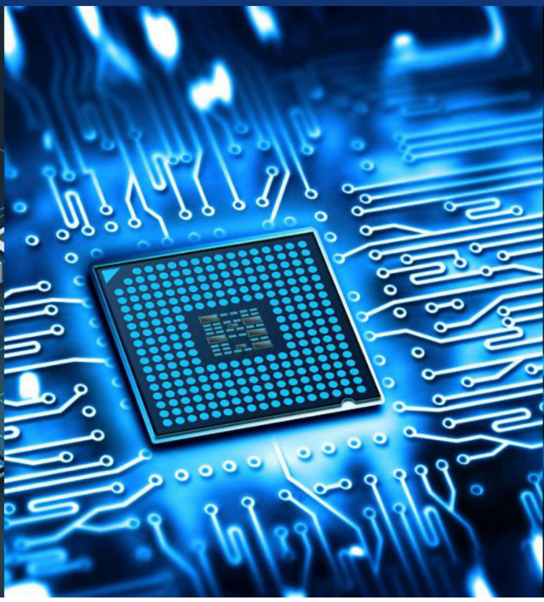
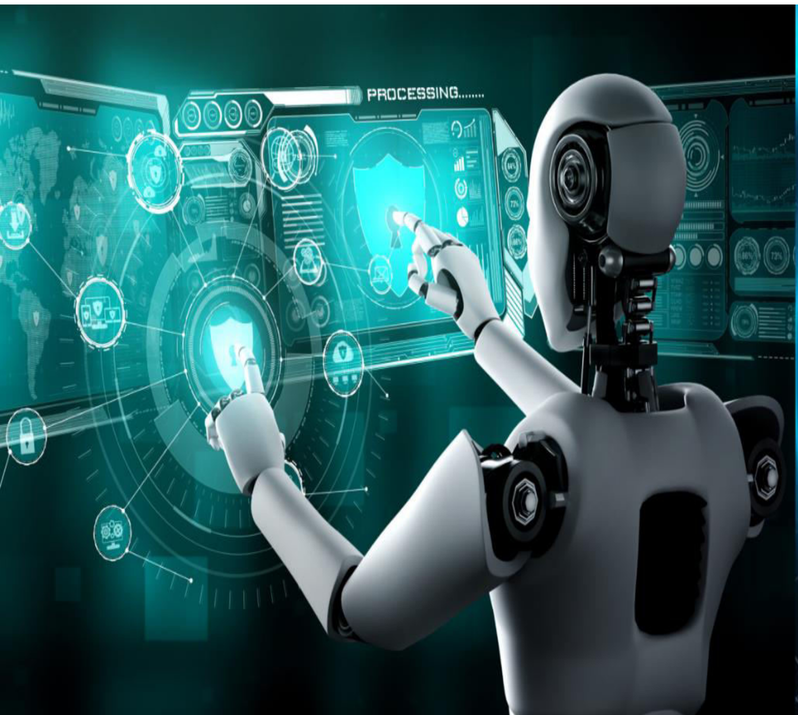
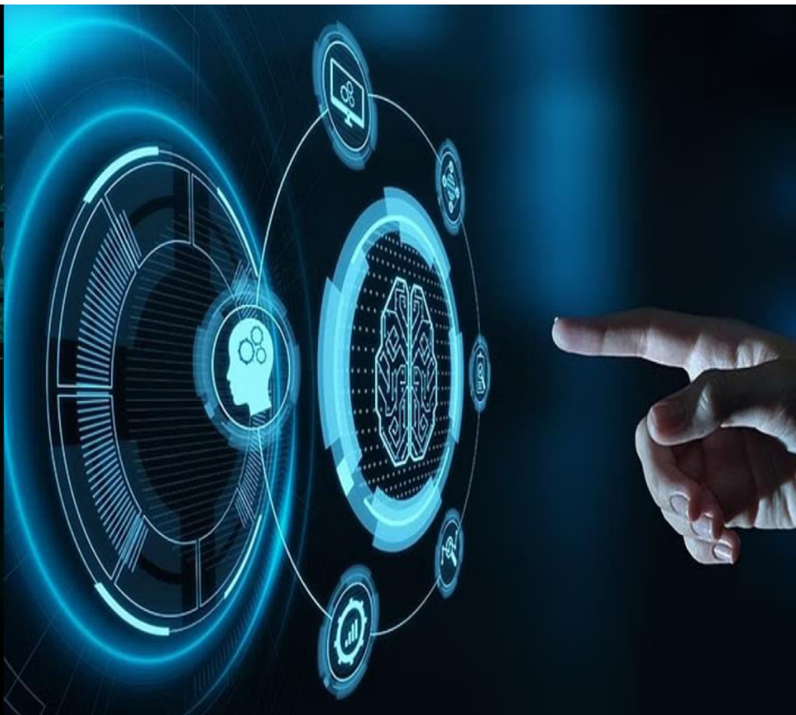
ISSN(O): 2320-9801
ISSN(P): 2320-9798



International Journal of Innovative Research in Computer and Communication Engineering

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)





Impact Factor: 8.771

Volume 13, Issue 3, March 2025

www.ijircce.com | e-ISSN: 2320-9801, p-ISSN: 2320-9798| Impact Factor: 8.771| ESTD Year: 2013|

International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

Siamese-Based Face Recognition using Convolutional Neural Networks

Rahman Touhidur[1], Tsoy Yelizaveta[1], Mithun Sheikh Rubaet Rahim[2], Kawser Md Labib[2], Zamadder Md Iqbal Hossain[1]

MSC Student, Department of Computer Science and Technology, School of Computer Science, School of Cyber Science and Engineering, Nanjing University of Information Science and Technology, Nanjing, China[1]

BSC Student, Department of Computer Science and Technology, School of Computer Science, School of Cyber Science and Engineering, Nanjing University of Information Science and Technology, Nanjing, China[2]

ABSTRACT: Given its potential applications in fields ranging from security and healthcare to biometrics and marketing, face recognition has emerged as an important area of research. Therefore, this technique has received extensive attention due to its scalability, high recognition rate, and uniqueness of face. Therefore, deep learning, particularly Convolutional Neural Networks (CNNs), became a very changed method in recent years. Siamese Neural Networks is a type of architecture used for deep learning, which is well-suited for problems where we compare and rank images, so the face recognition tasks. We trained the model, which was built with a CNN, on three types of data: positive and anchor images captured through the webcam with OpenCV and negative samples pulled from the Labeled Faces in the Wild dataset. Not only was the performance on the Test set very good (accuracy: 97%; precision: 96%; recall: 98%; F1 score: 96%) these metrics also illustrate a great performance on the model to differentiate between the matching and quickly not comparable faces with confidence. Also, its lightweight and memory-efficient system can achieve fast and accurate face detection and identification using OpenCV, which is practical for real applications.

KEYWORDS: Face recognition, Siamese Neural Networks, Convolutional Neural Network, Siamese One-Short

I. INTRODUCTION

In addition to being a biometric feature that is widely used in this sector with applications in identity management, security, etc., face recognition is one of the current research topics. Computer vision, which falls under the domain of artificial intelligence, is one of the new era's artificial intelligence fields, including deep learning and pattern recognition [1]. Since the advent of deep convolutional neural networks, a wide range of applications have been implemented; access control security, attendance, candidate identification, character recognition, and face payment are just a few of the areas where the accuracy of face recognition has risen significantly [1]. Healthcare technology and emotion recognition facilitate the automated diagnosis of diseases such as Down syndrome. And other areas of diagnosing diseases like Down syndrome. Also being used by the advertising media to gain an understanding of a customer's facial expressions and potential emotions. There are many effective and attractive face recognition algorithms; they face several issues like poor lighting, variations in pose and background, occlusion, etc [2].

There are two orders for the facial recognition challenge. The first is a one-to-one matching challenge that involves face verification. Face verification is used, for instance, when you unlock your phone with your lovely face. Another instance is when you pass through a system at some airports that uses your passport and a facial recognition scan to confirm that you are who you say you are. In order to obtain answers to the inquiry, "This person?" the alternate bone is a facial recognition task. This matching problem is one-to-numerous. Face verification is one of the major tasks that CNN-grounded techniques have been employed to accomplish; face finding is significantly improved. One-shot literacy is another method of breaking these below jobs. The image serves as an illustration of how to learn representations from illustrations. The encodings of the input image are computed in Siamese neural networks (Siamese NNs), and the encoding of an image as input of a different individual is computed using the same network without altering the network parameters. Following these computations, we can investigate how similar these two photos are [3].

www.ijircce.com | e-ISSN: 2320-9801, p-ISSN: 2320-9798| Impact Factor: 8.771| ESTD Year: 2013|

International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

Approach

Usually, we make image representations through a supervised metric-based method using Siamese neural networks and can also take advantage of that network's features for one-shot learning with no retraining [4]. We focus, however, on character recognition in our experiments, but the underlying method can be generalized to nearly any modality (Figure 1). For this sphere, we use large Siamese CNN which a) are convenient for learning general image features which are useful for making predictions as to unknown class distribution even when plenty many samples from these new distributions are available; b) are simple enough to train during standard optimization techniques on pairs try from the source data, and c) are offering a competitive method which does not rely on sphere specific knowledge by instead using very deep knowledge methods. When we want to build a one-shot image recognition type model, first of all, we have to learn a neural network that is able to make the differentiation, which is the class identity of the image pair, which is the vanilla verification job for image recognition systems.

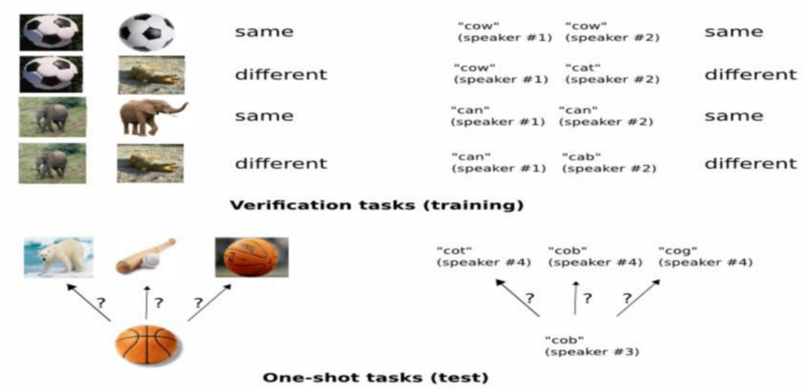


Figure 1: Siamese neural network basic structure.

Figure 1: In the paper, our general strategies are: 1) Train a model to differentiate between a collection of the same or different pairs. 2) Generalize to evaluate new categories based on learned characteristic mappings for verification.

We speculate that networks that perform well at verification should generalize to a one-shot variety. The verification model learns to identify input pairs based on their probability of being of the same class or different classes. The model can be used to estimate the new images, one for each class, in a pairwise fashion with the test image. The topmost probability for the one-shot task is assigned to the pair with the highest score corresponding to the verification network. They should, however, be enough for other dimensionalities — provided that the model has been exposed to various dimensionalities to promote separation between the learned features. The characteristics that can be discovered through the verification mannequin are enough to ascertain or deny the qualities of characters from a single collection of "rudiments" [4].

Research Objective

The primary objectives of this research are:

1. **Develop a Face Recognition System**: Using **Siamese Neural Networks** with **Convolutional Neural Networks (CNNs)** for high-accuracy face verification.
2. **Train and Evaluate the Model**: Implement the model on datasets (collected and LFW dataset) and assess performance using metrics like **accuracy, precision, recall, and F1 score**.
3. **Address Robustness Issues**: Improve the model's performance under various real-world conditions, such as lighting variations and different face angles.

II. RELATED WORK

Research Background Study

Propose an against-age variation face recognition that utilizes a discriminative model with the training of deep features [6]. In this research, a deep transfer learning CNN model, AlexNet, is used for learning high-level deep features. Then, these features are encoded with a high-dimensional codeword for image representation with a codebook. The encoding system guarantees similar code words for a person's photos taken at different times. A linear regression-based classifier is used for face recognition and probing of three datasets, one of which is available publicly on FGNET [7] proposed a face recognition system called Deep Stacked Denoising Sparse Autoencoders, which consists of a Convolutional Neural Network, autoencoder, and denoising (DS-DSA) [8]. The classification method uses a Multiclass Support Vector Machine (SVM) and Softmax classifier. The approach is tested on four publicly available datasets: ORL, Yale, and Pubfig. A deep local descriptor learning framework-based cross-modality face recognition is proposed, where both compact local information and discriminant features are learned directly from raw face patches. A face recognition system oriented to service robots was proposed, making use of the dataset collected from the webcam through many convolutional layers using a convolutional neural network to be trained with its own dataset [9].

In this paper, we have implemented the state-of-the-art face recognition methods and compared them. For the face recognition task, we will be exploring the pros and cons of using Convolutional and Siamese neural networks. We evaluate the performance of our novel face recognition system on the popular Labelled Faces in the Wild (LFW) dataset [3]. The Weighted PCA-EFMNet deep-learning feature extraction has been adopted by the authors of [10] to address issues such as expressions, spatial position, illumination, and occlusion. In [11], the authors have presented a part-based learning method to face verification, where a convolutional fusion network (CFN) is proposed to extract feature representation.

An ensemble of Support Vector Machines (SVM) and Star element analysis (PCA) on the network affair are used to complete the face verification assignment. Stated differently, the multi-stage approach solves the average face alignment problem to a 3D form model [12]. Over 4,000 people were trained by the authors to do face recognition. In this work, the Siamese network was investigated by maximizing the L1 distance between two facial features. The stylish outcome of the LFW dataset exploration was 97.35 [13] a low-cost, simple cipher network [10]. An ensemble of 25 networks was used in the study; each network focused on a distinct facial patch. We performed to a final score of 99.47 on LFW. The suggested technique does not, in fact, guarantee precise 2D/ 3D alignment in that work. Both bracket loss and verification loss are used to train the network. The triplet loss, which we will cover in the next part and employ in our face recognition system, is quite similar to the verification loss.

Some of the authors have considered other modes. Lake et al. also has some very recent work that applies a generative Hierarchical Hidden Markov mannequin for daft heathens with a Bayesian conclusion approach to identify specific words spoken by assorted audio system [14]. The only published work doing so through Bayesian networks and converting these specifications recursively is by Maas and Kemp on attributes for Ellis Island passenger data, in which the tentative distributions at every knot in the tree are made to be burdened a soft index λi [15]. This parameter is known as the attention parameter, which determines how deterministic a function is as a child of its parents. In some cases, this enables the linking of certain attributes considered critical to inferring the graph's idle variables, as such. Wu and Dennis cite one-shot knowledge in the area of way planning algorithms for robotic actuation [16] When actuation percept's are communicated to one another through a degree of knowledge, they can't be immediately reworded to new responsibilities as then situation restrictions will have transformed. By compressing the distance of this two-point embedding space introduced by the author, as a function of the coordinates of the Cartesian distortional place, the authors derived energy that can be minimized, ranging the learned action template back into a mapped action template.

III. SYSTEM DESIGN

System Functional Structure Design

With the limitation that we can only view one instance of each possible class before making a test case prediction, the bracket job is the only one that is especially intriguing. The suggested Siamese Convolutional Neural Network for One-shot face recognition is used in this paper to address one-shot literacy. The CNN model is trained using three different image types—the anchor image, the positive image, and the negative image—in order to identify the patterns of the faces. The CNN is then used to recognize faces using a one-shot model.

© 2025 IJIRCCE | Volume 13, Issue 3, March 2025| DOI: 10.15680/IJIRCCE.2025.1303091

www.ijircce.com | e-ISSN: 2320-9801, p-ISSN: 2320-9798| Impact Factor: 8.771| ESTD Year: 2013|

International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

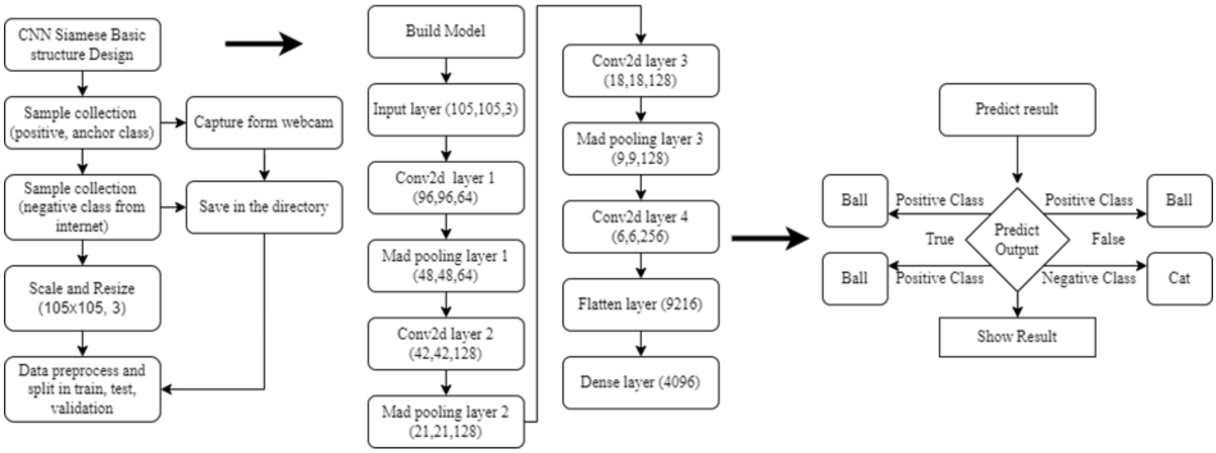


Figure 2: Siamese neural network basic structure design

Figure 2: The basic structure of the Siamese convolutional neural network shows the basic flow chart of the operational steps from data sample collecting to data set building with preprocessing, then model building steps with conventional and max-pooling layers, flatten and dense layer after that predicting result step.

Some failings of neural network technology in general, e.g., the computational burden, the over-learning of the results of operations, and the lack of original characters are well answered with the flourishing convolutional neural network. With its dispersion open area, participating weights, and time sphere or spatial sphere samples, the relegation, scaling, and deformation invariance of the results are maintained. Siamese convolutional networks are two-inflow nets, which means two inputs. It has applications in object discovery, visual shadowing, and more. For this reason, it directly uses the properties of the complicated operation to present the resultant in some manner [17]. The authors discussed how deep networks can be trained to produce such simple, abstracted representations from labeled data ([18]), which take the form of dichotomous tags on training images in order to produce dyads. [19] execution of Siamese convolutional networks in the field of Visual shadowing. This system dives into a third important field of computer vision which is Tracking, and the problem of arbitrary object Shadowing.

Data Collection

This model uses a webcam to collect the data of 2 different classes in 2 particular folders labeled as positive and Anchors using the OpenCV library, with the help of the webcam it takes 2000 images of each class by pressing different keys in the keyboard, saves those images in positive and anchor directories while taking the images it also uses the library called UUID and names the images using some random naming system in a particular range. While taking images for positive and anchor classes it also saves the images using the frame of 250 x 250 which means it can reshape the pre-defined image size, by pressing "A" on the keyboard, it will take an anchor image, by pressing "P" it will take the positive image and by pressing "Q" the image taking program will quit. For the negative class, it uses the data from the Labeled Faces in the Wild data set available from a sanctioned website. The data set provides a database of face photos for studying the problem of unconstrained face recognition [20]. The data set 5 includes far more than just images of faces scraped from the web. The name of the person pictured has been attached to each face. The data set contains two or more different prints for 1680 of these depicted individuals. These faces were detected with the Viola-Jones face sensor, the only constraint on them.

Data Loading and Preprocessing

To load the image and preprocess, we took 2000 images of each class through TensorFlow. Regarding the use of data sets, we need to take care of some important parts, such as:

1) We used Data Augmentation in the image to enhance and increase the quality of the image and image samples.
2) To preprocess the image, we need to scale and resize the images to 105 x 105 shapes so the image can fit in the CNN model, first, we need to read each image using TensorFlow IO and then decode the image into jpeg format after that preprocessing step use image resize the image to be 105 x 105 x 3 using image function from

www.ijircce.com | e-ISSN: 2320-9801, p-ISSN: 2320-9798| Impact Factor: 8.771| ESTD Year: 2013|

International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

TensorFlow framework then finally resize the images and preprocess it too. Create the labeled data set using TensorFlow. we created a leveled data set for positive and negative images because the original data cannot be fitted in the training part, so we need to convert them to a form that the computer can understand.

3) Then we created the data loader pipeline in the pipeline; we pre-processed the data and made twin data using the buffer size of 10,000. We need to use the zip function from TF data and the dataset and label those images with the respective labels in the directory.
4) Then we need to split the data into two different partitions: training and testing partition for the training partition. First, we need to build a data loader pipeline referencing the labeled dataset that we have created; to build the data loader pipeline, we will map the data through the process twin function, then we will cash the data and finally shuffle with the buffer size of 10000 to maintain the equality of the data.
5) In the training part, we are taking the total length of our dataset and then taking 70% of the data, and for the training partition, we are creating a batch of 16 and prefetch of 8 using the training data. All the training data will be used for the training model, and the data will fit in the model using batch operations.
6) In the training part, we are taking the total length of our dataset and skipping the first 70% and taking the rest of the 30% of data, and for the testing partition, then we are creating a batch of 8 and prefetch of 8 using the training data. All the testing data will be used for the testing model, and the data will fit in the model using batch operations.

IV. MODEL BUILDING AND TRAINING

Model Layers

Our baseline is a corresponding convolutional neural network with L layers where each layer has N_1 units h_(1,l) is the retirement vector in the l, layer for the first twin and h_(2,l) is the same for the second twin. Performing some rudimentary amelioration, we use simple rectifier (ReLU) units in the top L-2 layers and sigmoidal units in the remaining layers. It is a stacked architecture of convolutional layers, each utilizing a single channel with pollutants differing in size and tidily having a stride of 1. Convolutional pollutants are defined in multiples of 16 to optimize performance. The affair point charts were passed through a ReLU activation function, which was strictly followed by maximum pooling for both sludge size.

Siamese Network Forward Propagation and Activation Calculation

In the forward propagation step of the Siamese network, the model computes activations for the two input images (anchor and validation) using the following equations. These activations are essential for comparing the input pairs and determining their similarity.

The activations for each image are calculated as follows:

$$a_1^{(k)} = \max - \text{pool}\left(\max\left(0, W_l^{(k)} * h_1^{(l-1)} + b_l\right), 2\right) \qquad (1)$$
$$a_2^{(k)} = \max - \text{pool}\left(\max\left(0, W_l^{(k)} * h_2^{(l-1)} + b_l\right), 2\right) \qquad (2)$$

In these equations:

- $W_l^{(k)}$ represents the weight matrix for the k-th channel at layer l,
- $h_1^{(l-1)}$ and $h_2^{(l-1)}$ are the activations from the previous layer for the two input images,
- b_l is the bias term at layer l.

These activations are passed through the **ReLU activation function** (which is represented by the **max** operation) and then subjected to **max-pooling** with a pool size of 2. This process reduces the spatial dimensions of the feature maps while preserving important features, making the model more efficient and helping with translation invariance.

The final activations $a_1^{(k)}$ and $a_2^{(k)}$ are used for the similarity comparison between the anchor and validation images. These activations play a crucial role in determining the distance between the two images and, thus, their likelihood of being a match.

L1 Distance Function in Siamese Networks

In a Siamese network, the model learns to compare two input images and determine their similarity. One of the key concepts used for this comparison is the **L1 Distance Function**. This function calculates the absolute difference between the features extracted from the two input images.

Mathematically, the L1 Distance is represented as:

$$D(x_1, x_2) = \| f(x_1) - f(x_2) \|_1 = \sum_i | f(x_1)_i - f(x_2)_i | \qquad (3)$$

Here's how it works:

- x_1 and x_2 are the two images being compared.
- $f(x_1)$ and $f(x_2)$ represent the feature vectors of those images, which are extracted by the network.
- The function calculates the sum of the absolute differences between the corresponding feature values in these vectors.

The smaller the **L1 distance**, the more similar the two images are. Conversely, a larger L1 distance suggests that the images are quite different. This function helps the network decide if the two images belong to the same person (similar) or different people (dissimilar), making it a crucial part of the training process in a Siamese network.

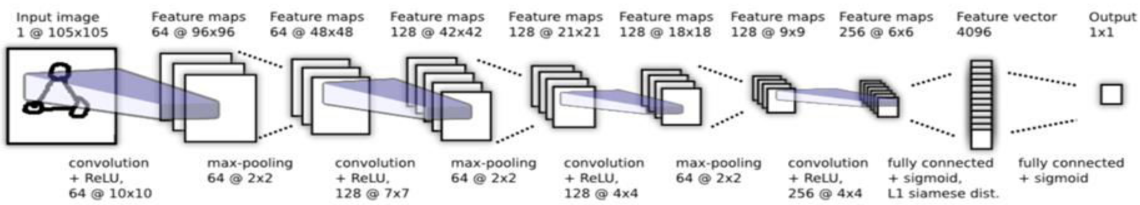


Figure 3: Different layers of the convolutional neural network.

Figure 3: For the verification task, a convolutional architecture that is suitable is chosen. After the 4096 unit fully connects layer wh1 and the L1 component-wise distance between vectors is calculated, the Siamese twin is not defined but joins incontinently.

Model Details

A Siamese convolutional neural network (CNN) with four conventions—three max polling, one flattened, and one dense layer with Relu and Sigmoid activation—is our state-of-the-art model.

Table 1: Siamese embedding model summary.

Layer names	**Output Shapes of Layers**	**Param numbers**
input_image (InputLayer)	(None, 105, 105, 3)	0
conv2d_16 (Conv2D)	(None, 96, 96, 64)	19,264
max_pooling2d_12 (MaxPooling2D)	(None, 48, 48, 64)	0
conv2d_17 (Conv2D)	(None, 42, 42, 128)	401,536
max_pooling2d_13 (MaxPooling2D)	(None, 21, 21, 128)	0
conv2d_18 (Conv2D)	(None, 18, 18, 128)	262,272
max_pooling2d_14 (MaxPooling2D)	(None, 9, 9, 128)	0
conv2d_19 (Conv2D)	(None, 6, 6, 256)	524,544
flatten_4 (Flatten)	(None, 9216)	0
dense_8 (Dense)	(None, 4096)	37,752,832
38,960,448 total parameters (148.62 MB) **Trainable parameters: 148.62 MB, 38,960,448** **Parameters that cannot be trained: 0 (0.00 B)**		

As shown in Table 1, the CNN model has the following segments:

First block: The first input convolution layer will take input of 105x105 shape with the ReLu of (64, 10x10) size. Then, the max-pooling of (64, 2x2) size will take the input from the first layer, and the feature map will have (64, 96x96) output.

Second block: The second convolution layer with the ReLu of (128, 7x7) shape will take input from the first convolution layer. And the max-pooling will be (64, 2x2) in size.

Third block: The third convolution layer with the ReLU of (128, 4x4) shape will take input from the second convolution layer. And the max-pooling will be (64, 2x2) in size.

Fourth and Final block: After receiving input from the third convolution layer, the final convolution layer with the ReLU of (256, 4x4) form will pass through a flattened layer, a dense layer, and a sigmoid activation. Additionally, there will be 38,960,448 total parameters, 38,960,448 parameters that can be trained, and 0 non-trainable parameters. The state will be Total parameters: 38,964,545, Trainable parameters: 38,964,545, and Non-trainable parameters: 0 following the addition of the L1 Distance Layer.

Table 2:Final Siamese CNN model summary

Layer	Output Shape	Param	Connected to
input_img (InputLayer)	(None, 105, 105, 3)	0	-
validation_img (InputLayer)	(None, 105, 105, 3)	0	-
embedding (Functional)	(None, 4096)	38,960,448	input_img[0][0], validation_img[0][0]
l1_dist_4 (L1Dist)	(None, 4096)		embedding[0][0], embedding[0][0]
dense_9 (Dense)	(None, 1)	4,097	l1_dist_4[0][0]
38,964,545 total parameters (148.64 MB) **Trainable parameters: 148.64 MB, 38,964,545**			

The Siamese network model in the final stage has the Input_img layer (Input layer) with the shape of (None, 105,105,3) and 0 parameters. The Validation_img layer (Input layer) with the shape of (None, 105,105,3) and 0 parameters. The Embedding layer (Functional) with the shape of (None, 4096) and 38960448 parameters and connected to input_img[0][0] and validation_img[0][0]. The Distance layer (L1Dist) with the shape of (None, 4096) and 0 parameters. and connected to embedding[2][0] and embedding[3][0]. The Dense_3 layer (Dense) with the shape of (None, 1) and 4097 parameters. and connected to distance [0][0]. Then total params will be 38,964,545, trainable params will be 38,964,545, and non-trainable params will be 0.

Model Training

The Siamese one-short CNN model is trained with 2000 samples on each class of the dataset over 15 EPOCHS the achieve a good prediction rate.

Table 3: Model training status during the training period

Topic	Start status	End status
Epoch	1	15
Loss	0.68	0.10
Recall	0.70	0.98
Precision	0.76	0.96
Accuracy	0.74	0.97

As shown in Table 1, the model's performance improved significantly over the training period. Initially, the binary cross-entropy loss was 0.68, but it decreased to 0.10 by the 15th epoch, indicating better model predictions. The recall started at 0.70 and improved to 0.98, showing that the model became much better at identifying true positives. Similarly, precision increased from 0.76 to 0.96, reflecting more reliable positive predictions. Accuracy also improved from 0.74 to 0.97, demonstrating the model's overall effectiveness in classifying both positive and negative samples.

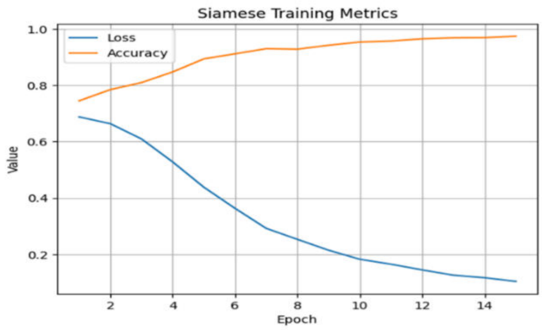


Figure 4: Siamese Training Metrics.

Figure 3 shows the training metrics of the Siamese network over 15 epochs. The blue line represents the loss, which starts high around 0.8 and steadily decreases, indicating the model is learning and improving its performance by minimizing errors. On the other hand, the orange line represents accuracy, which starts at a lower value and steadily increases, peaking towards 1.0. This indicates that as the loss decreases, the model becomes more accurate in its predictions. Overall, the graph demonstrates that the model is successfully learning and improving its performance over time, with both loss decreasing and accuracy increasing as training progresses.

V. EXPERIMENTAL RESULTS

Result explanation:

Figure 5: Siamese CNN model predictions.

In Figure 5. The image above shows the Siamese model's predictions for face verification. Each pair consists of an anchor image (on the left) and a validation image (on the right). The model's predictions are displayed with the predicted label (either 0 or 1) and the confidence score for each pair. For most pairs, the model performs exceptionally well, with high confidence scores (near 1.0), indicating correct predictions.

However, there are some instances where the model struggles, as seen with the incorrect predictions, such as in the fourth row, where the predicted score is much lower (0.43%), suggesting the model mistakenly identified the validation image. Despite a few errors, the overall performance shows that the model is strong at recognizing faces, with the majority of predictions being correct and confident.

© 2025 IJIRCCE | Volume 13, Issue 3, March 2025| DOI: 10.15680/IJIRCCE.2025.1303091

www.ijircce.com | e-ISSN: 2320-9801, p-ISSN: 2320-9798| Impact Factor: 8.771| ESTD Year: 2013|

International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

Result with Ground Truth:

Figure 6 The image above presents the Siamese model's predictions for face verification, where the Ground Truth (GT) images are shown in the first row, and the corresponding predicted images are displayed in the second row. For each pair, the model predicts whether the validation image matches the anchor image, with the predicted label (0 for no match, 1 for match) and confidence score displayed. In the first few samples, the model is highly confident with correct predictions, as seen with high scores near 1.0 for matching images (GT 1) and non-matching images (GT 0). However, some predictions are incorrect, like the ones with lower scores (0.00% and 0.03%), where the model struggled to match images with different identities. Despite these occasional errors, the model's performance remains strong overall, achieving high accuracy in matching faces. The confidence scores provide a clear indication of the model's certainty in its predictions.

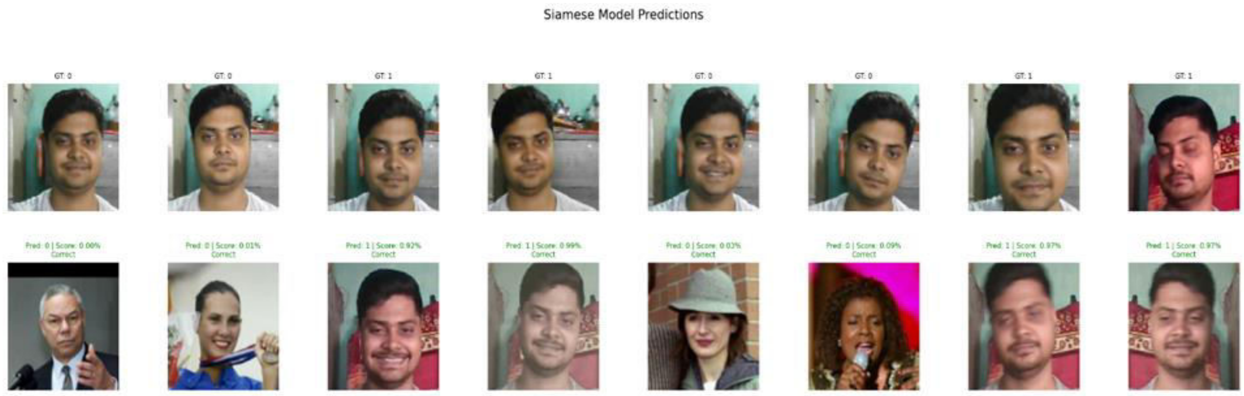


Figure 6: Siamese CNN model predictions with ground truth.

Evaluation Metrics:

Confusion matrix: Below, we can see the confusion matrix and get a proper idea of true or false positive and negative prediction rates.

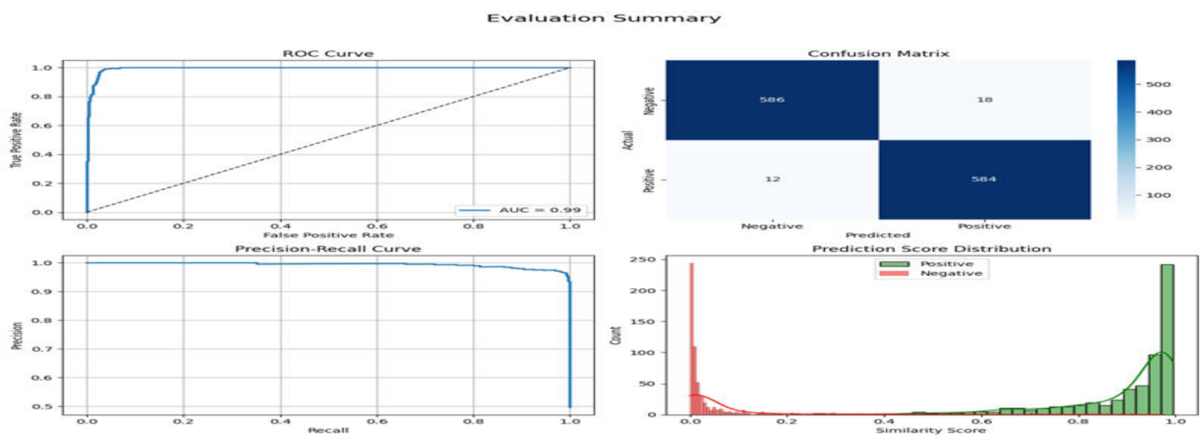


Figure 7: Evaluation Summary of the Siamese CNN Model.

Figure 7: The Evaluation Summary presented in the image includes multiple plots that summarize the performance of the model across various metrics.

1. ROC Curve: The trade-off between the True Positive Rate (TPR) and False Positive Rate (FPR) is displayed by the Receiver Operating Characteristic (ROC) curve. On a scale of 0 to 1, the Area Under the Curve, or AUC, is 0.99, which indicates that the curve is nearly perfect.
2. Confusion Matrix: The Confusion Matrix presents details about how well a model performed. For both, the model found 586 negative samples and 584 positive samples. There were, however, 18 false positives and 12

false negatives, and minor misclassifications. Large numbers down that diagonal mean a very good performance overall.

3. Precision-Recall Curve: The Precision-Recall curve and recall of the model at various thresholds. The curves are near perfect, with precision remaining high (~1.0) even at low recall levels. This means that the model is very capable of making positive predictions with a low false positive rate.
4. Prediction Score Distribution: The histogram of the score distribution depicts how confident the model on its predictions for both positive and negative classes. The positive class (green) has a clear peak around 1.0, which identifies the model as being confident about the positive matches. An alternative representation of the same effect can be seen with the plot for the negative class (red), which includes a peak near 0 and very few predictions in the ambiguous middle. This indicates that the model is confident in its predictions with a distinct separation of positive and negative scores.

Overall, these evaluation results demonstrate that the model performs exceptionally well, with high accuracy, precision, recall, and AUC, indicating its effectiveness in face verification tasks.

VI. CONCLUSION AND FUTURE WORK

The face recognition system started with collecting and preprocessing the face, then building the model, training it, and finally making real-time recognition. Then, we built and trained a convolutional neural network (CNN) using a self-built dataset and data downloaded from the internet. After tuning parameters, the model reached a test set accuracy of 97.37%. It achieved an impressive 90% average recognition rate when tested in real conditions with decent lighting. Simplicity, memory-efficient, and application-based approach are also some of the highlights of this model. It also does a good job of recognizing faces at a given angle. Some more bagging work is needed to make the lighting robust and will be part of future work to improve the system performance in other lighting conditions.

REFERENCES

1. M. Nakada, H. Wang, and D. Terzopoulos, "AcFR: Active Face Recognition Using Convolutional Neural Networks.".
2. Francis Xavier Engineering College and Institute of Electrical and Electronics Engineers, Proceedings of the 2nd International Conference on Smart Systems and Inventive Technology (ICSSIT 2019) : 27-29, November 2019.
3. W. Cui, W. Zhan, J. Yu, C. Sun, and Y. Zhang, "Face recognition via convolutional neural networks and siamese neural networks," in Proceedings - 2019 International Conference on Intelligent Computing, Automation and Systems, ICICAS 2019, Institute of Electrical and Electronics Engineers Inc., Dec. 2019, pp. 746–750. doi: 10.1109/ICICAS48597.2019.00161.
4. G. Koch, R. Zemel, and R. Salakhutdinov, "Siamese Neural Networks for One-shot Image Recognition.".
5. M. S. Shakeel and K. M. Lam, "Deep-feature encoding-based discriminative model for age-invariant face recognition," Pattern Recognit, vol. 93, pp. 442–457, Sep. 2019, doi: 10.1016/j.patcog.2019.04.028.
6. J. Zhao et al., "Look across Elapse: Disentangled Representation Learning and Photorealistic Cross-Age Face Synthesis for Age-Invariant Face Recognition," Proceedings of the AAAI Conference on Artificial Intelligence, vol. 33, no. 01, pp. 9251–9258, Jul. 2019, doi: 10.1609/aaai.v33i01.33019251.
7. P. Görgel and A. Simsek, "Face recognition via Deep Stacked Denoising Sparse Autoencoders (DSDSA)," Appl Math Comput, vol. 355, pp. 325–342, Mar. 2019, doi: 10.1016/j.amc.2019.02.071.
8. Chinese Association of Automation, M. IEEE Systems, and Institute of Electrical and Electronics Engineers, Proceedings, 2019 Chinese Automation Congress (CAC2019) : Nov. 22-24, 2019, Hangzhou, China.
9. 2018 4th International Conference on Advanced Technologies for Signal and Image Processing (ATSIP). IEEE, 2018
10. 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, Mar. 2016, doi: 10.1109/TCSVT.2015.2406191.
11. E. , et al van der Spoel, "Siamese Neural Networks for One-Shot Image Recognition," vol. 7, pp. 956–963, 2015.
12. Y. Sun, Y. Chen, X. Wang, and X. Tang, "Deep Learning Face Representation by Joint Identification-Verification."
13. B. Lake, C.-Y. Lee, J. Glass, B. M. Lake, J. R. Glass, and J. B. Tenenbaum, "UC Merced Proceedings of the Annual Meeting of the Cognitive Science Society Title One-shot learning of generative speech concepts Publication Date One-shot learning of generative speech concepts," no. 36, p. 36.

14. A. L. Maas and C. Kemp, "One-Shot Learning with Bayesian Networks."
15. Ieee and Ieee, 2012 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops.
16. Q. Li, Institute of Electrical and Electronics Engineers, and IEEE Engineering in Medicine and Biology Society, CISP-BMEI 2017 : proceedings, 2017 10th International Congress on Image and Signal Processing, BioMedical Engineering and Informatics : 14-16 October 2017, Shanghai, China.
17. C. Liu, "Probabilistic Siamese Network for Learning Representations," 2013
18. G. Hua and H. Jégou, Eds., Computer Vision – ECCV 2016 Workshops, vol. 9914. in Lecture Notes in Computer Science, vol. 9914. Cham: Springer International Publishing, 2016. doi: 10.1007/978-3-319-48881-3.
19. G. B. Huang, M. Ramesh, T. Berg, and E. Learned-Miller, "Labeled Faces in the Wild: A Database for Studying Face Recognition in Unconstrained Environments", Accessed: Mar. 25, 2025. [Online]. Available: http://vis-www.cs.umass.edu/lfw/.

INTERNATIONAL STANDARD SERIAL NUMBER INDIA

INNO SPACE

SJIF Scientific Journal Impact Factor

doi crossref

निस्केयर NISCAIR

INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN COMPUTER & COMMUNICATION ENGINEERING