International Journal of Computing and Digital Systems
ISSN (2210-142X)
Int. J. Com. Dig. Sys. 14, No.1 (Sep-2023)

http://dx.doi.org/10.12785/ijcds/140169

Face Detection and Recognition Using Siamese Neural Network

Nesrine Djawida HAMDANI[1], Nassima BOUSAHBA[2,3], Ahmed Houdaifa BOUSBAI[2] and Amina BRAIKIA[2]

[1] *Department OF Management, University Oran1 Ahmed Ben bella-ISTA, Oran, Algeria*
[2]*Computer science Department, Hassiba Benbouali University, Chlef, Algeria*
[3]*EEDIS Laboratory, Djillali Liabes University, Sidi Bel Abbes, Algeria*

Received 23 Dec. 2023, Revised 23 May 2023, Accepted 31 Jul. 2023, Published 01 Sep. 2023

Abstract:
Facial detection and recognition technology has progressed to the point where it may now play a crucial role in a variety of situations, such as when an individual is required to identify themselves in a secure environment or when they seek social services. Biometrics refers to unique qualities including face features, hand form, and even signatures. The discipline of "biometrics" has a subset called "face recognition." In order to improve the context of face recognition, a camera may be used. The goal of this study is to build a model of a Siamese neural network capable of facial recognition. This will be done by comparing two camera-taken pictures and figuring out how similar they are to one another. With the use of this technique, a pre-possessed picture may be created and supplied to the model for identification and prediction.

Keywords: Machine learning, Deep learning, Face Detection, face Recognition, Computer vision, Neural Networks, Siamese Neural Network

1. Introduction

Currently, a person must provide identification in various situations: when entering an airport, when entering military zones, when withdrawing cash from an ATM or paying in stores, and when requesting social services. This might result in a large number of codes and passwords to memorize and safeguard. Researchers are familiar with the term "biometrics," which refers to the identification and authentication process involving converting anatomical, morphological, or behavioural characteristics into digital imprints. They intend to validate an individual's uniqueness by measuring an unchanging or unquantifiable bodily trait. Behavioural biometrics focuses mostly on muscle and motion control, such as a signature, keyboard, gait, and voice. In contrast, physiological biometrics often rely on fingerprints, the face, a palm print, DNA, and the iris. These characteristics are approached as "pattern recognition" issues [1]. Facial recognition is a subset of biometric methods used for "identification" and "authentication."

For each schema, a gallery containing images of known individuals is built prior to displaying the probing image toward the system during the experiment. Intra-class variations complicate face recognition. Face identification generates a one-to-many similarity index to discover the precise identity of a probing face picture. In contrast, face verification calculates a one-by-one similarity index to assess whether two images correspond to the same individual.

The second challenge is the similarity between classes (between people or identities). For instance, twins, relatives, and even strangers with different identities may share the same physical characteristics. This suggests that variations in lighting, facial expression, location, cosmetics, haircut, age, etc., may cause an identity's appearance to vary.

Convolutional neural networks (CNN) have gained popularity in recent years thanks to the emergence of deep learning, the accessibility of large training datasets, and hardware and computational power improvements. CNNs are now one of the most commonly used models in computer vision problems like image classification [2], object detection [3], and image retrieval [4], among others. Depending on how many classes or categories are available, appropriate and diverse samples are required to streamline CNN training and improve the efficacy of image classification in publicly available datasets. The effectiveness of face recognition is significantly decreased since, despite the availability of many classes, there are occasionally not enough samples to allow for reliable face identification.

This research provides a novel method for face identification and recognition utilizing transfer learning and a Siamese Neural Network architecture, which comprises two similar CNN networks. In the suggested model, two face shots are input into the network. The network determines whether the set of images depicts the same person by

E-mail address: hamdani.nesrine@univ-oran1.dz, n.bousahba@univ-chlef.dz, houdaifabousbai07@gmail.com, aminabraikia1608@gmail.com

pulling traits from the images and producing a similarity index.

This paper is organized into seven sections. A review of the newest and most relevant studies is presented in Section 2. Siamese neural networks are described in general in Section 3. The suggested approach is outlined in Section 4. Sections 5 and 6 cover the empirical findings and assessment, and Section 8 is devoted to conclusions and suggestions for further research.

2. Related Work

When reviewing the traditional face recognition methods, we may observe how the principal component analysis (PCA)[5] method decreases computation costs by modifying the matrix. Nevertheless, external interference problems remain, like an obstruction that would erase the underlying spatial structure. Identifying the data's actual subspace structure is impossible when there is a disturbance. The classic algorithm's poor denoising impact on blurry images under non-restrictive face recognition situations, which lowers identification accuracy, is the current challenge that requires to be addressed .Tian et al. [6] innovated during the preprocessing stage by suggesting the use of a minimum representation for the picture deblurring approach. The traditional Denoising method requires the use of several relevant reference images as templates in order to obtain a satisfactory result. The authors first perform the blur removal programming method on the images using the sparse representation because they were already of the non-local self-similarity of the pictures. Once the images have been represented locally, the local image segments that come through the images' self-similarity are used as an input to the data thesaurus.

The image data-dictionary is constructed by utilizing both the principal component analysis methodology and the unsupervised clustering approach.Thus, It Can Be Considered That All Image Segments Are Similar.When interpreting noisy inputs, the old, traditional method produces a number of duplicated data, which lowers the algorithm's recognition effect and increases its computing cost. The sampling rate of a signal is determined by its sparsity and irrelevance rather than its breadth under the idea of compressed sensing (CS)[7].In order to avoid the development of significant volumes of duplicated data at the selection of samples end and to reduce saving and computing expenses, compressed sensing employs signal restoration at the output to reduce the data [8]. The target individual's face might be in the unconstrained situation in real life, which could prevent the facial recognition system from locating the person or cause it to incorrectly identify them. Zhang [9] created a middleman way for shared conversion of both visible and near-infrared light via combined restrictions of the image layer and the feature layer, through employing a GAN (Generative Adversarial Network) regarding limit the features of infrared and visible light pictures separately.

By increasing the sample size, employing adversarial networks and improving the generative network with a joint function, and via pedestrian significant point feature information to lessen the impact of the backdrop are all suggestions given by Liu et al.[10]. Zhao et al. [11] explains the fundamental characteristics of multi-pose faces based on vector machines in order to overcome the low recognition performance. Face recognition is performed in uncontrolled environments by measuring the separation between a person's facial characteristics, employing filters to separate the important features, obtaining their 3D data, and normalizing their greyscale.

Deep learning's advancement and innovation are built on Neural Networks. Convolutional Neural Networks, Siamese Neural Networks, and Deep Confidence Networks are just a few of the deep learning network models that have emerged as a result of deep network research. A deep confidence network's core is greedy layer-by-layer training of multi-layer Restricted Boltzmann Machines (RBM). To train the picture for unsupervised learning, the output of one layer is used as the input of the next layer.This improves model training's efficiency and speed, while also solving the issue of locally optimum solutions. Since then, progress in the construction of deep neural networks has accelerated.In addition, facial recognition accelerated development. Researchers Zhou [12] and colleagues claimed that integrating texture information derived from the LBP preprocessing of the image as input to the deep confidence network will improve the deep confidence network's capacity to interpret pictures. This would improve the efficiency of picture identification and assist the deep Confidence Network with its unsupervised training. Although the system is prone to error, it could be more effective at recognizing objects when there is a lot of background noise and data disturbance. Additionally, it is simple to construct an absence detection scenario. A crucial and useful network model is the Siamese Neural Network architecture.Hinton [13] combined the Siamese Network with a Convolutional Neural Network to create the Siamese Neural Network, which was then used for face recognition.

The Siamese network was first created to evaluate the legitimacy of U.S. check signatures. However, the Siamese neural network's creation was put off until 2010 due to technological constraints. Siamese neural networks were utilized to determine picture similarity in the article [14].Since then, due to the benefits of a clear model structure, a reasonable amount of parameter possibilities, and the coexistence of several network models, networks have significantly improved at finding and following targets. Xu et al. [15] implementing the Inception model with a cyclic learning rate optimization technique into the Siamese network to expedite training. Wu [16] integrated Siamese and Convolutional Neural networks utilizing the local response value normalization method to raise the eigenvalue with significant input over the eigenvalue with little input, accentuating the target's features and assuring precise target recognition.

Siamese Neural Networks and spatial transformation Networks have been combined, according to Shen et al. [17], to execute adaptive picture changes that increase accuracy and deal with image-target disruptions including distortion and rotation. They applied Siamese Networks as classifiers for image recognition. A Siamese Network's classification performance can be enhanced by manually collecting the necessary benchmark samples, but this takes more labour. Regularization is used in several studies to increase the performance and generalization of deep learning algorithms and models.In order to enhance small batch SGD through data intervention, Zheng et al.[18] developed the concept of normalization in deep learning from the perspective of consequence.In the training phase, they leverage the density difference between examples to identify anomalies, and by penalizing the data in the pre-training phase, they apply invisible regularization to improve feature boundaries. By improving the normalization impact of small batch SGDs and, as a result, the capacity for adaptation, data intervention enhances algorithm performance.According to the reference [19], they instead raise the likelihood that the network structure will generalize across the training and testing phases. Deep Learning generalization is also utilized by the spectral interference-based secondary data augmentation approach for automated modulation categorization, citing [20]. This technique strengthens the signal by including extra data in the radio transmission. The efficacy and generalizability of the algorithm are then enhanced by further data augmentation throughout the training and testing phases.

The Siamese Neural Network implemented in [21] was developed with Local Binary Pattern and Frequency Feature Perception in consideration. Based on the Siamese Neural Network, the network employs feature's frequencies perception and uniform LBP algorithms to identify human faces in open settings.

3. Siamese Neural Network

A Siamese network is a checking architecture that combines the outputs of two concurrent neural networks, each of which accepts a distinct input, to offer some prediction by computing the similarity between the two pictures.

Siamese's convolutional and pooling layers are in charge of extracting features from the picture. The input will be two photographs from the same class or other classes, and Siamese will compute the similarity score between the images of the various classes (in our case, each face is considered a different class). If the two photos are from the same class (i.e., identical), the output will be one; otherwise, it will be zero (i.e., not identical); Siamese architecture is similar to CNN architecture (convolutional and pooling layers), with the exception that the Siamese network lacks a softmax layer (only dense layers). The Siamese network still requires a significant number of APN trios for training. However, unlike classic datasets that require each image to be tagged and labelled, creating training data is significantly easier.

Max Pooling defines the maximum value for patches of a feature map in order to produce a down-sampled (pooled) feature map. It introduces a tiny level of translation invariance, which implies that most pooled output values are not significantly affected by small changes in the picture [22]. After a convolutional layer, it has commonly used. A layer that is deeply associated to the layer above it is said to be completely connected, that is, each layer's neurons are interconnected with those in the layer above it. This layer is the one that is most typically used in artificial neural network networks. In a neural network, a layer is fully linked when each input from that layer is coupled with each activation unit of the layer above it.

A. Contrastive Loss Functions

Put differently. It is a distance-based loss used to learn embeddings that take the network's output and estimate the similarity in which two similar points have a small Euclidean distance, and two different points have a large one. The loss is reduced if positive samples are encoded in identical (near) forms and negative ones in dissimilar (far) forms.

$$(1-x)\frac{1}{2}(D_w)^2 + (x)\frac{1}{2}max(0, m - D_w)^2 \quad (1)$$

D_w is calculated as follows and is referred to as the Euclidean Distance:

$$D_w = \sqrt{\{G_w(y_1) - G_w(y_2)\}^2} \quad (2)$$

G_w is the output of the network.

B. The triplet loss

The triplet loss trains the neural network by giving it three inputs: a positive image, an anchor image, and a negative image, Figure 1.

Positive Anchor Negative

Figure 1. APN trios

The positive and anchor images have a short distance and a high similarity score, however, the negative image is substantially different from the anchor image, Figure 2.

Figure 2. Triplet loss

In our case, a Siamese neural network for facial recognition is trained to comparison aspects of APN trios, the goal is to compare an anchor and a good or negative impression in terms of facial traits such as mouth shape, nose, brows, and even the distance between the eyes.

$$\lambda(B, P, M) = max(\|f(A)-f(P)\|^2-\|f(B)-f(P)\|^2+\Gamma, 0) \quad (3)$$

Where f(B), f(P), f(M) operate as the positive and negative feature embedding for the anchor. The triplet's distances between similar and dissimilar pairings are "stretched" using the margin term Γ.

C. Siamese Network for Face Recognition

Two facial images are used as input for face recognition, which will be processed by two related subnetworks of the Siamese neural network. The design, parameters, and weights of the Siamese network's two subnetworks are identical. These subnetworks will generate encoding, calculating the distance between the two inputs in Figure 3.

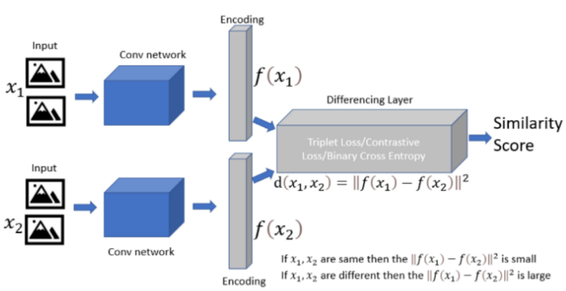


Figure 3. Siamese Neural Network

The Siamese neural network is similar to other convolutional neural networks. It takes images as input and turns their characteristics into numerical values. The distinction is visible in the output treatment. Classic CNNs fine-tune their parameters during training to correctly categorize each picture. On the other hand, the Siamese neural network is trained to compute the distance and similarity between the characteristics of two input pictures. This network does not assign an image to any of the output classes directly but rather compares inputs using a similarity function Figure 4, Table I, and Table II.

4. Methodology

This research aims to create a deep learning model that can detect and recognize human faces from photographs and videos or directly from a camera using a Siamese

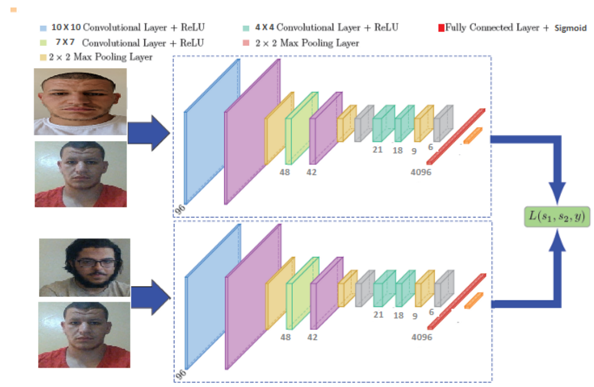


Figure 4. Siamese network for face recognition

neural network. The model will be connected to a Kivy-created application. Following that, the model will recognize faces by displaying the value. For positive verification, the model returns true; for negative verification, it returns false. Python libraries used in this project:Keras[23], TensorFlow[24], NumPy[25], Open CV[26], Google Colaboratory [27], Matplotlib[28], Kivy[29].

To create the model, we have to follow these steps:

A. Data set collection

In order to create a highly accurate model, we must first collect an acceptable dataset for our problem. The dataset consists of roughly, 8000 photos acquired from the computer science department of Hassiba Benbouali University, with each face having 10 samples taken from different perspectives with varied attitudes and accessories (hat, glasses). This dataset was gathered using a camera. Figure 5 shows a sample of one face with various positions and accessories; this dataset comprises two categories (positive and anchor), while the third class of negative data will be retrieved from a source known as "labelled faces in the wild" [30] and utilized 10000 photos from this dataset, Figure 6 and Table III summarizes the database .

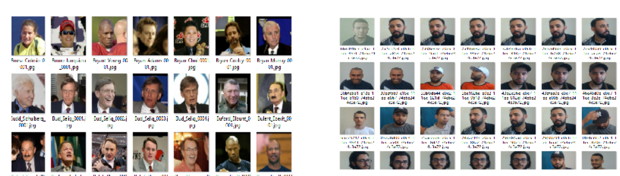


Figure 5. Samples of the dataset LFW and computer science department data

B. Data Preprocessing

Data Pre-processing is required to prepare data in a deep learning model, so first, the array() method is used to transform the data into NumPy arrays. Secondly, the value of the picture is transformed into the array from integer to float and divided by 255. As a result, each value now ranges

TABLE I. Layers of the networks

Type/stride	Filter shape	Input size
convolutional + ReLu	10*10	105*105
Max pooling	2*2	96*96
convolutional + ReLu	7*7	48*48
Max pooling	2*2	42*42
convolutional + ReLu	4*4	21*21
Max pooling	2*2	18*18
convolutional + ReLu	4*4	9*9

TABLE II. Output of Siamese Network

Type	Output
Dense + L1 dist	Feature Vector 4096
Dense + sigmoid	1*1

TABLE III. Summarize of the database

Dataset size	260.1 MB
Dataset type	categorical
Dataset partition	Anchor / Positive / Negative
Total number of instances	18000 Pictures
Images per partition	Anchor 1000 Pictures
	Positive 7000 Pictures
	Negative 10000
Average number of images for each face,	~10 pictures
Dataset sources	Labelled faces in the wild
	Computer science department - Chlef

Figure 6. Samples of one face

between 0 and 1. We must now import the dataset and use the transform's method to compress the picture size to 100 pixels in height and width for functional reasons.

A pre-processing function is made for loading the images from their directories, then resizing them and performing some scaling by converting all the image values from (0–255) to (0–1), which helps our neural network optimize a lot easier because instead of having a vast range of numbers, we only use 0 and 1, which effectively makes gradient descent easier and results in a better performing model. TensorFlow's dataset generator enables us to apply the preprocess method described over each case using our anchor, positive, and negative classes. The image should be scaled between 0 and 1. We split each pixel value, which is typically between 0 and 255, to be between 0 and 1, which scales our image.

We validate our picture by running it through our neural network with an anchor image, and a positive result should be 1 (confirmed), while a negative result should ideally be 0.

Furthermore, we use a function to process these directories, laminating our image as 100*100 pixels in three channels.

C. Model Architecture

One of the most helpful layers in Keras for deep learning applications is the embedding layer. There are two input layers, each of which leads to a Siamese neural network subnetwork (the subnetworks are identical in design, parameters, and weights and are referred to as "twin networks") that generates embeddings. The layer uses a Euclidean distance to integrate them, and the merged output is supplied into the final network. It allows us to turn each image into a fixed-length vector of a specific size. Finally, we build a distance layer and make a Siamese model, Figure 7.

```
Model: "embedding"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
input_image (InputLayer)     [(None, 100, 100, 3)]     0
_________________________________________________________________
conv2d_12 (Conv2D)           (None, 91, 91, 64)        19264
_________________________________________________________________
max_pooling2d_9 (MaxPooling2 (None, 46, 46, 64)        0
_________________________________________________________________
conv2d_13 (Conv2D)           (None, 40, 40, 128)       401536
_________________________________________________________________
max_pooling2d_10 (MaxPooling (None, 20, 20, 128)       0
_________________________________________________________________
conv2d_14 (Conv2D)           (None, 17, 17, 128)       262272
_________________________________________________________________
max_pooling2d_11 (MaxPooling (None, 9, 9, 128)         0
_________________________________________________________________
conv2d_15 (Conv2D)           (None, 6, 6, 256)         524544
_________________________________________________________________
flatten_3 (Flatten)          (None, 9216)              0
_________________________________________________________________
dense_5 (Dense)              (None, 4096)              37752832
=================================================================
Total params: 38,960,448
Trainable params: 38,960,448
Non-trainable params: 0
```

Figure 7. Layer

D. Training

The Siamese neural network training technique to initialize the network using the loss function and the Adam optimizer. After that, send the first and second images of the image pair. Using the first and second picture outputs, the loss is computing. Finally, the model's gradients are determined, and we use an optimizer to update the weights. The binary cross-entropy loss will be used to learn embeddings where two similar points have a small Euclidean distance and two different points have a sizeable Euclidean distance. The optimizer adjusts the weight parameters to minimize the loss function; Adam is a well-known gradient descent optimization approach. It is computationally efficient and uses very little memory. We can create a training loop. The model was trained on Google Colab for 50 epochs over four hours. 50 epochs imply that it will observe the dataset and the labels 50 times, Figure 8.

```
 Epoch 1/50
Tensor("binary_crossentropy/weighted_loss/value:0", shape=(), dtype=float32)
Tensor("binary_crossentropy/weighted_loss/value:0", shape=(), dtype=float32)
262/263 [============================>.] - ETA: 0sTensor("binary_crossentropy/weighted_loss/value:0", shape=(), dtype=float32)
263/263 [==============================] - 43s 163ms/step
0.85728246 0.94401914 0.9959616

 Epoch 2/50
263/263 [==============================] - 41s 156ms/step
0.1616693 0.9791073 0.99806386

 Epoch 3/50
263/263 [==============================] - 40s 152ms/step
0.025755242 0.98856056 0.99807507

 Epoch 4/50
263/263 [==============================] - 40s 150ms/step
0.21595995 0.99035215 0.99757046

 Epoch 5/50
263/263 [==============================] - 39s 150ms/step
5.019956e-05 0.9961959 0.9966698

 Epoch 6/50
 46/263 [====>.........................] - ETA: 32s
```

Figure 8. Training

5. Model Evaluation

By evaluating the accuracy and recall values for the recognition and classification performance parameters, this part assesses the model. The accuracy of each instruction differs significantly. This approach also aids in the detection of overfitting, which occurs when a function is overfitted to a limited data set. The calculated metrics are One of the most well-known measures in deep learning is the accuracy metric. It is straightforward to comprehend and apply. 1.0 0.96889135 The obtained recall after validating the model is 96.9% Figure 9 and precision are shown in Figure 10.

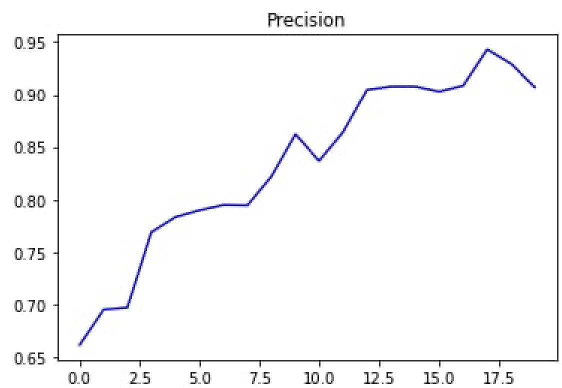


Figure 9. Recall of the Model

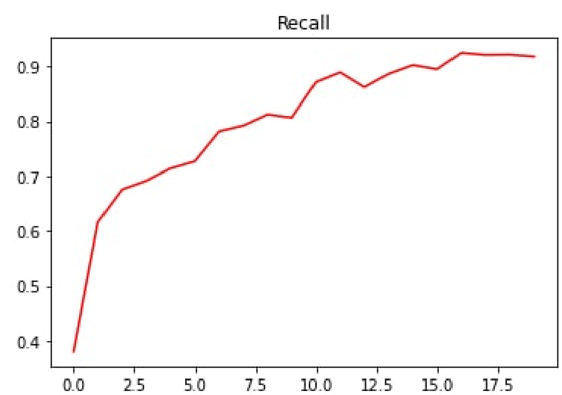


Figure 10. Precision of the Model

After finishing the engineering, training, and validation processes, the model is exported and used in a Kivy application,

In this part, we will use OpenCV to recognize faces in real-time from a live webcam broadcast. Constructing the Kivy application, We will train our model to recognize faces after it has been developed. We build a Deep Learning and TensorFlow model that replicates the results described in the paper "Siamese Neural Networks for One-shot Image Recognition." It will be possible to link it to a Kivy application and authenticate it once it has been educated. This face recognition system uses the Kivy framework to recognize faces.

This Kivy app is linked to the Siamese network. Images of various faces are input into the model, which produces a tuple with two values: ”true” represents the likelihood of proper face identification. At the same time, ”false” indicates that the system could not recognize the face.

A comment indicating the value and precision of the results is displayed at the bottom of the application’s user interface in the output. Figure 11 and Figure 12 depict positive verification, whereas Figure 13 depicts negative verification.

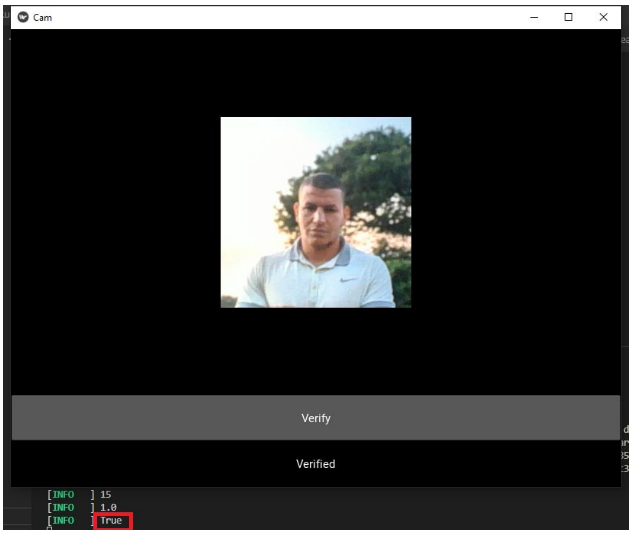


Figure 11. Face Verification

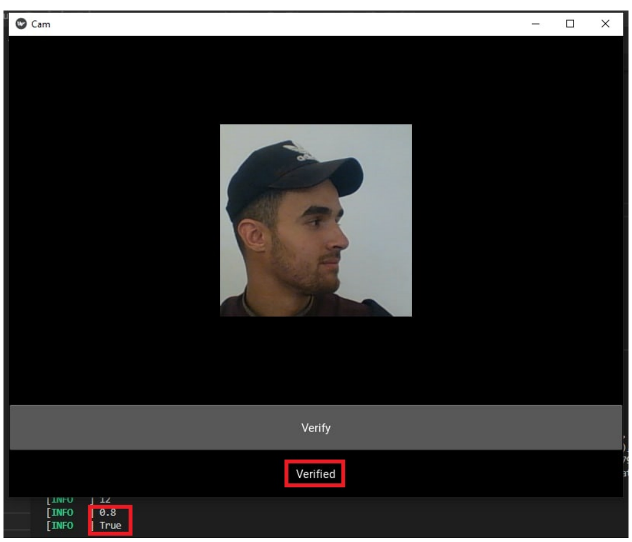


Figure 12. Positive Verification

6. Experimental Results and discussion

In this part, we execute a variety of experiments to evaluate the effectiveness of our suggested novel Siamese strategy and to compare it to cutting-edge methods, Table IV.

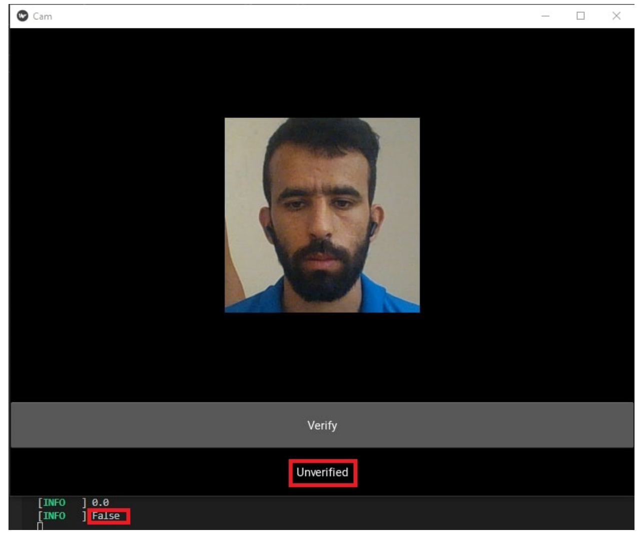


Figure 13. Negative Verification

For training, we used a mixed dataset with dense + sigmoid as activation function; the results were outstanding, with 96% accuracy.

The classifier MobileNet V2 was trained using two distinct datasets and various activation functions; the accuracy was greater utilizing face-net data and software functions. Then, we tested SRANet, which is a CNN-based architecture, with the LFW dataset, and the results were excellent, with a 95.83% rate of accuracy.

After reviewing the current state of the art and related research in face recognition, we discovered that the outcomes of various methods and settings were typically quite good, with an accuracy rate ranging from 90% to 96%. It is evident that the Siamese neural network with the ”Labelled faces in the Wild” dataset produced the best results, hence that model was chosen for this project.

Upon conducting a comparative analysis of the existing approaches, we have compiled a summary of the results in Table V, which outlines the comparison of the methods employed in our dataset. The findings indicate that the implementation of the Siamese Network approach on the dataset yielded a recognition rate of 96%. Additionally, the Resnet method [31] produced a recognition rate of 81.21%. The TripleNet method [32] resulted in a recognition rate of 84.32%, which increased to 85.52% when using the FaceNet approach [33]. However, the SN-LF method [21] was found to have a little higher recognition rate of 90.31%.

TABLE IV. Comparison with state-of-the-art approaches

Dataset	Classifier	Activation function	Accuracy
Face Net	MobileNet V2	Softmax	92%
Face Net	ResNet 50 V2	ReLu +Sigmoid	93%
Labelled faces in the wild	ResNet 50 V2	ReLu + Sigmoid	93%
Labelled Faces in the wild	SRANe	Softmax	95.83%
Labelled Faces in the wild	SRANe	Sigmoid	96 %

TABLE V. Comparison with other methods using our dataset

Methods	Recognition rate
Resnet [31]	81.21 %
TripleNet [32]	84.32%
FaceNet [33]	85.52%
SN-LF [21]	90.31%
Siamese Network	96 %

7. Conclusion and Future Work

The present study introduces a novel face verification system that utilizes a Siamese neural network. The network has been trained on facial samples collected from the computer science department at the University of Chlef, as well as the ”labelled faces in the wild” dataset. This system facilitates real-time recognition and comparison of facial similarities. The system's performance is improved through the integration of principles from both few-shot and one-shot learning. The potential of integrating this innovative technology with various other technologies in the future as a safety measure, such as smartphones or Internet of Things (IoT) devices in smart homes or cities, is noteworthy. The implementation of this technology could potentially enhance the safety and security of individuals across diverse settings.

References

[1] N. BOUSAHBA, R. ADJOUDJ, S. BELHIA, and L. CHACHOU, “Dimensionality reduction method apply for multi-view multimodal person identification,” *International Journal of Computing and Digital Systems*, vol. 12, no. 1, pp. 675–685, 2022.

[2] 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.

[3] 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.

[4] 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.

[5] A. Kaur and T. Sarabjit Singh, “Face recognition using pca (principal component analysis) and lda (linear discriminant analysis) techniques,” *International Journal of Advanced Research in Computer and Communication Engineering*, vol. 4, no. 3, pp. 308–310, 2015.

[6] L. Tian, C. Fan, and Y. Ming, “Multiple scales combined principle component analysis deep learning network for face recognition,” *Journal of Electronic Imaging*, vol. 25, no. 2, p. 023025, 2016.

[7] V. J. Barranca, “Neural network learning of improved compressive sensing sampling and receptive field structure,” *Neurocomputing*, vol. 455, pp. 368–378, 2021.

[8] J. Liu, “Research on the key technology of face recognition [d],” *Beijing University of posts and Telecommunications*, 2015.

[9] Y. Zhang, L. Tan, and J. Chen, “Cross-modality person re-identification based on joint constraints of image and feature,” *Acta Autom. Sinica*, vol. 47, no. 8, pp. 1943–1950, 2021.

[10] Y. Liu, W. Zhou, J. Liu, G.-J. Qi, Q. Tian, and H. Li, “An end-to-end foreground-aware network for person re-identification,” *IEEE Transactions on Image Processing*, vol. 30, pp. 2060–2071, 2021.

[11] Z. Zhao and G. Ni, “Automatic recognition algorithm for unconstrained multi-pose face key features under unqualified conditions,” *Comput. Sci.*, vol. 46, no. 9, pp. 250–253, 2019.

[12] L. Zhou, M. Gao, and C. He, “Study on face recognition under unconstrained conditions based on lbp and deep learning,” *Journal of Computational Methods in Sciences and Engineering*, vol. 21, no. 2, pp. 497–508, 2021.

[13] V. Nair and G. E. Hinton, “Rectified linear units improve restricted boltzmann machines,” in *Icml*, 2010.

[14] S. Zagoruyko and N. Komodakis, “Learning to compare image patches via convolutional neural networks,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 4353–4361.

[15] X.-F. Xu, L. Zhang, C.-D. Duan, and Y. Lu, “Research on inception module incorporated siamese convolutional neural networks to realize face recognition,” *IEEE Access*, vol. 8, pp. 12 168–12 178, 2019.

[16] H. Wu, “Face tracking using siamese convolutional neural networks,” *Comput. Eng. Appl.*, vol. 54, no. 14, pp. 175–179, 2018.

[17] Y. Shen, H. Wang, and Y. Dai, "Deep siamese network-based classifier and its application," *Computer Engineering & Applications*, 2018.

[18] Q. Zheng, M. Yang, J. Yang, Q. Zhang, and X. Zhang, "Improvement of generalization ability of deep cnn via implicit regularization in two-stage training process," *IEEE Access*, vol. 6, pp. 15 844–15 869, 2018.

[19] Q. Zheng, M. Yang, X. Tian, N. Jiang, and D. Wang, "A full stage data augmentation method in deep convolutional neural network for natural image classification," *Discrete Dynamics in Nature and Society*, vol. 2020, 2020.

[20] Q. Zheng, P. Zhao, Y. Li, H. Wang, and Y. Yang, "Spectrum interference-based two-level data augmentation method in deep learning for automatic modulation classification," *Neural Computing and Applications*, vol. 33, no. 13, pp. 7723–7745, 2021.

[21] C. Song and S. Ji, "Face recognition method based on siamese networks under non-restricted conditions," *IEEE Access*, vol. 10, pp. 40 432–40 444, 2022.

[22] R. S. Shankar, C. Raminaidu, V. S. Raju, and J. Rajanikanth, "Detection of epilepsy based on eeg signals using pca with ann model," in *Journal of Physics: Conference Series*, vol. 2070, no. 1. IOP Publishing, 2021, p. 012145.

[23] N. Ketkar, "Introduction to keras," in *Deep learning with Python*. Springer, 2017, pp. 97–111.

[24] B. Pang, E. Nijkamp, and Y. N. Wu, "Deep learning with tensorflow: A review," *Journal of Educational and Behavioral Statistics*, vol. 45, no. 2, pp. 227–248, 2020.

[25] T. E. Oliphant, *A guide to NumPy*. Trelgol Publishing USA, 2006, vol. 1.

[26] D. Mitra, S. Gupta, and A. Goyal, "Security system using open cv based facial recognition," in *2022 2nd International Conference on Advance Computing and Innovative Technologies in Engineering (ICACITE)*. IEEE, 2022, pp. 371–374.

[27] T. Carneiro, R. V. M. Da Nóbrega, T. Nepomuceno, G.-B. Bian, V. H. C. De Albuquerque, and P. P. Reboucas Filho, "Performance analysis of google colaboratory as a tool for accelerating deep learning applications," *IEEE Access*, vol. 6, pp. 61 677–61 685, 2018.

[28] N. Ari and M. Ustazhanov, "Matplotlib in python," in *2014 11th International Conference on Electronics, Computer and Computation (ICECCO)*. IEEE, 2014, pp. 1–6.

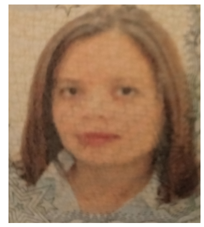
[29] R. Ulloa, *Kivy: interactive applications in python*. Packt Publishing Ltd, 2013.

[30] G. B. H. E. Learned-Miller, "Labeled faces in the wild: Updates and new reporting procedures," University of Massachusetts, Amherst, Tech. Rep. UM-CS-2014-003, May 2014.

[31] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition. arxiv preprint arxiv: 151203385," 2015.

[32] W. Ma, Y. Cui, N. Shao, S. He, W.-N. Zhang, T. Liu, S. Wang, and G. Hu, "Triplenet: Triple attention network for multi-turn response selection in retrieval-based chatbots," *arXiv preprint arXiv:1909.10666*, 2019.

[33] F. Schroff, D. Kalenichenko, and J. Philbin, "Face net: A unified embedding for face recognition and clustering'2015 ieee conference on computer vision and pattern recognition (cvpr). rakesh saini, abhishek saini, deepak agarwal," *Analysis of Different Face Recognition Algorithms*, 2014.



Nesrine Djawida Hamdani is affiliated to the Department of gestion d'entreprise et administration (University of Oran 1, Ahmed Ben Bella-ISTA,Algeria). She received her Engineering degree in Computer Science in 2008. She also received her Master's in computer and automatic from the University of Oran 1 Ahmed Ben Bella, Algeria in 2012. She obtained her PhD degree at the University of Oran1 Ahmed Ben Bella, Algeria in 2019. Her research interests include decision making, Multicriteria Analysis, Group Decision support system, web service, deep learning and machine learning.

Nassima Bousahba is a Ph.D. Student in the computer science department at the University of Sidi Bel-Abbes, Algéria. She earned a Master's (Magister) degree in computer science in 2013 from Oran University 1 Ahmed Ben Bella - Oran. Her research focuses on pattern recognition, biometrics, image processing, and artificial intelligence. She is a member of the EEDIS Laboratory, Computer Science Department.

Ahmed Houdaifa BOUSBAI received a Master's degree in computer science in 2021-2022 from Chlef University Hassiba Ben Bouali. Algeria. Her research focuses on pattern recognition, biometrics, image processing, and artificial intelligence.

Amina Braikia received a Master's degree in computer science in 2021-2022 from Chlef University Hassiba Ben Bouali, Algeria . Her research focuses on pattern recognition, biometrics, image processing, and artificial intelligence.