity and the need for sophisticated normalization techniques to ensure consistent and reliable performance [20, 21]. Additionally, Brydinskyi et al. provide a comparative analysis of modern deep-learning models for speaker verification, demonstrating the critical role of model selection and combination in enhancing verification accuracy [22].

These studies collectively underscore the necessity of integrating multiple models to leverage their unique strengths and mitigate individual weaknesses, aligning with our research objective of using concatenated embeddings to enhance facial verification systems' accuracy and robustness. The proposed approach builds on the foundations laid by these works, aiming to address their limitations through the strategic combination of diverse neural network models.

3. Methodology

3.1. Dataset

The CFP dataset plays a pivotal role in our study, offering a nuanced exploration of facial verification across varying poses. Its construction and attributes are as follows:

- Size and Volume: The dataset consists of images of 500 individuals, with 10 frontal images per individual.
- Resolutions and Quality: Including a mix of resolutions and qualities, the dataset mirrors the variability encountered in real-world applications, ranging from high-definition to lower-quality images, challenging the adaptability of verification systems to varying image fidelity.
- Diversity of Conditions: It spans a broad spectrum of real-life conditions— different lighting scenarios from natural daylight to artificial and low light environments, varied backgrounds from simple to cluttered scenes, and a wide range of facial expressions and poses, especially focusing on extreme profile views that pose a significant challenge to current algorithms.
- Source: Images are sourced from the internet, capturing "in the wild" conditions that include a balanced representation of genders, ethnicities, and professions. This approach ensures the dataset reflects the complexity and diversity of facial appearances and expressions in everyday life.

CFP dataset examples are shown below in Fig. 1: 4 random face images for each of the 3 individuals from the dataset.

Figure 1: CFP dataset example images of individuals

3.2. Models

This study employs various neural network models, each with unique architectures and characteristics, to determine the effectiveness of concatenated systems in facial verification. The models utilized include VGG-Face, Facenet, Facenet512, OpenFace, ArcFace, and SFace, each designed to extract and analyze facial features from images, transforming them into numerical representations known as embeddings. A comparison of architecture, embedding dimensions, training focus, and key features of each model is described in Table 1.

3.3. Concatenation system

The concatenation system forms a pivotal component of our methodology, designed to harness the collective strengths of multiple facial recognition models. This approach seeks to enhance the robustness and accuracy of facial verification by leveraging the diverse feature representations extracted by different models. The process involves several key steps, each contributing to the formation of a comprehensive feature set that is used for facial verification:

1. Model Selection: The first step involves selecting a set of neural network models, such as VGG-Face, Facenet, OpenFace, ArcFace, and SFace, each known for its unique approach to capturing facial features. This diversity is crucial for assembling a wide-ranging feature set.
2. Output Extraction: For each model, we extract the output embeddings that represent the facial features identified by that model. These embeddings are the high-dimensional vectors that encapsulate the model's interpretation of the facial features.
3. Z-Score Normalization [28]: To standardize the embeddings from different models, we apply Z-Score normalization to each embedding vector. This normalization process adjusts the embeddings so that they have a mean of 0 and a standard deviation of 1. This step is essential for mitigating the variance in scale and distribution of the embeddings across different models, ensuring that no single model's output disproportionately influences the concatenated feature vector.

Table 1
Neural network model characteristics

Model	Architecture	Embedding Dimension	Training Focus	Key Features
VGG-Face[23]	VGG-16	4096	Facial Recognition	Deep convolutional layers, trained on large facial image dataset, use small (3×3) convolution filters, capture fine facial details
Facenet[24]	Inception-ResNet v1	128	Triplet Loss Function	Compact embeddings optimize distance between similar/dissimilar faces, use triplet-based loss function to enhance verification accuracy
Facenet512	Inception-ResNet v1	512	Extended Triplet Loss Function	Higher-dimensional embeddings, capture more nuanced features, an extension of Facenet with increased embedding size for a richer representation
OpenFace[25]	nn4.small2	128	Real-time Recognition	Balances accuracy and computational efficiency, suitable for real-time applications, a lightweight model designed for practical use on modest hardware
ArcFace[26]	ResNet-100	512	Additive Angular Margin Loss	Enhances discriminative power, improves geometric accuracy of feature space, uses additive angular margin loss to manage class margins
SFace[27]	Xception-39	128	Scale Variations	Efficient handling of scale issues, rapid and accurate recognition, perform well on high-resolution images, notable efficiency and accuracy, especially on large datasets

4. Concatenation: Following normalization, the embeddings from all selected models are concatenated into a single, comprehensive feature vector. This concatenated vector represents a fusion of the diverse facial features recognized by the individual models, capturing a broader spectrum of facial characteristics than any single model could.
5. L2 Normalization [29]: The concatenated feature vector undergoes L2 normalization, which scales the vector to have a unit norm. This normalization step is critical for preparing the feature vector for similarity calculations, ensuring that the magnitude of the vector does not affect the distance measurements.
6. EER Determination: Upon calculating the L2 distances between facial image pairs, we identify the Equal Error Rate (EER), the point where the FAR and the FRR converge. Determining the EER is essential, as it represents an optimal balance point for the system's decision threshold, minimizing both false positives and false negatives. This optimal threshold is then used to distinguish between matches and non-matches across the entire dataset, allowing for the proper measurement of verification metrics such as accuracy, FAR, and FRR.

3.4. Evaluation metrics

To analyze the performance of our facial verification systems, including both single and combined models, we use three main metrics: accuracy, FAR, and FRR. These metrics help us understand the systems' performance in correctly identifying faces.

False Acceptance Rate: FAR measures the likelihood that the system incorrectly verifies an impostor as a genuine user. It is crucial to evaluate the security aspect of the facial verification system, with lower values indicating higher security. FAR is calculated as:

$$FAR = \frac{FP}{(FP + TN)}, \quad (1)$$

where FP is the number of false positives, and TN is the number of true negatives.

False Rejection Rate: FRR assesses the frequency at which the system wrongly rejects an authentic match. This metric is important for understanding the usability of the system, as a high FRR may lead to user frustration. Lower FRR values are desirable, indicating better performance. FRR is calculated as:

$$FRR = \frac{FN}{(TP + FN)}, \quad (2)$$

where FN represents false negatives, and TP denotes true positives.

Accuracy: This metric measures the overall effectiveness of the facial verification system. It is calculated as the ratio of correctly identified instances (both true positives and true negatives) to the total number of instances. High accuracy indicates that the system is effective in correctly verifying facial identities. The formula for accuracy is given by:

$$Accuracy = \frac{TP + TN}{(TP + TN + FP + FN)}, \quad (3)$$

where TP represents true positives, TN denotes true negatives, FP stands for false positives, and FN signifies false negatives.

Together, these metrics provide a comprehensive overview of the system's performance, offering insights into its accuracy, security, and usability. By evaluating these metrics, we can make informed decisions on optimizing model configurations and improving facial verification systems.

3.5. Technical setup

Experiments were conducted on a defined technical framework comprising specific hardware and software components.

Hardware Configuration: MacBook Pro 16 with an M1 Pro processor and 16GB RAM, offering enough

computational power for handling neural network operations.

Software Configuration:

- Python 3.11: Selected for its widespread support for data analysis and machine learning tasks.
- Tensorflow-metal 1.1.0: Optimized for the M1 Pro, enhancing machine learning computation speeds.
- OpenCV-python 4.9.0: Utilized for image processing tasks such as loading, resizing, and cropping.
- Deepface 0.0.83: A library providing access to several facial recognition model weights (VGG-Face, Facenet, OpenFace, ArcFace, SFace) and their functionalities, streamlining the embedding extraction.

4. Experiments and results

4.1. Data preprocessing

Data preprocessing is a crucial initial phase in our experiment, ensuring facial images are properly conditioned for analysis by various neural network models. Here's an outline of the preprocessing steps undertaken:

Loading Images: Images are first loaded in RGB color space, retaining their essential color information which is crucial for accurate analysis of facial features.

Scaling Pixel Values: To standardize the images, pixel values for each color channel are scaled to a range from 0 to 255.

Model-Specific Normalization: Depending on each model's requirements, specific normalization techniques are applied to the image data to match the conditions under which the models were trained [30].

For Facenet model:

$$img = \frac{img - mean(img)}{std(img)}, \tag{4}$$

where mean and std are the mean and standard deviation of the image's pixel values, respectively.

For Facenet512 and ArcFace models:

$$img = \frac{img}{127} - 1, \tag{5}$$

For the VGGFace model:

$$img = img - \begin{bmatrix} 93.5940 \\ 104.7624 \\ 129.18633 \end{bmatrix}, \tag{6}$$

this formula represents the subtraction of mean values for each color channel (R, G, B) based on VGGFace1 training data.

For OpenFace and SFace models:

$$img = \frac{img}{255}. \tag{7}$$

4.2. Singular models evaluation

In the evaluation phase of our experiments, each neural network model was assessed individually to establish its performance on the CFP dataset. A crucial part of this assessment involved determining the EER for each model, which provides a threshold at which the rate of false acceptances is equal to the rate of false rejections.

The process began with the calculation of distances between facial embeddings for both genuine and impostor pairs. Following this, we computed the EER for each model, which then served as a basis for determining the corresponding accuracy at the EER point and the best overall accuracy achieved by the model. These metrics give us insight into the models' capabilities in facial verification tasks under the diverse conditions presented by the CFP dataset.

The results of the singular model evaluations are summarized in Table 2.

Table 2
Singular model metrics on the CFP dataset

Model	EER(%)	EER Accuracy(%)	Best Accuracy(%)
VGG-Face	4.7	95.28	95.28
Facenet	3.4	96.62	**97.45**
Facenet512	**3.15**	**96.85**	97.37
OpenFace	18.3	81.70	81.72
ArcFace	5.95	94.07	94.65
SFace	18.5	81.42	81.80

Analyzing results, we observe a wide range in performance across different models. Models such as Facenet and Facenet512 show promising EER values and high accuracy, indicating their robustness in handling facial verification. Conversely, models like OpenFace and SFace, demonstrate the challenges of achieving high accuracy in diverse CFP dataset conditions.

4.3. Concatenated clusters evaluation

The exploration of concatenated clusters is an integral part of the research, aimed at harnessing the collective strengths of multiple neural network models to enhance facial verification accuracy. This section discusses the evaluation of clusters formed by all possible combinations of six distinct models: VGG-Face, Facenet, Facenet512, OpenFace, ArcFace, and SFace. Each cluster is identified by a unique ID for ease of reference and comparative analysis.

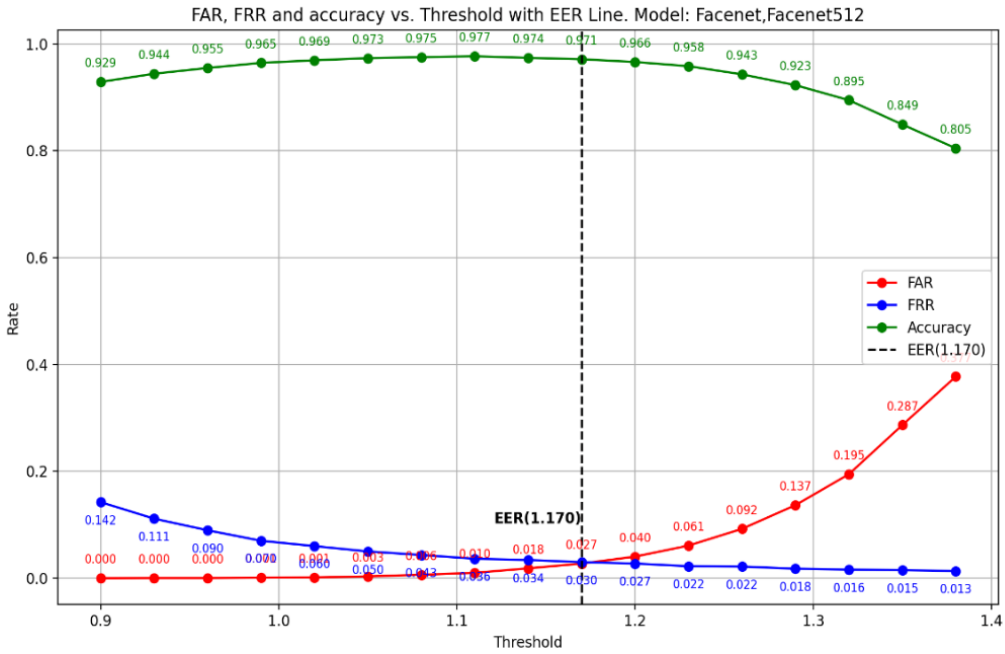

Figure 2: Visual analysis of performance metrics for Facenet and Facenet512 cluster (Cluster ID 5)

The evaluation methodology began with the determination of the EER for each cluster. EER serves as a crucial metric for assessing the balance between security and user convenience. By employing this threshold, we derived the EER-based accuracy and the best accuracy achievable across a range of thresholds, thereby quantifying the models' verification capabilities. The metrics graphs and threshold range analysis examples are shown in Fig. 2.

Following the graphical analysis, the performance results for each cluster are presented in Table 3. This table arranges the EER accuracy and the best accuracy observed for each cluster in 57 combinations.

In the evaluation of concatenated clusters, our data indicates that selected clusters achieve a marginal increase in accuracy over the highest-performing individual model, Facenet512. Specifically, clusters 5, 9, 11, 25, 26, 27, 45, and 55 demonstrate a modest enhancement, improving upon the best singular model's accuracy by approximately 0.23%. While this improvement showcases the potential advantages of model concatenation, it is crucial to consider the computational trade-offs associated with such a strategy.

5. Discussions

5.1. Insights from the results

The study's exploration into the performance of facial verification models, individually and in combined clusters, has revealed several key insights:

- **Impact of Model Pairing on Performance:** Our findings highlight a notable trend where clusters combining models with lower initial accuracy see significant performance boosts. For instance, pairing OpenFace with Sface(Cluster ID 13) resulted in a 4.33% increase in accuracy, achieving an 86.13% rate. This contrasts with clusters of high-performing models, which, on average, only show about a 0.5% improvement in accuracy. This observation suggests that strategic pairing, especially involving models with varied strengths, can effectively compensate for individual weaknesses.
- **Variable Outcomes from Mixed Model Clusters:** Not all model combinations lead to positive outcomes. In some cases, such as the cluster of Facenet and VGG-Face(Cluster ID 0), the resulting accuracy was slightly lower than that of the Facenet model on its own. This points to the complexity of model interactions within clusters and indicates that combining models does not guarantee enhanced performance and may result in suboptimal results in certain configurations.
- **Considerations on Computational Efficiency:** While some model clusters achieve minor improvements in accuracy, like Facenet with Facenet512(Cluster ID 5) with a 0.23% increase, the requisite computational resources increase significantly. This raises important considerations about the cost-benefit ratio of employing concatenated models, especially when the gains in performance are marginal compared to the added computational demand.
- **Maintaining High Accuracy and Security:** It is noteworthy that both individual and clustered models achieving the highest performance were able to maintain their accuracy without any false acceptances on the CFP dataset. This demonstrates their potential in scenarios demanding high security, where maintaining accuracy without compromising on false acceptance rates is crucial.

Table 3
Cluster models metrics on the CFP dataset

Cluster-ID	Models	EER Accuracy(%)	Best Accuracy(%)
0	VGG-Face,Facenet	95.63	95.63
1	VGG-Face,Facenet512	95.97	96.18
2	VGG-Face,OpenFace	95.33	95.33
3	VGG-Face,ArcFace	95.88	95.88
4	VGG-Face,SFace	95.55	95.55
5	Facenet,Facenet512	97.13	97.68
6	Facenet,OpenFace	95.25	95.53
7	Facenet,ArcFace	95.25	96.35
8	Facenet,SFace	95.57	96.13
9	Facenet512,OpenFace	96.87	97.28
10	Facenet512,ArcFace	96.65	97.43
11	Facenet512,SFace	97.20	97.33
12	OpenFace,ArcFace	93.53	94.83
13	OpenFace,SFace	85.83	86.13
14	ArcFace,SFace	93.67	94.62
15	VGG-Face,Facenet,Facenet512	96.13	96.35
16	VGG-Face,Facenet,OpenFace	95.57	95.63
17	VGG-Face,Facenet,ArcFace	95.93	96.03
18	VGG-Face,Facenet,SFace	95.67	95.67
19	VGG-Face,Facenet512,OpenFace	95.90	96.25
20	VGG-Face,Facenet512,ArcFace	96.13	96.52
21	VGG-Face,Facenet512,SFace	95.92	96.23
22	VGG-Face,OpenFace,ArcFace	95.68	95.98
23	VGG-Face,OpenFace,SFace	95.35	95.35
24	VGG-Face,ArcFace,SFace	95.82	95.92
25	Facenet,Facenet512,OpenFace	96.68	97.55
26	Facenet,Facenet512,ArcFace	96.65	97.53
27	Facenet,Facenet512,SFace	97.30	97.57
28	Facenet,OpenFace,ArcFace	94.90	96.13
29	Facenet,OpenFace,SFace	94.38	94.62
30	Facenet,ArcFace,SFace	95.98	96.15
31	Facenet512,OpenFace,ArcFace	96.80	97.23
32	Facenet512,OpenFace,SFace	96.57	97.23
33	Facenet512,ArcFace,SFace	96.55	97.3
34	OpenFace,ArcFace,SFace	93.53	94.37
35	VGG-Face,Facenet,Facenet512,OpenFace	95.97	96.52
36	VGG-Face,Facenet,Facenet512,ArcFace	96.20	96.72
37	VGG-Face,Facenet,Facenet512,SFace	96.12	96.47
38	VGG-Face,Facenet,OpenFace,ArcFace	95.77	96.07
39	VGG-Face,Facenet,OpenFace,SFace	95.47	95.70
40	VGG-Face,Facenet,ArcFace,SFace	95.97	96.03
41	VGG-Face,Facenet512,OpenFace,ArcFace	96.08	96.55
42	VGG-Face,Facenet512,OpenFace,SFace	95.97	96.32
43	VGG-Face,Facenet512,ArcFace,SFace	96.10	96.63
44	VGG-Face,OpenFace,ArcFace,SFace	95.63	95.88
45	Facenet,Facenet512,OpenFace,ArcFace	96.92	97.35
46	Facenet,Facenet512,OpenFace,SFace	96.78	97.43
47	Facenet,Facenet512,ArcFace,SFace	96.57	97.43
48	Facenet,OpenFace,ArcFace,SFace	94.93	95.88
49	Facenet512,OpenFace,ArcFace,SFace	96.80	97.10
050	VGG-Face,Facenet,Facenet512,OpenFace,ArcFace	96.12	96.72
51	VGG-Face,Facenet,Facenet512,OpenFace,SFace	95.98	96.53
52	VGG-Face,Facenet,Facenet512,ArcFace,SFace	96.13	96.75
53	VGG-Face,Facenet,OpenFace,ArcFace,SFace	95.73	96.07
54	VGG-Face,Facenet512,OpenFace,ArcFace,SFace	96.07	96.55
55	Facenet,Facenet512,OpenFace,ArcFace,SFace	96.95	97.35
56	VGG-Face,Facenet,Facenet512,OpenFace,ArcFace,SFace	96.08	96.67

- **Strategic Composition of Clusters for Optimal Performance:** The analysis further reveals that the most successful clusters often include a combination of the top two performing models along with a lower-performing one. This composition suggests that the diverse feature recognition capabilities of the combined models contribute to a more comprehensive analysis, thereby enhancing the overall system's performance.

5.2. Challenges encountered

Throughout this research, we encountered several challenges that impacted both the implementation of our experiments and the analysis of results:

- **Input Data Normalization**: For effective performance, each neural network model requires input data to be normalized according to the specific training data it was developed with. This normalization process involved adjusting the color space and scaling for each model to match its training conditions. We successfully applied model-specific normalization for most of the models, ensuring that the input data closely mirrored the conditions under which the models were originally trained.
- **Z-Score Normalization for Model Output Embeddings:** Given the variance in scale and distribution of embeddings across different models, a significant challenge was standardizing these embeddings for consistent comparison. By implementing Z-Score normalization on each embedding vector, the embedding was adjusted to have a mean of 0 and a standard deviation of 1. This crucial step allowed us to mitigate the disparities across model outputs.
- **Heavy Computation Without Heavy Server Resources:** The computation required for generating embeddings for 57 clusters, along with individual model evaluations, was significant. To manage this, the caching mechanism [31] was implemented for embeddings post-system setup. The strategy enabled the reuse of embeddings across different clusters and singular model experiments, saving dozens of hours in computational time.
- **Poor Accuracy for OpenFace and SFace Models:** The lower-than-expected accuracy for OpenFace and SFace models raised concerns. This may have resulted from inaccurate normalization information or deviations from the default training data used in these model weights. While this paper did not directly address enhancements to these models' accuracy, identifying the potential causes paves the way for future improvements.
- **Average Size and Resolution Dataset** While the CFP dataset was sufficiently comprehensive for our experimental purposes, its size and the variability of its data presented limitations. A larger and more diverse dataset could potentially reveal insights and issues not observed with the CFP dataset used in the study. This acknowledgment serves as a recommendation for future research directions to explore more extensive datasets for a deeper analysis.

6. Conclusions and future work

The findings from the experiments offer valuable insights into the performance synergy of employing concatenated model clusters for facial verification systems. While several clusters achieved incremental improvements in accuracy, the requisite increase in computational resources was significant. For applications prioritizing computational efficiency, singular models like Facenet or Facenet512, which provide high accuracy without substantial computational overhead, might be more advisable. Specifically, the cluster combining Facenet and Facenet512 (Cluster ID 5) presents a compelling option, marginally outperforming the accuracy of the Facenet singular model by 0.23%, achieving a 97.68% accuracy rate. This slight improvement might justify the additional computational resources in scenarios where maximizing accuracy is paramount.

In contexts where the verification system can accommodate extended inference times and has access to extended computational power, employing model clusters could be beneficial. For verification systems bound by computational and time constraints yet seeking to improve upon the accuracy provided by singular fast-inference models like OpenFace, forming clusters with other rapid-inference models offers a strategic solution. For example, pairing OpenFace with SFace led to a significant 4.33% accuracy increase over the singular OpenFace model, achieving an 86.13% accuracy rate. This strategy allows for a balanced enhancement in accuracy while maintaining essential high-speed inference capabilities, suitable for applications where both efficiency and accuracy are valued.

The exploration of concatenated model clusters in facial verification creates numerous opportunities for future research. A promising direction involves analyzing the specific features within each model's embeddings that most influence verification decisions. By identifying and prioritizing these impactful features, it may be possible to filter out less relevant or noisy features from model embeddings [32]. This approach holds potential not only for singular model systems but could notably enhance the performance of clustered model systems by focusing on the combination of the most determinant features for L2 distance calculations.

Future research could also explore the efficiency of alternative distance metrics such as Cosine [33] and L1 distances. These metrics may produce different distributions, thresholds, and ultimately accuracies for model clusters, offering new insights into the optimization of verification systems. Additionally, further investigations could evaluate how these systems scale and perform under larger, higher-quality datasets with more varying conditions, potentially uncovering benefits not observed in the current dataset.

Given that certain models are highly dependent on the alignment of facial images, integrating dynamic alignment techniques tailored to each model within a cluster could improve accuracy. This personalized approach to face alignment may optimize the performance of each model's contributions to the cluster. The initial success of combining lower-performing models with fast inference rates suggests a valuable strategy for developing efficient verification systems suited for embedded environments. Future work could focus on identifying and testing combinations of efficient models to create a verification system that balances accuracy with the computational speed necessary for real-time applications in constrained environments.

In conclusion, the decision to employ singular models or concatenated clusters should be guided by the specific requirements and constraints of the facial verification system in question. The strategic composition of clusters, balancing between computational efficiency and marginal gains in accuracy, remains a critical consideration for the deployment of robust and effective biometric authentication solutions. Additionally, the model of a decoy system based on dynamic attributes for cybercrime investigation, as proposed by Vasylyshyn et al., offers a novel approach to enhancing system security and can be integrated into future research to address emerging threats [34].

References

[1] G. Alfarsi, et al., Techniques for Face Verification: Literature Review, 2019 International Arab Conference on Information Technology (ACIT) (2019) 107–112. doi: 10.1109/ACIT47987.201 9.8990975.

[2] S. Yevseiev, et al., Models of Socio-cyber-physical Systems Security: Monograph, PC TECHNOLOGY CENTER (2023).

[3] M. Zulfiqar, et al., Deep Face Recognition for Biometric Authentication, 2019 International Conference on Electrical, Communication, and Computer Engineering (ICECCE) (2019) 1–6. doi: 10.1109/ICECCE47252.2019.8940725.

[4] S. Sengupta, et al., Frontal to Profile Face Verification in the Wild, IEEE Conference on Applications of Computer Vision (2016).

[5] Y. LeCun, Y. Bengio, G. Hinton, Deep learning, Nature 521 (2015) 436–444. doi: 10.1038/nature14539.

[6] C. Ding, D. Tao, Robust Face Recognition via Multimodal Deep Face Representation, IEEE Transactions on Multimedia 17(11) (2015) 2049–2058. doi: 10.1109/TMM.2015.2477042.

[7] M. Egmont-Petersen, D. de Ridder, H. Handels, Image Processing with Neural Networks—a review, Pattern Recognit. 35(10) (2002) 2279–2301. doi: 10.1016/S0031-3203(01)00178-9.

[8] N. Polyzotis, et al., Data Lifecycle Challenges in Production Machine Learning: A Survey. SIGMOD Rec. 47(2) (2018) 17–28. doi: 10.1145/3299887. 3299891.

[9] P. Hlushchenko, V. Dudykevych, Exploratory Survey of Access Control Paradigms and Policy Management Engines, in: Proceedings of the 7th International Workshop on Computer Modeling and Intelligent Systems, vol. 3702 (2024) 263–279.

[10] X. Wang, et al., Automated Concatenation of Embeddings for Structured Prediction, Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing 1 (2021) 2643–2660. doi: 10.18653/v1/2021.acl-long.206.

[11] R. Tronci, G. Giacinto, F. Roli, Designing Multiple Biometric Systems: Measures of Ensemble Effectiveness, Eng. Appl. Artificial Intel. 22(1) (2009) 66–78. doi: 10.1016/j.engappai.2008.04.007.

[12] C. Ding, D. Tao, Trunk-Branch Ensemble Convolutional Neural Networks for Video-Based Face Recognition, IEEE Transactions on Pattern Analysis and Machine Intelligence 40(4) (2018) 1002–1014. doi: 10.1109/TPAMI.2017. 2700390.

[13] P. Nagrath, et al., SSDMNV2: A Real-time DNN-based Face Mask Detection System Using Single Shot Multibox Detector and MobileNetV2, Sustainable Cities and Society 66 (2021). doi: 10.1016/j.scs. 2020.102692.

[14] Y. Li, et al., Improving Deep Learning Feature with Facial Texture Feature for Face Recognition, Wireless Personal Commun. 103 (2018) 1195–1206. doi: 10.1007/s11277-018-5377-2.

[15] H. Moon, C. Seo, S. Pan, A Face Recognition System Based on Convolution Neural Network Using Multiple Distance Face, Soft Comput. 21(17) (2016) 4995–5002. doi: 10.1007/s00500-016-2095-0T.

[16] Q. Yang, et al., Federated Machine Learning: Concept and Applications, ACM Trans. Intel. Syst. Technol. 10(2) (2019) 1–19. doi: 10.1145/3298981.

[17] M. Bhuiyan, S. Khushbu, M. Islam, A Deep Learning Based Assistive System to Classify COVID-19 Face Mask for Human Safety with YOLOv3, International Conference on Computing and Networking Technology (2017). doi: 10.1109/ICCNT.2017.8211490.

[18] C.-Z. Gao, et al., Privacy-preserving Naive Bayes Classifiers Secure Against the Substitution-then-comparison Attack, Inf. Sci. 444 (2018) 72–88. doi: 10.1016/j.ins.2018.02.003.

[19] M. Hanmandlu, D. Gupta, S. Vasikarla, Face Recognition Using Elastic Bunch Graph Matching, Applied Imagery Pattern Recognition Workshop (AIPR): Sensing for Control and Augmentation, IEEE (2013) 1–7. doi: 10.1109/AIPR. 2013.6749303.

[20] R. Ghiass, et al., Infrared Face Recognition: A Comprehensive Review of Methodologies and Datasets, Pattern Recognit. 47 (2014) 2807–2824. doi: 10.1016/j.patcog.2014.03.005.

[21] C. Kotropoulos, I. Pitas, Face Authentication Using Morphological Dynamic Link Architecture, Audio- and Video-based Biometric Person Authentication (1997) 169–176. doi: 10.1007/3-540-64473-6_22.

[22] V. Brydinskyi, et al., Comparison of Modern Deep Learning Models for Speaker Verification, Appl. Sci. 14(4) (2024). doi: 10.3390/app14041329.

[23] Q. Cao, et al., VGGFace2: A Dataset for Recognising Faces across Pose and Age, 13th IEEE International Conference on Automatic Face & Gesture Recognition (2018) 67–74. doi: 10.1109/FG.2018. 00020.

[24] F. Schroff, D. Kalenichenko, J. Philbin, FaceNet: A Unified Embedding for Face Recognition and Clustering, IEEE Conference on Computer Vision and Pattern Recognit. (CVPR) (2015) 815–823. doi: 10.1109/CVPR.2015.7298682.

[25] T. Baltrušaitis, P. Robinson, L.-P. Morency, OpenFace: An Open Source Facial Behavior Analysis Toolkit, IEEE Winter Conference on Applications of Computer Vision (WACV) (2016) 1–10. doi: 10.1109/WACV.2016.7477553.

[26] J. Deng, et al., ArcFace: Additive Angular Margin Loss for Deep Face Recognition, IEEE Transactions on Pattern Analysis and Machine Intelligence 44(10) (2022) 5962–5979. doi: 10.1109/TPAMI.2021. 3087709.

[27] J. Wang, et al., SFace: An Efficient Network for Face Detection in Large Scale Variations. ArXiv (2018).

[28] A. Jain, K. Nandakumar, A. Ross, Score Normalization in Multimodal Biometric Systems, Pattern Recognit. 38(12) (2005) 2270–2285. doi: 10.1016/j.pat cog.2005.01.012.

[29] V. Perlibakas, Distance Measures for PCA-based Face Recognition, Pattern Recognit. Lett. 25 (2004) 711–724. doi: 10.1016/j.patrec.2004.01.011.

[30] F. Günther, S. Fritsch, Neuralnet: Training of Neural Networks, R. J., 2(30) (2010). doi: 10.32614/RJ-2010-006.

[31] L. Fasnacht, Mmappickle: Python 3 Module to Store Memory-mapped Numpy Array in Pickle Format, J. Open Source Softw. 3(651) (2018). doi: 10.21105/JOSS.00651.
[32] H. Ye, et al., Towards Robust Neural Graph Collaborative Filtering via Structure Denoising and Embedding Perturbation, ACM Transactions on Information Systems 41 (2022) 1–28. doi: 10.1145/3568396.
[33] H. Nguyen, L. Bai, Cosine Similarity Metric Learning for Face Verification (2010) 709–720. doi: 10.1007/978-3-642-19309-5_55.
[34] S. Vasylyshyn, et al., A Model of Decoy System Based on Dynamic Attributes for Cybercrime Investigation, Eastern-European J. Enterp. Technol. 1(9(121) (2023) 6–20. doi: 10.15587/1729-4061.2023.273363.