A Siamese Neural Network-Based Face Recognition from Masked Faces

Chapter · January 2022

DOI: 10.1007/978-3-030-96040-7_40

3 authors, including:

Rajdeep Chatterjee
KIIT University
89 PUBLICATIONS **810** CITATIONS
SEE PROFILE

Soham Roy
Jis Group Educational Initiatives
1 PUBLICATION **7** CITATIONS
SEE PROFILE

A Siamese Neural Network-based Face Recognition from Masked Faces

Rajdeep Chatterjee[*]
School of Computer Engineering
KIIT Deemed to be University
Bhubaneswar-751024, Odisha, India
cse.rajdeep@gmail.com

Soham Roy
School of Computer Engineering
KIIT Deemed to be University
Bhubaneswar-751024, Odisha, India
soham.roy2538@gmail.com

Satyabrata Roy
Manipal University Jaipur,
Rajasthan 303007, India
satyabrata.roy@jaipur.manipal.edu

Abstract

In modern days, face recognition is a critical aspect of security and surveillance. Face recognition techniques are widely used for mobile devices and public surveillance. Occlusion is a challenge while designing face recognition applications. In the COVID19 pandemic, we are advised to wear a face mask in public places. It helps us prevent the droplets from entering our body from a potential COVID19 positive person's nose or mouth. However, it brings difficulty for the security personnel to identify the human face by seeing the partially exposed face. Most of the existing models are built based on the entire human face. It could either fail or perform poorly in the scenario as mentioned above. In this paper, a solution has been proposed by leveraging Siamese neural network for human face recognition from the partial human face. The prototype has been developed on the celebrity faces and validated with the state-of-the-art VGGFace2 (Resnet50) model. Our proposed model has performed well and provides very competitive results of 93% and 84.80 ± 4.71% best-of-five and mean accuracy for partial face-images, respectively.

***K*eywords** Deep learning · Face recognition · One-shot learning · Siamese network · VGGface2

1 Introduction

Since late 2019, the World has witnessed a new virus called COVID-19. It started in Wuhan city of China and spread to the rest of the World. Not only does it claim millions of human lives, but also the consequences are catastrophic economically. It brings the entire world to a standstill and disrupting every regular human activity. One way to prevent contact; is the safety precaution and preventive measures given by World Health Organization (WHO). Among these safety precautions, one of the most effective measures is to wear a face mask to prevent the virus from spreading. According to a WHO statement, it says "Masks should be used as part of a comprehensive strategy of measures to suppress transmission and save lives" [1–3]. The face is the most important means of identifying a person in various public places such as ATMs, railway stations, airports, etc. It becomes challenging to recognize a person wearing a mask. Most of the face recognition models have been developed based on features extracted from the whole human face. Due to this, many of such algorithms perform poorly with half faces. It is imperative to effectively improve the existing face recognition approaches to learn and distinguish from the in-exposed face. The learning needs to be done only from the upper half of the face, visible without the mask.

[*] Corresponding author.

A Siamese neural network-based face recognition model from the half-face images has been introduced to overcome the issue. The selection of a Siamese network over a typical Convolutional network for face recognition has been made because of its low computational requirement for training.

The paper has been organized into a total of six sections. Section 2 discusses the related research works. The background concepts have been explained in Section 3. The proposed model has been described in Section 4. It is followed by Section 5 which contains details of the datasets, experimental set-up, and results. Finally, the paper has been concluded in Section 6.

2 Related Work

A reasonable number of research articles have been available in recent years on the domain of face recognition. The primary focus is to apply learning generalized face recognition to work across domains. In the paper by Guo et al. [4], they use a 28-layer ResNet as the backbone, but with a channel-number multiplier of 0.5. This model has been implemented on a pre-trained model and trained on the Full face dataset. Accuracy of over 99% has been achieved. In the paper by Ling et al. [5], they have introduced an attention-based mechanism that uses standard convolutional neural networks (CNNs), such as ResNet-50, ResNet-101. Their work has enabled more discriminative power for deep face recognition.

In the paper by Song et al. [6], they propose to discard feature elements that occlusions have corrupted. They identify face occlusions using PDSN(Pairwise difference Siamese Network) and created an algorithm to recognize the face. However, the occlusions are randomly spread out all over the face. It is not necessarily restricted to the face portion from the nose. They used the Facescrub dataset with training data of 0.5 million images and gained an accuracy of 99.20%.

In the paper by Guo and Zhang [7], they study face recognition of imbalanced train image classes under different lightning, age pose, and variations. They choose a 34 layer standard residual network (ResNet-34) as their feature extractor. They introduced a concept called Classification vector-centered Cosine Similarity to train a better face feature extractor. A concept is introduced called underrepresented-classes promotion, which effectively addresses the data imbalance problem in one-shot learning. The recognition coverage rate has been increased from 25.65% to 94.89% at the precision of 99% for one-shot classes, while still keep an overall accuracy of 99.8% for regular classes.

Most of the work in this domain has been done using deep learning. These models are trained on multitudes of layers and millions of parameters. Many of them are already using pre-trained models and improving upon them. Needless to say that they are very hardware intensive and time-consuming. Although most of them have used full faces to extract facial features, few studies have been done with occlusions, a wide array of lightning, ambiance, and age to improve the model's applicability.

Currently, in the trying times as everyone is wearing a mask in public places, it is almost difficult for the existing models to recognize the face effectively. The problem in hand requires the usage of the face where the portion below the nose is in-exposed while taking into account only the top half of the face. Therefore, the new approach to dealing with the problem requires focusing on the face region, precisely above the tip of the nose. To the best of our knowledge, such an approach is not available in the literature. Here, a Siamese neural network-based face recognition model has been proposed to address the issue.

3 Background concepts

"Siamese network" is a type of neural network architecture that has been first introduced by Bromley et al. [8] for signature verification work purposes. They have trained two similar *Siamese* neural networks having shared the same parameters and weights, which gave an output of two feature vectors when two input signatures are fed to the network. The outcome of two signatures is two vectors. These vectors are compared based on some distance measure that has been used as a loss function during learning. Later, Siamese network has been incorporated into other aspects in computer vision tasks including face verification [9], one-shot image recognition [10]. The crux of the Siamese neural network is to learn general feature representations with a distance (or similarity) metric calculated from the feature vectors from two similar inputs (images in our case). Siamese neural network is a class of network architectures that usually contains two identical networks. The two networks have the same number of layers and configurations with the same parameters and shared weights. The parameter updating in one network is reflected across the other network since the configuration is shared. This framework has been successfully used for dimensionality reduction in weakly supervised metric learning and face verification in [11]. The top layer of these networks consists of a loss function, which computes the similarity or dissimilarity score using the Euclidean distance or the cosine similarity between the feature vector representation on both networks. Two such popular loss functions used with the Siamese network are the contrastive loss [12] and the Triplet Loss [11]. In our work, we have used the Contrastive Loss function (see Eq. 1), which is defined as follows:

$$L(i_1, i_2, Label) = \alpha \times (1 - Label) D_w^2 + \beta \times Label \times max(0, m - D_w)^2 \tag{1}$$

Where i_1 and i_2 are two samples (image of the portion of the face above the tip of the nose), we refer to them as images in the rest of the paper for simplicity. The label is a binary value showing whether the two face images belong to the same class or not; α and β are two constants, both are set to 1 in our study, and m is the margin equal to 2. In Eq. 2, f is a function that maps an image to a vector feature space, i indicates an image, l denotes the index of an image (from image-bank), and w is the learnt weight of the underlying network.

$$\vec{x}_l = f(i_l, w) \tag{2}$$

Eq. 3, is the pair-wise Euclidean distance computed from the feature vector representation from the two images through the network. Here, n is the number of embedding vector lengths (256), the model output.

$$D_w = \left(\sum_{j=1}^{n} |x_{1j} - x_{2j}|^p \right)^{\frac{1}{p}} \tag{3}$$

A Siamese network's objective is to make the feature vector representation of the input images that have the same class label closer and push away the feature vector representation of the input images are different class labels. Because of the Contrastive loss function (Eq. 1), after the learning/training stage of the model, the feature vector output has the characteristic that the Euclidean distance of the images of the same class is closer to the images of different classes. To decide whether two images belong to a similar class (label as 0) or a different class (label as 1), we need to determine a threshold value on the cosine dissimilarity of the distance between the embedded vector representations. This step is typically determined by training the network and studying similarity scores from genuine-genuine and genuine-fake images. A top K match is considered as qualification criteria from the image bank based on the threshold value.

VGGFace2 model [13] developed by researchers at the Visual Geometry Group at Oxford. The dataset has been prepared before modeling by the VGG team itself. Models are trained on the dataset, specifically a ResNet-50 and a SqueezeNet-ResNet-50 model (SE-ResNet-50 or SENet). Given an image as an input, the model gives the output of a 2048 vector embedding. The vector's length is normalized using the $L2$ vector norm (Euclidean distance from the origin). They obtain a vector for each image called a face descriptor. The way they get the face descriptor is that the extended bounding box of the face is resized to make the shorter side of the image 256 pixels; then the center 224×224 dimension crop of the face image is used as input to the network. The face descriptor is extracted from the layer adjacent to the classifier layer, the last layer of the model. It leads to a 2048 length vector representation of the image, which is then L2 normalized. The similarity between the two images is calculated by the cosine similarity function from the two face descriptors. The deep models (ResNet-50 and SENet) trained on VGGFace2 achieve state-of-the-art performance on the IJB-A, IJB-B, and IJB-C benchmarks.

4 Proposed Model

A research workflow diagram has been given in Fig. 1. It shows that the used pipeline takes an image or video as input and detects faces using Multi-task Cascaded Convolutional Neural Network (MTCNN) [14] algorithm. Subsequently, input image pre-processing is performed, such as gray scaling, face alignment, etc. Then, the proposed model has been used to match partial face image input with the existing face image-bank. Cosine distance metric has been employed to find out the face similarity.

In our proposed model, a Siamese neural network (SNN) is to be prepared for lightweight face image recognition. The model has been used to get the feature representation of an image when it is passed through the network. The model has two sub-networks, as with a Siamese network having shared weights and biases. An extra padding layer is required all along the layers to keep the image's dimension constant. The model has six layers in total. The first layer is a convolution $2d$ layer with an input channel of 1 as the image is gray-scale and the output has 4 channels. We use ReLu (Rectified Linear Unit) activation layer and batch normalization for the four output channels. We are using a kernel size of 3 and padding up with 1 to keep the image dimension the same. The second layer of the image

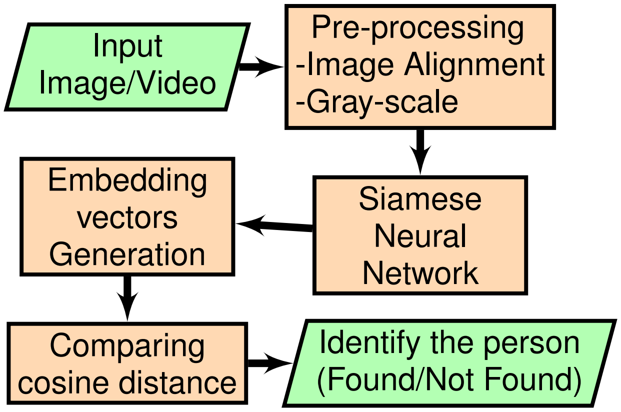

Input Image/Video
Pre-processing -Image Alignment -Gray-scale
Siamese Neural Network
Embedding vectors Generation
Comparing cosine distance
Identify the person (Found/Not Found)

Figure 1: Algorithm Workflow of this paper

is almost similar to the first one. In this layer, the only difference is that the input is of four channels and the output is eight channels, and the batch normalization is happening across all eight channels. The third layer is also familiar; here, the number of input channels is eight, and no output channel is also eight, followed by batch normalization. At the end of the third layer, we connect it to a fully connected sequential layer with the input dimension of $8 \times 100 \times 100$ and 512 nodes. In the fully connected layers also we are using the ReLu activation function. The second layer in the Linear space or the fifth layer in the model is a linear layer with input and output dimensions as 512 and 512. We are using ReLu activation here. Our model's final layer is linear with a 512 input dimension and a 256 output dimension. This 256 output dimension is our model's final output, which is the feature embedding of an input image when passed through the network. The Contrastive loss function is the objective function to learn the similar and dissimilar classes of images. A block diagram of the proposed Siamese network architecture is given in Fig. 2.

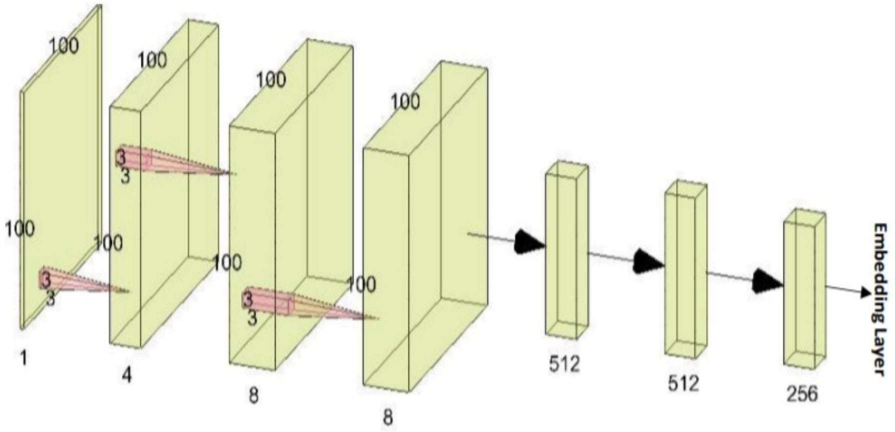


Figure 2: Proposed Siamese Neural Network Architecture

After the learning/training stage of the model, the last layer output is a vector of length 256. It has the characteristic that the Euclidean distance of the same class's images is closer to the images of different classes. Two images belong to a similar class (label as 0) or a different class (label as 1), which is decided on a threshold (Th.=0.1) value. It is calculated on the cosine distance of the dissimilarity between the embedded vector representations. This step is typically determined by a rigorous study of the similarity scores from the genuine-genuine and genuine-fake images. After this step, we can determine a top K match for an anchor image(cropped masked face) in an image bank (consisting of the individuals' full faces). A bird's-eye view of the implementation has been given in Fig. 3.

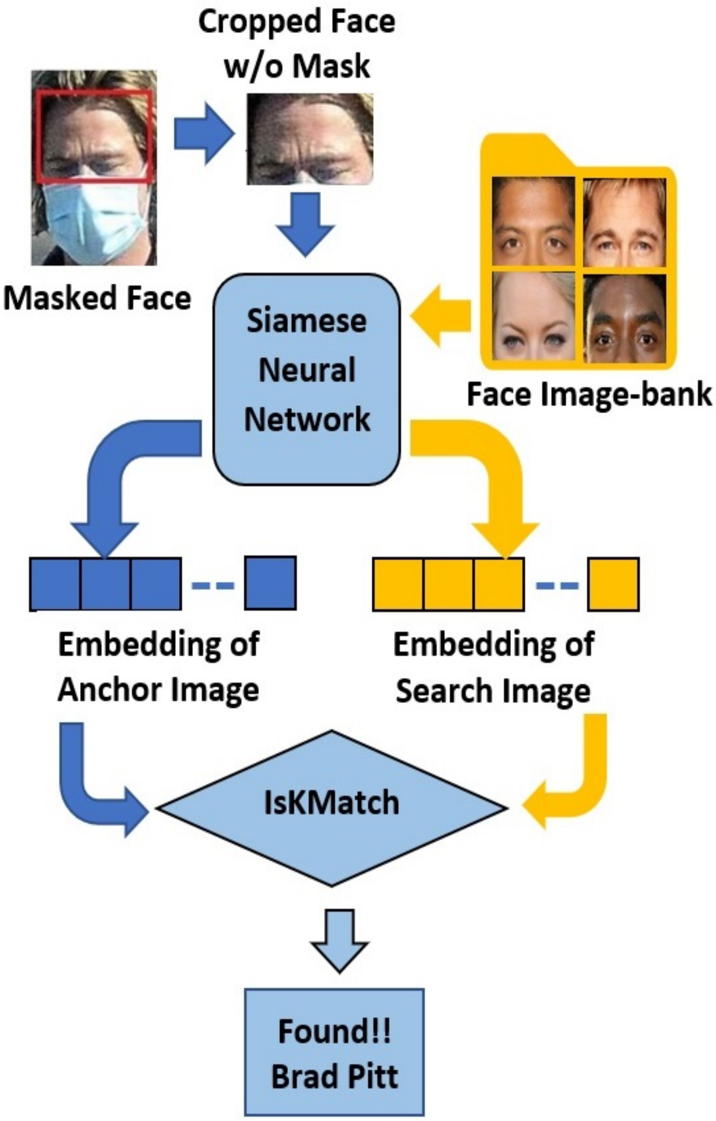


Figure 3: Bird's-eye view of the proposed Siamese Neural Network pipeline

5 Experimental Results and Analysis

5.1 Datasets

The dataset[2] contains 20 celebrity faces (*Blake Lively, Brad Pitt, Brendan Gleeson, Brian Cox, Brie Larson, Britney Spears, Brittany Snow, Bruce Lee, Bruno Mars, Cameron Diaz, Cate Blanchett, Chadwick Boseman, Chris Hemsworth, Chris Martin, Christian Bale, Emma Roberts, Emma Stone, Emma Watson, Eric Bana, Ethan Hawke*), with the images in each class varying in age, luminosity, brightness, expression and face alignment, and visual quality. The purpose of this is to generalize the data as much as possible for each class. Each folder contains around 100 images for a celebrity. Since a mask can be worn in varying degrees, it can be worn too high or low. The used dataset has been prepared in a way to resemble the realistic situation. A test dataset is also created from the same source data containing only 30 images per person.

[2] Dataset:https://github.com/prateekmehta59/Celebrity-Face-Recognition-Dataset/blob/master/README.md

5.2 System Configuration

The paper is implemented using Python 3.7 and PyTorch (GPU) 1.5.1 on an Intel(R) Core(TM) $i7-9750H$ CPU (9^{th} Gen.) 2.60GHz, 16GB RAM and 6GB NVIDIA GeForce RTX 2060 with 64 bits Windows 10 Home operating system. Again, the same codes have been validated on the Online Google Colab environment with more learning parameters. For implementing the VGGFace2 model on the same datasets for comparison, the Tensorflow-GPU 1.14 has been used simultaneously. The dimension of the output embedding is 256.

Three types of optimization techniques are used to explore the best possible model. These are AdaBelief (lr=$1e-3$, eps=$1e-16$ and betas=$(0.9, 0.999)$), Adam (lr=$1e-3$, weight_decay=0.0005) and Stochastic Gradient Descent (SGD) (lr=$1e-3$, momentum=0.9) [15, 16].

5.3 Results

The proposed model has been implemented with different configurations. The first experiment (test-case-I): the configurations vary based on embedding vector length, threshold, optimization algorithms, and the number of learning epochs, etc. Three distinct types of optimization techniques have been employed to observe their effects. It is noticed that AdaBelief outperforms others in early epochs in terms of the total loss. However, SGD performs better than Adam and AdaBelief algorithms in longer iterations. The obtained results are given in Tables 1 and 2.

Table 1: Results obtained from different optimization algorithms and half-face images using the proposed model where the threshold (Th.)=0.1 and the embedding vector length=64

Embedding Length	**Optim**	**Epoch**	**Total Loss**	**Time (min.)**
64	AdaBelief	100	0.0567	1.7320
	Adam		0.1595	1.7560
	SGD		0.0751	1.6506
	AdaBelief	500	0.0035	9.1326
	Adam		0.0493	8.9267
	SGD		0.0098	8.5420

Table 2: Results obtained from different optimization algorithms and half-face images using the proposed model where the threshold (Th.)=0.1 and the embedding vector length=256

Embedding Length	**Optim**	**Epoch**	**Total Loss**	**Time (min.)**
256	AdaBelief	100	0.2110	1.5600
	Adam		0.3450	1.6500
	SGD		0.2025	1.6300
	AdaBelief	500	0.0219	9.7975
	Adam		0.0508	9.4183
	SGD		0.0170	8.4834

The best model from our experiments has been used on a randomly generated face test-image-bank (containing 20 persons × 35 half face-images) for evaluating the model accuracy (%). The second experiment (test-case-II): The anchor image is a half-face, and the test-image-bank contains half faces. It is very satisfactory that the proposed model built from scratch provides very efficient results in our experiments. The experiment has been executed 5 times independently to examine its robustness. The empirical data has been given in Table 3. The best of five (#Bo5) accuracy, the mean accuracy, and the standard deviation are 93%, 84.80%, and ±4.71, respectively.

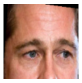
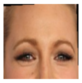
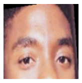
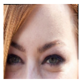
In the third experiment (test-case-III): four images (*68bradpitt.png, 4blakelively.png, 22chadwikboseman.png, and 45emmastone.png*) have been randomly picked from the test set, and a comparison between the proposed best model and the VGGFace2 have been made. The four images are of celebrities Brad Pitt, Blake Lively, Chadwick Boseman, and Emma Stone. A K-shot learning [17] approach has been used to recognize the actual person from a partial face image (anchor). Here, K$\geq$ 10 is used. It suggests that the similarity score between 10 or more images and the anchor image is always less than or equals 0.1 (this threshold is achieved through an elbow technique). In literature, it is instructed to use 0.5 as the threshold value for VGGFace2 (if the similarity score falls below 0.5, it is considered a match). Even for the VGGFace2 implementation, K$\geq$ 10 matches have been used to qualify a match.

Table 3: Results obtained from the proposed model for 5 independent runs with the threshold (Th.)=0.1 and embedding vector length=256

#Run	Accuracy(%)	Time (Sec.)	Mean Acc.	Std.
1	82	3.1524	84.80	±04.71
2	79	2.9113		
3	**93**	3.0011		
4	84	2.8905		
5	86	3.2021		

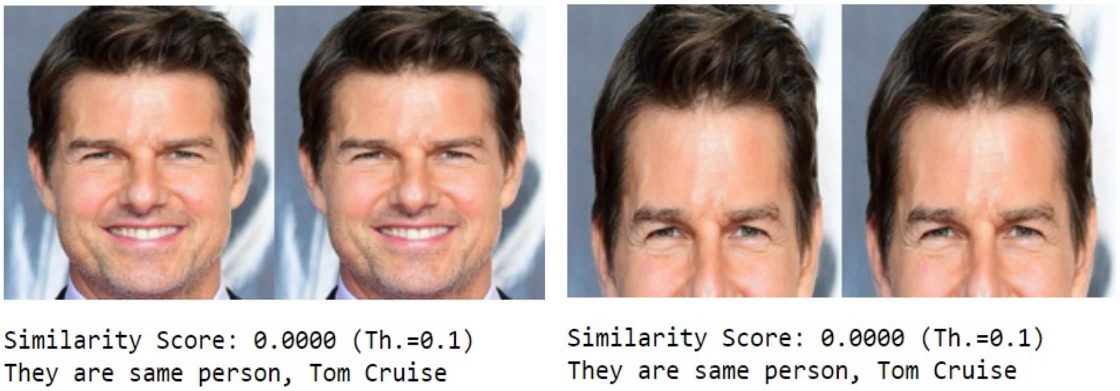
In Table 4, the performance of our proposed Siamese neural network-based model outperforms the state-of-the-art VGGFace2 in terms of average time taken for searching and accuracy. It must be noted that the complete image bank has 20 different personalities, and the test-case-III has been implemented for the whole test dataset without the first match-only match (FMOM) policy.

Table 4: Results obtained from different celebrity half-face images using ours and VGGFace2 models, respectively. K=10 indicates the dissimilarity values between anchor image and at least 10 images fall below the threshold (Th.)

Celebs (Anchor Image)	Models	Th.	Total Time	Avg. Time	Searched Images	Top Searched Names (K=10)	Scores
	**Ours**	0.1	0.0515	0.000098	531	Brad Pitt	0.0015
						Brendan	0.7730
						Chris martin	0.0136
	VGGFace2	0.5	0.5585	0.000112	549	Brad Pitt	0.2081
						Chris Hemsworth	0.4953
	**Ours**	0.1	0.4991	0.000092	541	Blake Lively	0.0011
	VGGFace2	0.5	0.5651	0.000128	442	Blake Lively	0.3154
						Cate Blanchett	0.3963
						Brad Pitt	0.4197
						Emma Stone	0.4252
						Brittany Snow	0.4263
	**Ours**	0.1	0.5286	0.000096	551	Chadwik Boseman	0.0122
	VGGFace2	0.5	0.5494	0.000104	529	Chadwik Boseman	0.2577
						Bruno Mars	0.4347
	**Ours**	0.1	0.5183	0.000094	553	Emma Stone	0.0842
	VGGFace2	0.5	0.5589	0.000124	452	Emma Stone	0.2769
						Cate Blanchett	0.3639
						Blake Lively	0.4284
						Chris Hemsworth	0.4350
						Brittany Snow	0.4476

All test cases (I, II, and III) are tested on the known faces (suggests model built using the same persons' images). Here, the model has been tested on few unknown faces (not used in learning). Two prominent examples are shown in Figs. 4, 5 and 6. In Figs. 4 and 5, comparisons have been given for Tom Cruise and Keanu Reeves based on both the full and half face similarity. Similarly, a dissimilarity comparison has also been shown in Fig. 6. Both Tom Cruise and Keanu Reeves face images have not been used while the model building; even then, the obtained results are accurate using the same embedding scheme and adopted threshold.

The first adaption of the model, which we have built from scratch, gives encouraging results compared with the state-of-the-art. However, as the model is constructed from a moderate-sized dataset and constrained resources (hardware), we believe that if the architecture is tested in a broader dataset run on a high-performing machine, the robustness of the model would undoubtedly improve.



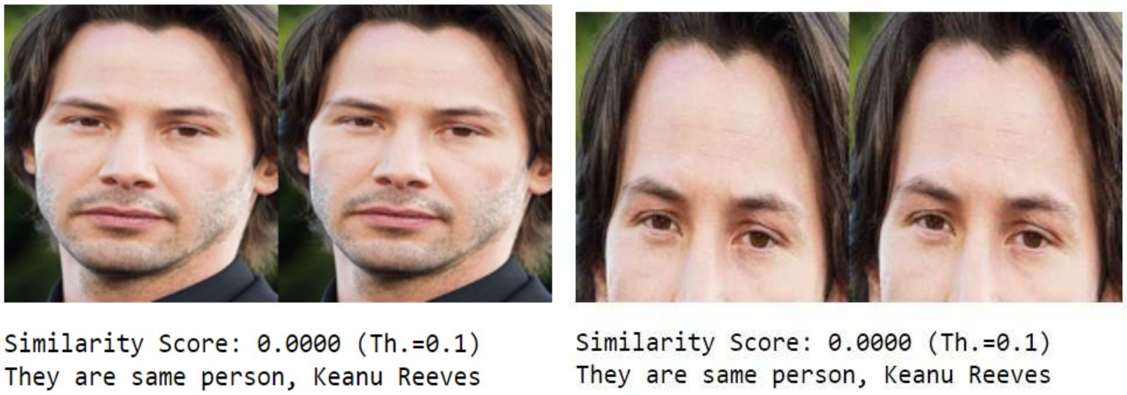
```
Similarity Score: 0.0000 (Th.=0.1)
They are same person, Tom Cruise
```

(a) Proposed model on unkonwn full face

```
Similarity Score: 0.0000 (Th.=0.1)
They are same person, Tom Cruise
```

(b) Proposed model on unkonwn half face

Figure 4: Example of Tom Cruise face similarity using the proposed model, where the embedding vector length=256



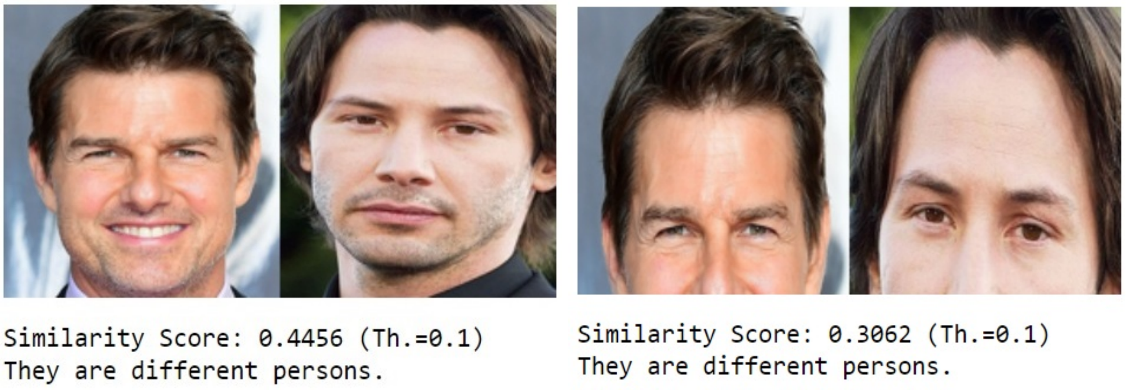
```
Similarity Score: 0.0000 (Th.=0.1)
They are same person, Keanu Reeves
```

(a) Proposed model on unkonwn full face

```
Similarity Score: 0.0000 (Th.=0.1)
They are same person, Keanu Reeves
```

(b) Proposed model on unkonwn half face

Figure 5: Example of Keanu Reeves face similarity using the proposed model, where the embedding vector length=256



```
Similarity Score: 0.4456 (Th.=0.1)
They are different persons.
```

(a) Proposed model on unkonwn full face

```
Similarity Score: 0.3062 (Th.=0.1)
They are different persons.
```

(b) Proposed model on unkonwn half face

Figure 6: Example of face dissimilarity using the proposed model, , where the embedding vector length=256

6 Conclusion

Deep learning-based face detection and recognition are typical computer vision applications. However, there is still room for improvement in human face recognition. It is already established that CNN-based deep learning requires many input images and takes a more prolonged training time. On the other hand, a Siamese neural network can work with fewer input data and provides similarity or dissimilarity scores between any two faces. The proposed SNN model developed based on half of the human face gives very competitive results in identifying a person from only the exposed upper portion of the face (mainly above the tip of the nose or nostrils). The proposed model takes less search-time/per image than VGGFace2 while searching a face. The proposed SNN model gives 93% #Bo5 accuracy and $84.80 \pm 4.71\%$ mean accuracy for the partial face matching.

In the future, the model can be extended to train based on a larger dataset and examined on a real-time video.

Compliance with ethical standards

Financial support and funding Nil.

Conflict of interest The authors declare that they have no conflict of interest.

Informed Consent Informed consent was obtained from all individual participants included in the study.

Ethical Approval All procedures performed in studies involving human participants were in accordance with the ethical standards of the institutional and/or national research committee and with the 1964 Helsinki declaration and its later amendments or comparable ethical standards.

References

[1] Steffen E Eikenberry, Marina Mancuso, Enahoro Iboi, Tin Phan, Keenan Eikenberry, Yang Kuang, Eric Kostelich, and Abba B Gumel. To mask or not to mask: Modeling the potential for face mask use by the general public to curtail the covid-19 pandemic. *Infectious Disease Modelling*, 5:293–308, 2020.

[2] Tom Li, Yan Liu, Man Li, Xiaoning Qian, and Susie Y Dai. Mask or no mask for covid-19: A public health and market study. *PloS one*, 15(8):e0237691, 2020.

[3] Mohamed Loey, Gunasekaran Manogaran, Mohamed Hamed N Taha, and Nour Eldeen M Khalifa. A hybrid deep transfer learning model with machine learning methods for face mask detection in the era of the covid-19 pandemic. *Measurement*, 167:108288, 2021.

[4] Jianzhu Guo, Xiangyu Zhu, Chenxu Zhao, Dong Cao, Zhen Lei, and Stan Z Li. Learning meta face recognition in unseen domains. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6163–6172, 2020.

[5] Hefei Ling, Jiyang Wu, Junrui Huang, Jiazhong Chen, and Ping Li. Attention-based convolutional neural network for deep face recognition. *Multimedia Tools and Applications*, 79(9):5595–5616, 2020.

[6] Lingxue Song, Dihong Gong, Zhifeng Li, Changsong Liu, and Wei Liu. Occlusion robust face recognition based on mask learning with pairwise differential siamese network. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 773–782, 2019.

[7] Yandong Guo and Lei Zhang. One-shot face recognition by promoting underrepresented classes. *arXiv preprint arXiv:1707.05574*, 2017.

[8] Jane Bromley, Isabelle Guyon, Yann LeCun, Eduard Säckinger, and Roopak Shah. Signature verification using a" siamese" time delay neural network. *Advances in neural information processing systems*, pages 737–737, 1994.

[9] Yaniv Taigman, Ming Yang, Marc'Aurelio Ranzato, and Lior Wolf. Deepface: Closing the gap to human-level performance in face verification. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1701–1708, 2014.

[10] Gregory Koch, Richard Zemel, and Ruslan Salakhutdinov. Siamese neural networks for one-shot image recognition. In *ICML deep learning workshop*, volume 2. Lille, 2015.

[11] Florian Schroff, Dmitry Kalenichenko, and James Philbin. Facenet: A unified embedding for face recognition and clustering. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 815–823, 2015.

[12] Sumit Chopra, Raia Hadsell, and Yann LeCun. Learning a similarity metric discriminatively, with application to face verification. In *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, volume 1, pages 539–546. IEEE, 2005.

[13] Qiong Cao, Li Shen, Weidi Xie, Omkar M Parkhi, and Andrew Zisserman. Vggface2: A dataset for recognising faces across pose and age. In *2018 13th IEEE international conference on automatic face & gesture recognition (FG 2018)*, pages 67–74. IEEE, 2018.

[14] Kaipeng Zhang, Zhanpeng Zhang, Zhifeng Li, and Yu Qiao. Joint face detection and alignment using multitask cascaded convolutional networks. *IEEE Signal Processing Letters*, 23(10):1499–1503, 2016.

[15] Juntang Zhuang, Tommy Tang, Sekhar Tatikonda, Nicha Dvornek, Yifan Ding, Xenophon Papademetris, and James S Duncan. Adabelief optimizer: Adapting stepsizes by the belief in observed gradients. *arXiv preprint arXiv:2010.07468*, 2020.

[16] Nikhil Ketkar. Introduction to pytorch. In *Deep learning with python*, pages 195–208. Springer, 2017.

[17] Flood Sung, Yongxin Yang, Li Zhang, Tao Xiang, Philip HS Torr, and Timothy M Hospedales. Learning to compare: Relation network for few-shot learning. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1199–1208, 2018.