A Modified Siamese Network for Facial Assimilation

[1]ADIL HUSSAIN, [2]ASAD ULLAH*, [3]AYESHA ASLAM, [3]AMNA KHATOON
[1]School of Electronics and Control Engineering
[2,3]School of Information Engineering
[1,3]Chang'an University
[2]Xi'an Eurasia University
Xi'an, Shaanxi Province
CHINA

Abstract: - We have proposed a face recognition model that uses modified Siamese Networks to give us a distance value that indicates whether 2 images are the same or different. We have used a pre-trained Xception CNN model based on InceptionV3 for Encoder. The Siamese networks take 3 input images (anchor, positive and negative), and Encoder encodes images to their feature vectors. The main objective of this research is to propose a model for face recognition with high accuracy and low classification process time, that is why we have implemented the model using a custom training and testing loop and loss function to be able to compute the triplet loss using three embeddings produced by Siamese Network. The model is trained using batches of triplets, and testing is performed using test triplets. The performance of the proposed model shows high accuracy. Also, the custom loop lowers the computational time during training and testing.

Key-Words: - Face Recognition, Siamese Network, Siamese Neural Network, Xception CNN, Neural Networks, Siamese Model

Received: May 23, 2022. Revised: July 18, 2023. Accepted: August 19, 2023. Published: September 19, 2023.

1 Introduction

Face recognition is a highly effective biometric technique utilized for verifying and confirming one's identity. It finds extensive application across various domains, including military operations, financial matters, public security, and everyday activities. "Face Verification" and "Face Identification" are two separate categories for the tasks related to face recognition. Each experimental scenario involves documenting a group of previously taken pictures of people in a certain gallery. The system is then shown the probe image when experimenting. The face verification technique involves computing an individual similarity index between gallery and probe photographs to evaluate whether these images belong to the same person. Recently, Convolutional neural networks (CNN) have become prevalent among researchers in computer vision, specifically in tasks like image classification [1], object detection [2], and image retrieval [3]. This trend can be attributed to the advancements in deep learning techniques, the abundance of large-scale training datasets, and the enhancements in hardware and computational capacities. To enhance the efficiency of image classification in public datasets and streamline the training process of CNNs, it is imperative to ensure an ample and diverse set of samples commensurate with the number of classes or categories present. Nevertheless, a limited number of samples can facilitate precise face recognition in specific scenarios. Conversely, the presence of numerous classes poses a significant challenge, resulting in a notable decrease in the efficiency of face recognition.

CNN possesses various applications across diverse domains, encompassing image detection, image recognition, and semantic segmentation. The utilization of deep features has proven to be effective. In this paper, we proposed a Siamese Neural Network (SNN) based on a pre-trained Xception CNN model used as an encoder using a custom loop for training and testing. The dataset folders are created in the form of sets, which are then used to create train and test triplets for training and testing of the model.

This paper will be structured as follows. Section II of the paper discusses the relevant literature and previous research in the field. In this section, the methodology employed in our study is elucidated. The implementation and results are presented in Section IV. Section V of the document encompasses a comprehensive discussion, while Section VI serves as the concluding section, encompassing the

presentation of results and outlining future research plans.

2 Related Work

The field of face recognition encompasses the processes of face identification and face authentication/verification. Face identification refers to the procedure of discerning an individual's identity by utilizing an image of their facial features.

Face recognition systems are the subject of many papers, like Duan, et al. [4], which suggested a local binary feature learning method. Xu, et al. [5] compile comprehensive reviews on sparse coding and dictionary learning algorithms in facial recognition software. When developing systems for mobile devices. Hassan and Elgazzar [6] and Oravec, et al. [7] concentrate on finding a solution while working with limited resources. Alternative methods gradually replace sparse coding, with CNN emerging as the most well-known. Utilizing CNN for face recognition [8] and [9]. To effectively train this method, CNNs have a notable limitation that demands many different images for each class. Melekhov, et al. [10] and Khalil-Hani and Sung [11] proposed using CNN with a Siamese architecture to address this issue. The amalgamation of two parallel networks is typically achieved by utilizing a cost function, which primarily functions to classify features derived from said networks. The Borghi, et al. [12] presented an advanced Siamese architecture known as JanusNet, which integrates depth, RGB, and hybrid Siamese networks through fusion. Fan and Guan [13] devised two CNN architectures that were explicitly designed to cater to different scenarios. The models underwent rigorous training on a comprehensive dataset of facial images and were subsequently enhanced through the application of embedding triplet techniques.

Ameur, et al. [14] have utilized the Weighted PCA-EFMNet deep learning feature extraction method to address issues about expression, position, illumination, and occlusion changes. Majumdar, et al. [15] introduce a new method (Auto-Encoder) for face verification called class sparsity supervised encoding (CSSE). This method uses supervised training data to teach feature representation. Xiong, et al. [16] have proposed a part-based learning method for face verification in which a convolutional fusion network (CFN) is used to extract feature representation. Chong, et al. [17] suggest a double layer block (DLB)-based metric learning method for better resolution of a pair of face images and a faster general process in face verification.

Zhang, et al. [18] presented a method for developing a new CNN and implementing it in Siamese architecture to achieve 94.8% accuracy in face recognition by training their model on a small-sample dataset LFW. To achieve facial recognition. Heidari and Fouladi-Ghaleh [19] use transfer learning in a Siamese network structure comprising two identical CNNs. The findings imply that the suggested model performs on par with sophisticated models that have been trained on substantial datasets. Furthermore, compared to techniques that are trained using datasets with a limited number of samples, it increases the accuracy of facial recognition. According to evaluations using the Labeled Faces in the Wild (LFW) dataset, the precision is 95.62%.

Wu, et al. [20] present a novel convolutional Siamese network architecture for the purpose of face recognition. Like many conventional face recognition systems, face detection is employed to determine the precise location of the face within an image. Deep learning techniques are employed to leverage facial characteristics. The process of comparing the detected faces with those stored in the database is completed. Lai and Lam [21] present a novel approach utilizing a deep Siamese network to effectively tackle the challenge of low-resolution face recognition (LRFR). The approach employed in our study involves utilizing a Siamese network to extract profound characteristics from facial images at varying resolutions. Additionally, a shared classifier is employed to facilitate the comparison of deep features from different resolutions with identical class center vectors.

3 Methodology

A facial recognition system compares a human face captured in a digital image or video frame with a pre-existing database of faces. This system is commonly utilized for the purpose of verifying user identities through ID verification services. It involves identifying and measuring specific facial characteristics extracted from a provided image. Transfer learning is a widely adopted methodology in the field of machine learning, particularly in the domain of deep learning. It involves leveraging pre-existing models to address various computer vision tasks. This approach is primarily employed in situations where there is a limited quantity of data accessible for the purpose of modeling a novel problem. Hence, it is possible to employ deep learning models that have been pre-trained on large datasets and possess established principles to construct a transfer learning model that leverages the

knowledge acquired from the prior model to address the current problem.

We present a face recognition model utilizing Siamese Networks using a pre-trained Xception convolutional neural network (CNN) as an Encoder model, with the addition of Fully Connected layers. To create the model, we use a custom training loop and a loss function that calculates the triplet loss from the three embeddings generated by the Siamese network. The model is trained to utilize batches of triplets. The Encoder encodes the images and employs the feature vectors to calculate the distance between the images. To determine the distance between the encodings of the images, the distance formula is used to determine whether the distance exceeds a specific threshold, in which case it is considered "different," or if it falls below the threshold, in which case it is considered "same."

3.1 Siamese Network Model

In contrast to a traditional CNN, the Siamese Network does not classify images based on predefined categories or labels. However, it solely calculates the distance between two specified images. If the images possess identical labels, the neural network is expected to acquire parameter knowledge, specifically the weights and biases, to minimize the distance between the two images. The magnitude of the distance should be more significant when the data points belong to distinct labels. Fig 1 illustrates the architectural design of the Siamese network.

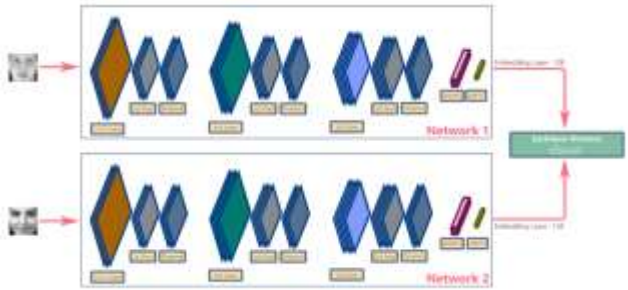

Fig. 1: Siamese Network Architecture

The network consists of six convolutional layers, each utilizing kernels of sizes 11x11, 5x5, and 3x3, respectively. Each convolutional layer in the network is associated with a Max-Pooling layer that utilizes a 2x2 pooling kernel, followed by dropout. The neural networks consist of two fully connected layers, each containing 1024 and 128 neurons, respectively. The convolutional and fully connected layers employ the Rectified Linear Unit (ReLU) as their activation function. Relu demonstrates efficacy in two distinct dimensions: There are two main objectives in the field of deep learning: (1) addressing the challenge of the vanishing gradient problem and (2) enhancing the efficiency of the training process. These two aspects are mutually reinforcing as they address the issue of gradient disappearance, resulting in improved training efficiency.

The Siamese Network utilizes three input images: anchor input, positive input, and negative input. These images are encoded using the aforementioned encoder to obtain their respective feature vectors. The aforementioned characteristics are transmitted to a distance layer, which calculates the distance between pairs consisting of an anchor and a positive sample and pairs consisting of an anchor and a negative sample. A custom layer is defined to compute the distance. Fig. 2 shows the Siamese model architecture.

Fig. 2: Siamese Model Architecture

The distance formula used in the distance layer to compute the distance is generated using Equation 1 and Equation 2:

$$n = \left\| f_i^a - f_i^p \right\|_2^2 \qquad (1)$$

$$p = \left\| f_i^a - f_i^p \right\|_2^2 \qquad (2)$$

3.2 Encoder

Encoders are responsible for transforming input images into feature vectors. The utilization of a pre-trained Xception model has been employed in our study. This model is derived from the Inception V3 model. Transfer learning allows us to cut training time and dataset size significantly. The Model is linked to Fully Connected (Dense) layers, with the final layer applying L2 Normalization to normalize the data. L2 Normalisation is a technique that modifies the dataset values so that the sum of the squares will always be up to 1 in each row. Fig. 3 shows the architecture of the Encoder.

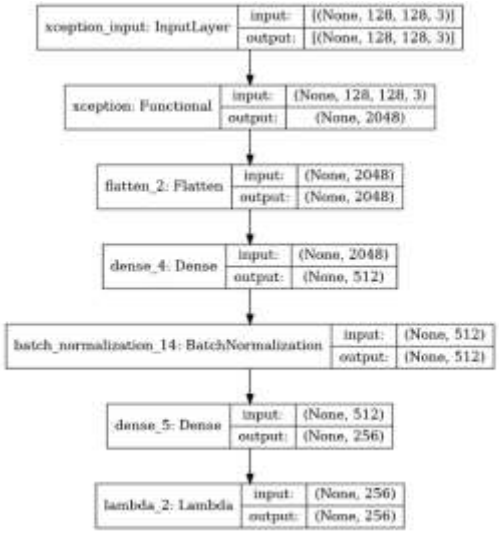

Fig. 3: Encoder Model Architecture

3.3 Dataset

The dataset consists of extracted faces obtained from the face-recognition dataset derived from the LFW Dataset. The Extracted Faces comprise facial images that have been extracted from the original images by utilizing the Haar-Cascade Face-Detection algorithm implemented in the CV2 library. Fig. 4 displays a collection of sample images.

- The dataset comprises a total of 1324 individuals, each of whom is associated with a varying number of images ranging from 2 to 50.
- The dimensions of the images are (128, 128, 3) and are encoded using the RGB color model.
- Each folder and image is assigned a numerical name, such as 0.jpg, 1.jpg.

Fig. 4: Sample Images of the Dataset

4 Implementation and Results

A facial recognition system compares a human face captured in a digital image or video frame with a pre-existing database of faces. This system is commonly utilized for the purpose of verifying user identities through ID verification services. It involves identifying and measuring specific facial characteristics extracted from a provided image.

This section includes implementing the Siamese model, including the loss function, model training, and testing. The model results are shown in graphical form in the results sub-section.

4.1 Implementation

Implementing the Siamese model involves utilizing a customized training loop and loss function to compute the triplet loss. This loss is calculated based on the three embeddings generated by the Siamese network. A metric instance with a negative connotation is established to monitor the magnitude of the loss function throughout the training process.

Training and Test sets of folders(lists) are created at the start from the dataset images. The length of the training list is 1191, whereas the length of the testing list is 133. The sample of the test list is shown in Fig. 5. The train and test lists are then used to create the training and testing triplets of (anchor, positive, negative) face data. Where positive is the same person, and negative is a different person. Fig. 6 shows the test triplets.

Fig. 5: Test List

```
Examples of triplets:
(('1308', '6.jpg'), ('1308', '7.jpg'), ('736', '0.jpg'))
(('422', '1.jpg'), ('422', '2.jpg'), ('1616', '12.jpg'))
(('1410', '3.jpg'), ('1410', '5.jpg'), ('1400', '1.jpg'))
(('1597', '3.jpg'), ('1597', '4.jpg'), ('495', '1.jpg'))
(('1580', '3.jpg'), ('1580', '4.jpg'), ('1072', '0.jpg'))
```

Fig. 6: Test Triplets

4.1.1 Triplet Loss Function:

The triplet loss function used to compute loss value is shown in the Equation 3.

$$\text{Loss} = \sum_{i=0}^{N} \left\| f_i^a - f_i^p \right\|_2^2 - \left\| f_i^a - f_i^n \right\|_2^2 + \alpha \quad (3)$$

4.1.2 Training the Model

The Siamese model is trained using batches of triplets. The training loss and additional metrics from

testing are noted at each epoch. The model weights are saved when it outperforms the previous maximum accuracy value. More metrics must be collected to evaluate the model and increase the model's accuracy. We have used only 30 epochs to avoid going over time constraints.

4.1.3 Test Function

The test function is responsible for testing the model using test triplets. Metrics, including Accuracy, are Mean, collected by the test function using prediction for train data. The test function also computes the model accuracy after testing.

4.1.4. Using the Model

After finishing the model's training, the encoder is extracted to encode the images and then get the feature vectors to compute the distance among images.

4.1.5 Classify Images

A distance formula is used to compute the distance between the encodings of the images. The distance over a certain threshold is considered "different," and below that threshold is considered "same". Two lists of images are passed to encoders twice to compute the distance value, and then the prediction value is calculated using the distance formula, considering the threshold.

4.2. Results

The Training loss of the model during the implementation is shown in Fig. 7. The training loss at the start of the training was very high, which decreased during the process. The loss value after ten epochs is better, which continues to fall and is very low.

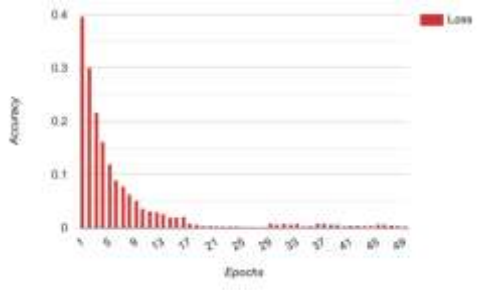

Fig. 7: Training Loss

Fig. 8 shows the test accuracy of the model, which is computed using test triplets. The model accuracy at the start of the process was a little low, which improved with time and the number of epochs. The model's accuracy enhanced at the end of the process, as shown in the graph.

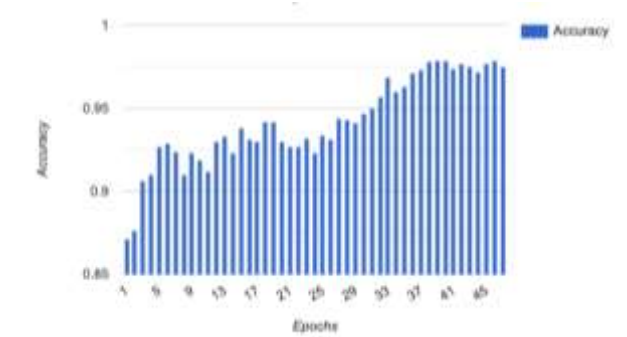

Fig. 8: Test Accuracy

The computational time for each epoch is also noted and shown in Fig. 9. The computational time at the first epoch was the highest, whereas the time was reduced during the process.

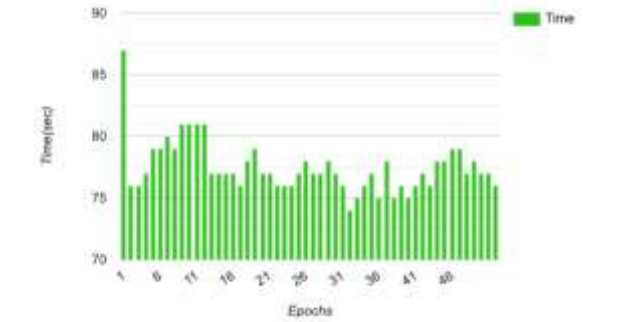

Fig. 9: Epochs Time

5 Discussion

The model performance regarding training loss is excellent because of shallow loss during training and testing. The model accuracy is also improved slowly during the implementation phase. The overall computational time of the model on a given dataset at each epoch is also good. The confusion metrics are created based on the model performance for face recognition. The metrics function calculates the model metrics, including the model's accuracy. The test accuracy achieved by the model is 95.65%, and the confusion matrix, as shown in Fig. 10, is computed to show the overall prediction accuracy of the model. The confusion metric is plotted using Predicted and Actual labels with Different and Similar values. The confusion metrics show that the model test performance for the given dataset, as the True Similar value, is 47.10%, and False Similar is

2.90%. The True Different value is 48.55%, and the False Different value is 1.45%.

Fig. 10: Confusion Matrix

The comparison of our model with some of the existing Siamese network models shows that our model performs better than the existing Siamese network models. The comparison of our model with the existing model shows that our model performance for face recognition is better, also as compared to the Siamese-VGG model, which is based on pre-trained VGG16 CNN.

Table 1: Comparison with Existing Methods

Method	Accuracy %
DLB [19]	88.50
CFN+APEM [18]	87.50
L-CSSE+KSRC [17]	92.02
SiameseFace1 [21]	94.80
Weighted PCA-EFMNET [16]	95.00
Siamese-VGG [22]	95.62
Siamese-Xception (Our)	95.65

6 Conclusion

This study presents a modified facial recognition algorithm that utilizes Siamese convolutional neural networks trained through deep learning techniques. Using a pre-trained Convolutional Neural Network (CNN) model, specifically Xception, serves as an encoder for encoding images. The training process involves utilizing batches of triplets, while the evaluation is conducted on a separate test set consisting of test triplets. The dataset utilized for training and testing purposes consists of Extracted Faces obtained from the face-recognition dataset derived from the LFW Dataset. The computation of the model's training loss and test accuracy is performed. A threshold is employed to ascertain the degree of similarity or dissimilarity between input images. The training loss of the Siamese network exhibits a limited depth, as does the corresponding test accuracy, which demonstrates an upward trend throughout the training process. The duration of each epoch is also displayed. A small dataset is used for training and testing. In the future, we will implement our model using a significant dataset that is suitable for implementation.

References:

[1] W. Rawat and Z. Wang, "Deep convolutional neural networks for image classification: A comprehensive review," *Neural computation,* vol. 29, no. 9, pp. 2352-2449, 2017.

[2] J. Han, D. Zhang, G. Cheng, N. Liu, and D. Xu, "Advanced deep-learning techniques for salient and category-specific object detection: a survey," *IEEE Signal Processing Magazine,* vol. 35, no. 1, pp. 84-100, 2018.

[3] L. Zheng, Y. Yang, and Q. Tian, "SIFT meets CNN: A decade survey of instance retrieval," *IEEE transactions on pattern analysis and machine intelligence,* vol. 40, no. 5, pp. 1224-1244, 2017.

[4] Y. Duan, J. Lu, J. Feng, and J. Zhou, "Context-aware local binary feature learning for face recognition," *IEEE transactions on pattern analysis and machine intelligence,* vol. 40, no. 5, pp. 1139-1153, 2017.

[5] Y. Xu, Z. Li, J. Yang, and D. Zhang, "A survey of dictionary learning algorithms for face recognition," *IEEE access,* vol. 5, pp. 8502-8514, 2017.

[6] G. Hassan and K. Elgazzar, "The case of face recognition on mobile devices," in *2016 IEEE wireless communications and networking conference*, 2016: IEEE, pp. 1-6.

[7] M. Oravec, D. Sopiak, V. Jirka, J. Pavlovičová, and M. Budiak, "Clustering algorithms for face recognition based on client-server architecture," in *2015 International Conference on Systems, Signals and Image Processing (IWSSIP)*, 2015: IEEE, pp. 241-244.

[8] C. Ding and D. Tao, "Trunk-branch ensemble convolutional neural networks for video-based face recognition," *IEEE transactions on pattern analysis and machine intelligence,* vol. 40, no. 4, pp. 1002-1014, 2017.

[9] Y. H. Kim, H. Kim, S. W. Kim, H. Y. Kim, and S. J. Ko, "Illumination normalisation using convolutional neural network with application to face recognition," *Electronics letters,* vol. 53, no. 6, pp. 399-401, 2017.

[10] I. Melekhov, J. Kannala, and E. Rahtu, "Siamese network features for image matching," in *2016 23rd international conference on*

pattern recognition (ICPR), 2016: IEEE, pp. 378-383.
[11] M. Khalil-Hani and L. S. Sung, "A convolutional neural network approach for face verification," in *2014 International Conference on High Performance Computing & Simulation (HPCS)*, 2014: IEEE, pp. 707-714.
[12] G. Borghi, S. Pini, F. Grazioli, R. Vezzani, and R. Cucchiara, "Face Verification from Depth using Privileged Information," in *BMVC*, 2018, p. 303.
[13] Z. Fan and Y.-p. Guan, "A deep learning framework for face verification without alignment," *Journal of Real-Time Image Processing,* vol. 18, pp. 999-1009, 2021.
[14] B. Ameur, M. Belahcene, S. Masmoudi, and A. B. Hamida, "Weighted PCA-EFMNet: A deep learning network for Face Verification in the Wild," in *2018 4th International Conference on Advanced Technologies for Signal and Image Processing (ATSIP)*, 2018: IEEE, pp. 1-6.
[15] A. Majumdar, R. Singh, and M. Vatsa, "Face verification via class sparsity based supervised encoding," *IEEE transactions on pattern analysis and machine intelligence,* vol. 39, no. 6, pp. 1273-1280, 2016.
[16] C. Xiong, L. Liu, X. Zhao, S. Yan, and T.-K. Kim, "Convolutional fusion network for face verification in the wild," *IEEE Transactions on Circuits and Systems for Video Technology,* vol. 26, no. 3, pp. 517-528, 2015.
[17] S.-C. Chong, A. B. J. Teoh, and T.-S. Ong, "Unconstrained face verification with a dual-layer block-based metric learning," *Multimedia Tools and Applications,* vol. 76, pp. 1703-1719, 2017.
[18] J. Zhang, X. Jin, Y. Liu, A. K. Sangaiah, and J. Wang, "Small Sample Face Recognition Algorithm Based on Novel Siamese Network," *Journal of Information Processing Systems,* vol. 14, no. 6, 2018.
[19] M. Heidari and K. Fouladi-Ghaleh, "Using Siamese networks with transfer learning for face recognition on small-samples datasets," in *2020 International Conference on Machine Vision and Image Processing (MVIP)*, 2020: IEEE, pp. 1-4.
[20] H. Wu, Z. Xu, J. Zhang, W. Yan, and X. Ma, "Face recognition based on convolution siamese networks," in *2017 10th International Congress on Image and Signal Processing, BioMedical Engineering and Informatics (CISP-BMEI)*, 2017: IEEE, pp. 1-5.
[21] S.-C. Lai and K.-M. Lam, "Deep siamese network for low-resolution face recognition," in *2021 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC)*, 2021: IEEE, pp. 1444-1449.

Contribution of Individual Authors to the Creation of a Scientific Article (Ghostwriting Policy)

Adil Hussain, Dr Asad Ullah carried out the implementation and the results.
Ayesha Aslam has worked on the introduction and related work.
Amna Khatoon has worked on the methodology and paper formatting.

Sources of Funding for Research Presented in a Scientific Article or Scientific Article Itself
No funding was received for conducting this study.

Conflict of Interest
The authors have no conflicts of interest to declare that are relevant to the content of this article.