Journal of Innovative Image Processing (ISSN: 2582-4252)
www.irojournals.com/iroiip/

Face Recognition System for Real Life Applications using Augmentation Techniques

Namrata Mishra[1], H. Karthikeyan[2*], Tanya Jayanti[3]

[1, 3]Final Year Students, Department of Computer Science and Technology (CSE), SRM Institute of Science and Technology, Chennai-603203, India.
[2] Assistant Professor, Department of Computer Science and Technology (CSE), SRM Institute of Science and Technology, Chennai-603203, India.
E-mail: [1]namratamishra2615@gmail.com, [2]karthikh@srmist.edu.in, [3]tanyajayanti01@gmail.com

Abstract

A facial recognition system may recognise a person by comparing his face to the ones in a database of faces taken from a digital image or video frame. It is primarily used to verify the identity of users by analysing facial features from an image. According to several studies, facial recognition technology has made enormous strides with the growth of science and technology, but there is still opportunity for its improvement in practical use. Some issues like rotation, occlusion, and meta learning approach can be handled for improved model accuracy by using 3D technologies like image augmentation to supplement 2D images. Today's facial recognition technology can be used for more than only identifying people; it can also be used for security purposes in residential, business, and commercial applications. Due to its convenience, face recognition technology is widely employed in the financial sector as well. This research proposes face recognition for real life applications using Siamese network and Deep Neural Network, Facenet.

Keywords: AI (Artificial Intelligence), Siamese Network, Facenet, One-shot Learning

1. Introduction

The several business sectors utilising and implementing facial recognition technology include:

1) To secure their facilities, security corporations use facial recognition technology.

DOI: https://doi.org/10.36548/jiip.2023.1.004
Received: 16.03.2023, received in revised form: 16.04.2023, accepted: 26.04.2023, published: 06.05.2023

2) In immigration checkpoints, facial recognition is utilised to provide more sophisticated border control.

3) Ride-sharing companies can use facial recognition technology to ensure that the right drivers are picking up the right passengers.

4) To identify individuals in rescue operations.

A computer programme called face detection locates and measures the human face in any given (digital) image. It can be seen as a specific case of object-class identification where the task is finding and figuring out the sizes of all objects in an image that belong to a certain class. One way to think of face localization is as a more generalised version of face detection. Finding the position and dimension of a known quantity of face is the goal of face localization (usually one).

There are four steps in the methodology:

1) Face Detection

2) Face Extraction

3) Utilizing Siamese network and Deep Neural Network Facenet, the model is trained using the extracted data as well as additional created data from the existing data.

4) Meta Face Recognition: To put forth a solution that shows resilience against image variations like rotated images and occlusion and increase the accuracy of face identification.

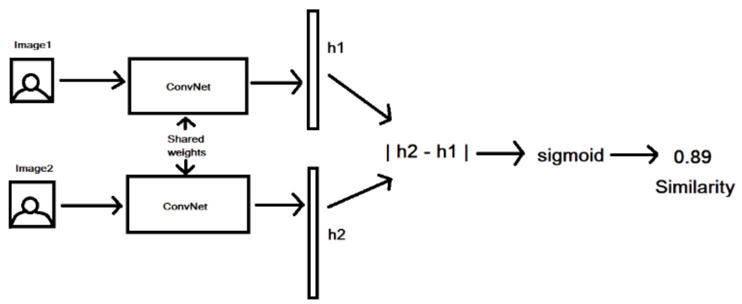

Figure 1. Siamese Network Architecture

2. Methodology

The methodology is divided into parts, namely:

- Face Detection and Capture
- Model Training
- Face Similarity Match

A. Face Detection and Capture

-> Face detection is the first step in locating and identifying faces in images. Based on the subject's facial characteristics, analogue data is converted to digital data via face detection.

-> The picture is first snapped.

-> The obtained face is then recognized.

B. Model Training

->The model is trained using anchor and positive images that were captured and partitioned into test and trained data.

-> Siamese Network, a metric spaced Meta learning algorithm and Facenet is employed to train the proposed model.

-> By contrasting an anchor image with both a positive sample and a negative sample, the network is trained.

C. Face Similarity Match

->The face match establishes whether the given face corresponds to any face in the dataset.

-> A live video feed of the face is captured. The identifying process then uses the dataset.

->The Siamese neural network calculates the encodings of the captured input image.

->Then, without altering the network's configuration, a different person's image is obtained as input using the same network.

-> Its encoding is established after that.

->These calculations are followed by an analysis of the similarity score between the two images.

3. System Architecture

The system's architecture is shown below.

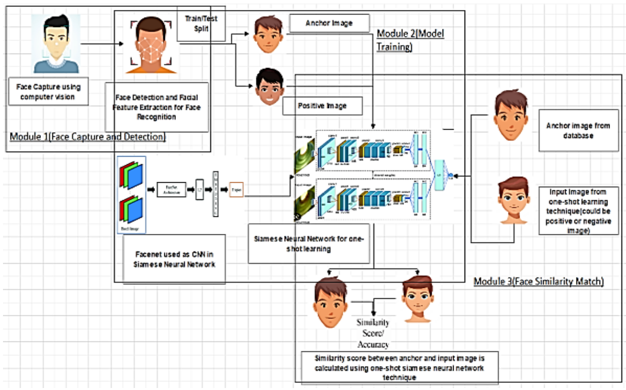

Figure 2. System Architecture

An architecture diagram shows how a software system's components are physically implemented. It displays the overall architecture of the software system as well as the constraints, connections, and divisions between the various components. Diagrams greatly facilitate information retention. They can aid with comprehension and memory as well.

4. Implementation

The implementation process is divided into the following steps:

A. Importing all the necessary Libraries and Photo Capture

- In Jupyter notebook, firstly all the necessary header files are imported. The accompanying data is for extraordinary libraries like TensorFlow, sklearn, and matplotlib.
- The data for negative images is downloaded.
- Then the images for anchor images and positive images for similarity checking are captured.

B. Face detection and capture using Computer Vision

Using Computer Vision (CV), face is identified from the captured image. The whole idea for the CV is to make an automatic replica of the human eye, which is more efficient than the human eye. The steps involved are:

1. Generating index for each label
2. Generating image pairs and labels
3. Get the data directory paths
4. Load the images and labels
5. Then visualize positive and negative images.

C. Network Architecture

The network architecture comprises of:

- Embedding Layer
- Siamese Network

1) Embedding Layer

The embedding layer comprises of three sub layers.

2) Siamese Network

Two identical neural networks, each taking one of the two input pictures, make up a Siamese network. The contrastive loss function, which assesses how similar the two images are, receives the final layers from both networks.

The different steps are:

1. The Siamese neural network determines encodings for the input picture that has been captured.
2. After that, an image of a different individual is given into the same network without changing any of its settings.
3. After that, its encoding is chosen.

4. After these computations, it can be determined if the two photos are comparable.

```python
# Siamese L1 Distance Class
class L1Dist(Layer):
    def __init__(self, **kwargs):
        super().__init__()

    # Similarity Clarification
    def call(self, input_embedding, validation_embedding):
        return tf.math.abs(input_embedding - validation_embedding)
```

```python
l1 = L1Dist()
```

```python
# Siamese Model
def make_siamese_model():

    # Anchor Image
    input_image = Input(name="input_img", shape=(100, 100, 3))

    # Validation Image
    validation_image = Input(name="validation_img", shape=(100, 100, 3))

    # Siamese distance components
    siamese_layer = L1Dist()
    siamese_layer._name = "distance"
    distances = siamese_layer(embedding(input_image), embedding(validation_image))

    # Classification Layer
    classifier = Dense(1, activation="sigmoid")(distances)

    return Model(inputs=[input_image, validation_image], outputs=classifier, name="SiameseNetwork")
```

```python
siamese_model = make_siamese_model()
```

Figure 3. Siamese Network

D. Training

At this stage, the model is trained using the characteristics that were extracted in the previous step as input. By comparing an anchor image with both positive and negative samples, the network is trained. The anchor images and the positive image must differ slightly from one another, but the anchor image and the negative image must differ greatly.

```python
def train(data, EPOCHS):
    # Loop through epochs
    for epoch in range(1, EPOCHS+1):
        print('\n Epoch {}/{}'.format(epoch, EPOCHS))
        progbar = tf.keras.utils.Progbar(len(data))

        # Creating a metric object
        r = Recall()
        p = Precision()

        # Loop through each batch
        for idx, batch in enumerate(data):
            # Run train step here
            loss = train_step(batch)
            yhat = siamese_model.predict(batch[:2])
            r.update_state(batch[2], yhat)
            p.update_state(batch[2], yhat)
            progbar.update(idx+1)
        print(loss.numpy(), r.result().numpy(), p.result().numpy())

        # Save checkpoints
        if epoch % 10 == 0:
            checkpoint.save(file_prefix=checkpoint_prefix)
```

```python
EPOCHS = 50
```

```python
train(train_data, EPOCHS)
```

Figure 4. Training

E. Testing

Following these calculations, the similarity score is used to determine whether the two images are comparable or not.

5. Output

Siamese Network and Facenet are used to construct a face recognition model, which accepts images from various perspectives (varying angles). Occlusion, like as wearing a mask, is also taken into account. Low resolution issue is also taken into account. Siamese Network - Metric Spaced Meta Learning Algorithm, calculates a result score based on how closely input images from the dataset match those that were actually recorded, as demonstrated in the preceding slides. One-shot learning is put into practice. The model's accuracy has increased when compared to old machine learning models such as PCA and SVM.

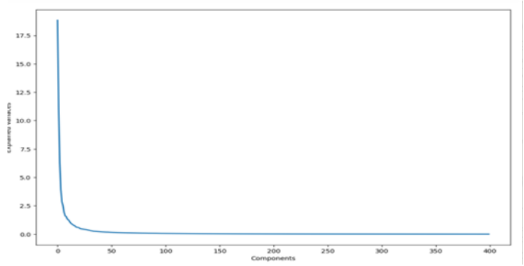

Figure 5. Variation graph of face recognition system using PCA

```
print("Accuracy score:{:0.2f}".format(metrics.accuracy_score(y_test, y_pred)*100))

========== SVM RESULT ==========
Accuracy score:91.67
```

Figure 6. Accuracy of face recognition system using PCA

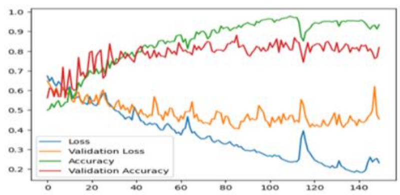

Figure 7. Accuracy vs Loss Graph of Siamese network- based facial recognition

```
print(np.max(history.history["accuracy"])*100)

97.81249761581421
```

Figure 8. Accuracy of Siamese network-based facial recognition

5. Results and Discussion

A face recognition model is created using Siamese Network - Metric Spaced Meta Learning Algorithm, which offers result score based on similarities between input images from the dataset and collected images as illustrated in the preceding slides, and Facenet. Learning from mistakes is used in practise, and a greater precision is attained.

Here,

1) An input image is contrasted with images from the test images dataset.

2) To calculate the similarity score, the input image and the test images are compared.

3) The output is then given as the last image match.

A. Module 1: Face Detection and Capture

Figure 9. Face Detection and Capture

So, the face for detection is captured using computer vision.

B. Module 2: Face Similarity Match

The similarity between the captured image and existing images from the dataset is matched.

Figure 10. Face Similarity Score

```
r = Recall()
p = Precision()

for test_input, test_val, y_true in test_data.as_numpy_iterator():
    yhat = siamese_model.predict([test_input, test_val])
    r.update_state(y_true, yhat)
    p.update_state(y_true,yhat)

print(r.result().numpy(), p.result().numpy())
```

```
1.0 1.0
```

Figure 11. Precision and Recall

6. Conclusion

This concludes the use of Siamese Network and Facenet in the face recognition model. It catches an image from many angles. Occlusion, such as wearing a mask, is also taken into account. Low Resolution issue is also taken into account. Utilizes the metric spaced meta learning method Siamese Network, which calculates a result score based on the similarity between input images from the dataset and collected images, as demonstrated in the preceding slides. The use of one-shot learning is made. Greater precision is attained. There is still room for improvement, such as increasing the model's learning efficiency and accuracy. This is because we can recognise a person from a captured image by using even a single image of them as an input in the dataset, and the pre-existing model, without having to train the model from scratch with all the images. Model’s use is highly specific as one-shot learning eliminates the need of training a billion images to a model. There is an element of variance, i.e., one-shot learning is highly susceptible to variance.

References

[1] S. K. Roy, P. Kar, M. E. Paoletti, J. M. Haut, R. Pastor-Vargas and A. Robles-Gómez, "SiCoDeF² Net: Siamese Convolution Deconvolution Feature Fusion Network for One-Shot Classification," in IEEE Access, vol. 9, pp. 118419-118434, 2021, doi: 10.1109/ACCESS.2021.3107626

[2] M. Heidari and K. Fouladi-Ghaleh, "Using Siamese Networks with Transfer Learning for Face Recognition on Small-Samples Datasets," 2020 International Conference on Machine Vision and Image Processing (MVIP), 2020, pp. 1-4,doi:10.1109/MVIP49855.2020.9116915.

[3] Jiménez-Bravo, D.M., Murciego, Á.L., Mendes, A.S., Silva, L.A., Iglesia, D.H.D.L. (2022). An Analysis of One-Shot Augmented Learning: A Face Recognition Case Study. In: de Paz Santana, J.F., de la Iglesia, D.H., López Rivero, A.J. (eds) New Trends in Disruptive Technologies, Tech Ethics and Artificial Intelligence. DiTTEt 2021. Advances in Intelligent Systems and Computing, vol 1410. Springer, Cham. https://doi.org/10.1007/978-3-030-87687-6_6

[4] Arulprakash, A., Meraz, M., Javed, M. (2021). One Shot Learning Based Human Tracking in Multiple Surveillance Cameras. In: Singh, S.K., Roy, P., Raman, B., Nagabhushan, P. (eds) Computer Vision and Image Processing. CVIP 2020.

Communications in Computer and Information Science, vol 1378. Springer, Singapore. https://doi.org/10.1007/978-981-16-1103-2_26

[5] V. Dalai and K. V. Kadambari, "Face Identification using visible eye region via vanilla CNN and Siamese Networks," 2022 2nd International Conference on Advance Computing and Innovative Technologies in Engineering (ICACITE), Greater Noida, India, 2022, pp. 2597-2600, doi: 10.1109/ICACITE53722.2022.9823668.

[6] Zhengfang, He & Su, Weibin & Bi, Zhimin & Wei, Ming & Dong, Yingguo & Xu, Gang. (2019). The Improved Siamese Network in Face Recognition. 443-446. 10.1109/ICICAS48597.2019.00099.

[7] W. Cui, W. Zhan, J. Yu, C. Sun and Y. Zhang, "Face Recognition via Convolutional Neural Networks and Siamese Neural Networks," 2019 International Conference on Intelligent Computing, Automation and Systems (ICICAS), 2019,pp.746-750,doi: 10.1109/ICICAS48597.2019.00161.

[8] C. Song and S. Ji, "Face Recognition Method Based on Siamese Networks Under Non-Restricted Conditions," in IEEE Access, vol. 10, pp. 40432-40444, 2022, doi: 10.1109/ACCESS.2022.3167143.

[9] L. Ouannes, A. B. Khalifa and N. E. B. Amara, "Siamese Network for Face Recognition in Degraded Conditions," 2022 6th International Conference on Advanced Technologies for Signal and Image Processing (ATSIP), Sfax, Tunisia, 2022, pp. 1-6, doi: 10.1109/ATSIP55956.2022.9805878.

[10] P. Rama Devi, R. Yashashvini, G. Navyadhara and M. Ruchitha, "Similar Face Detection for Indian Faces using Siamese Neural Networks," 2021 2nd International Conference for Emerging Technology (INCET), 2021, pp. 1-5, doi: 10.1109/INCET51464.2021.9456199.

[11] P. Satyagama and D. H. Widyantoro, "Low-Resolution Face Recognition System Using Siamese Network," 2020 7th International Conference on Advance Informatics: Concepts, Theory and Applications (ICAICTA), 2020, pp. 1-6, doi: 10.1109/ICAICTA49861.2020.9428885.

[12] A.Stalin, A. Sha, A. S. Kumar, S. Nandakumar and G. Gopakumar, "Face Recognition at varying angles from distant CCTV Footage using Siamese Architecture," 2022 3rd International Conference for Emerging Technology (INCET), 2022, pp. 1-6, doi: 10.1109/INCET54531.2022.9824723.