Verification of Identity using Triplet Network

Sai Anuhya Kondubhatla
The University of Texas at Dallas
800 W Campbell Rd, Richardson, TX 75080
SaiAnuhya.Kondubhatla@UTDallas.edu

Kirthi Menon
The University of Texas at Dallas
800 W Campbell Rd, Richardson, TX 75080
Kirthi.Menon@UTDallas.edu

Chandan Raju Vysyaraju
The University of Texas at Dallas
800 W Campbell Rd, Richardson, TX 75080
ChandanRaju.Vysyaraju@UTDallas.edu

Harshavardhan Naregudem
The University of Texas at Dallas
800 W Campbell Rd, Richardson, TX 75080
Harshavardhan.Naregudem@UTDallas.edu

Abstract

Given a dataset of biometric images (such as facial images, iris images, or handwritten text images), the goal is to train a deep learning model that can accurately verify the identity of individuals based on their biometric data. The model should be able to take an anchor image, a positive image of the same individual, and a negative image of a different individual, and determine if the anchor and positive images belong to the same individual. The performance of the model should be evaluated using evaluation metrics such as accuracy, precision, recall, F1 score, AUC-ROC, cosine similarity, and triplet loss. The objective is to outline the scope of identity verification using triplet networks, and to provide a clear understanding of the desired outcomes and requirements of the solution.

In the Triplet Network, selecting the right triplets for training is crucial for achieving good performance. Inspired by the idea of curriculum learning, we propose a novel online negative exemplar mining strategy that ensures consistently increasing difficulty of triplets as the network trains. By gradually introducing harder negative examples during training, the network can learn to distinguish between more similar faces and achieve better performance.

1. Introduction

Identity verification is a critical task in various applications, including security systems, financial transactions, and access control. Traditional methods for identity verification, such as passwords and PINs, can be compromised, leading to unauthorized access or fraud. Biometric methods, such as fingerprint or face recognition, are more secure, but they can be fooled by spoofing attacks. Triplet networks are a type of deep learning architecture that can learn to identify similarities and differences between images. They can be used for face recognition, person re-identification, and other tasks that require identifying unique features of an individual. By using a triplet network, we can compare an input image to two other images and determine if they belong to the same person or not.

The framework for identity verification using triplet networks typically involves three components: an anchor image, a positive image, and a negative image. The anchor image represents the person whose identity we want to verify, while the positive image is another image of the same person. The negative image is an image of a different person. During training, the network learns to minimize the distance between the anchor and positive images while maximizing the distance between the anchor and negative images. During testing, we compare the distance between the anchor and the test image to a threshold value to determine if the identity is verified or not.

In summary, the motivation for working on the problem of identity verification using triplet networks is to develop a more secure and reliable method for verifying identity that can be used in various applications. The framework involves training a triplet network to learn unique features of individuals and using these features to compare images and verify identity.

2. Related Work

Several deep learning-based approaches have been proposed for identity verification. These approaches include convolutional neural networks (CNNs), Siamese networks, and triplet networks. Triplet networks have been shown to outperform other methods in various identity verification tasks. Previous research on triplet networks for identity verification has focused on various aspects, including triplet selection strategies, network architectures, and loss functions. Some works have proposed using attention mechanisms to focus on relevant parts of the face image, while others have explored different loss functions such as contrastive loss and margin-based loss. However, few works have explored online negative exemplar mining strategies and hard-positive mining techniques for improving the performance of triplet networks for identity verification.

3. Method

3.1. Triplet Network

The proposed framework consists of three main components: the anchor network, the positive network, and the negative network. The positive and negative networks create embeddings for the input image, while the anchor network creates an embedding for the reference image. Using the triplet loss function, which increases the distance between the anchor and positive embeddings and decreases the distance between the anchor and negative embeddings, the network is trained. A Triplet network is comprised of 3 instances of the same feed-forward network (with shared parameters). When fed with 3 samples, the network outputs 2 intermediate values - the L2 distances between the embedded representation of two of its inputs from the representation of the third. If we will denote the 3 inputs as x, x +, and x - and the embedded representation of the network as Net(x), the one before the last layer will be the vector:

$$TripletNet(x, x^-, x^+) = \begin{bmatrix} \|Net(x) - Net(x^-)\|_2 \\ \|Net(x) - Net(x^+)\|_2 \end{bmatrix} \in \mathbb{R}^2_+$$

The Triplet Network is trained using a loss function called the Triplet Loss Function. This loss function is used to learn a distance metric between faces in an embedding space. The embedding space is a high-dimensional space where each point represents a face. The Triplet Loss Function minimizes the distance between an anchor image and a positive image (an image of the same person), while maximizing the distance between the anchor image and a negative image (an image of a different person).During the training phase, the Triplet Network takes in three images as input: an anchor image, a positive image, and a negative image. The anchor image and positive image are images of the same person, while the negative image is an image of a different person. The Triplet Network then computes the distance between the anchor image and positive image and the distance between the anchor image and negative image. The Triplet Loss Function then computes the loss based on these distances and updates the weights of the network to minimize the loss.

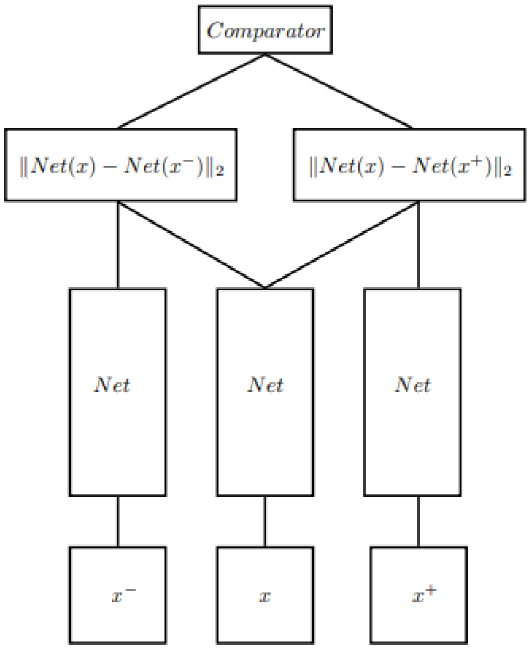
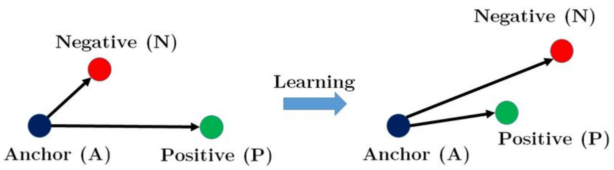


Figure 1: Triplet network structure

3.2. Triplet Loss

The embedding is represented by $f(x) \in R^d$. An image is inserted into a two-dimensional Euclidean space. We further confine this bedding to exist on the d-dimensional hypersphere, with $||f(x)||_2 = 1$. In the context of nearest-neighbor classification, this loss is motivated. Here, we want to make sure that an image of a particular person or its "anchor" is closer to all other photographs of that person than it is to any other image, whether positive or negative.



There would be a lot of triplets that could easily satisfy the requirement in Equation (1) if all feasible triplets were produced. These triplets would still be sent through the network but would not contribute to the training, resulting in slower convergence. Selecting hard triplets who are active and can hence help improve the model is key. Thus we want,

$$||x_i^a - x_i^p||_2^2 + \alpha < ||x_i^a - x_i^n||_2^2, \forall (x_i^a, x_i^p, x_i^n) \in \tau (1)$$

where α is a margin that is enforced between positive and negative pairs. τ is the set of all possible triplets in the training set and has cardinality N. The loss that is being minimized is then L =

$$\sum_i^N [||f(x_i^a) - f(x_i^p)||_2^2 - ||f(x_i^a) - f(x_i^n)||_2^2 + \alpha]_+ (2)$$

3.3. Triplet Selection

It refers to the process of selecting triplets of images from a large dataset that are suitable for training the Triplet Network.

The goal of triplet selection is to select triplets that are challenging for the Triplet Network to learn from. If the triplets are too easy, the Triplet Network may not learn the features necessary for accurate face recognition. On the other hand, if the triplets are too difficult, the Triplet Network may not be able to learn at all. The process of triplet selection involves selecting an anchor image, a positive image, and a negative image for each triplet. The anchor image and positive image are images of the same person, while the negative image is an image of a different person.

To select triplets, several strategies can be used. One common strategy is to use all possible combinations of images in the dataset. However, this can be computationally expensive and may result in many easy or redundant triplets. Another strategy is to use online triplet mining, where triplets are dynamically selected during training based on their loss value. This approach can be more efficient and can result in more challenging triplets.

To select triplets using online triplet mining, the Triplet Network is first trained on a set of randomly selected triplets. After each iteration of training, the loss value of each triplet is calculated. Triplets with the highest loss values are selected as the most challenging triplets and are used for the next iteration of training.

In summary, triplet selection is the process of selecting triplets of images that are challenging for the Triplet Network to learn from. This process is critical for training an accurate face recognition system using the Triplet Network architecture.

4. Experiments

In this project we have experimented using triplet network which involves training a deep neural network, specifically a triplet network, to verify the identity of an individual using their facial features.

4.1. Datasets

For this project, we used the Labeled Faces in the Wild (LFW) dataset which consists of 13,233 images of 5,749 people. Each image is labeled with the name of the person in the image. We divided the dataset into 80% training set and 20% validation set.

Labeled Faces in the Wild (LFW) is the de-facto academic test set for face verification. We follow the standard protocol for unrestricted, labeled outside data and report the mean classification accuracy as well as the standard error of the mean.

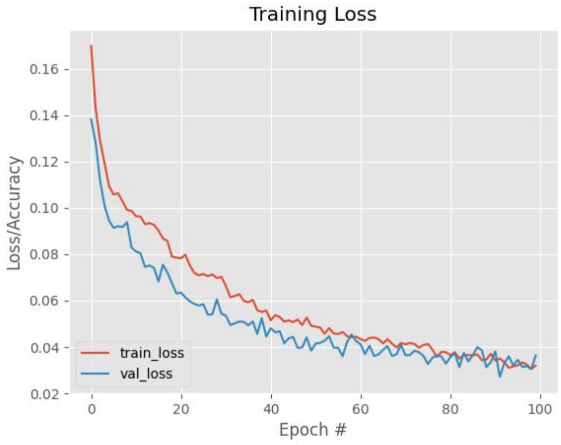
4.2. Evaluation metrics

Accuracy is used to measure the overall correctness of the model which is the percentage of image pairs that are correctly classified as the same or different person. Precision is Used to measure the number of true positives divided by the number of true positives and false positives. The Recall is used to measure the number of true positives divided by the number of true positives plus false negatives. F1 Score between Precision and Recall is calculated as $2 * (precision * recall)/(precision + recall)$

4.3. Training Triplet Network

We are training a triplet network using the LFW dataset to verify the identity of a person given an image. The network will be trained for 100 epochs using the Adam optimizer with a learning rate of 0.001. We are using the triplet loss function to train the network.

In the Initial phase, we took an image of the person from the input dataset whose identity needs to be verified. This will be the anchor image and is used to generate a fixed-dimensional embedding. For the next step, we selected another image of the same person. It will be taken as the positive image and is used to encourage the embeddings of the anchor and positive image to be close together. Now, we have selected an image of a different person which acts as a negative image. This is used to encourage the embeddings of the anchor image and negative image to be far apart. Use the triplet function to generate embeddings for each of the three images. The triplet function takes an input image and returns a fixed-dimensional embedding vector. The distances between the anchor image and positive image embeddings and also the distance between the anchor and negative image embeddings are calculated. Now to verify the identity the distance between the anchor and positive embeddings should be smaller than the distance between the anchor and negative embeddings by a pre-determined margin.



The hyper parameters such as batch_size, epochs, learning_rate, and steps_per_epoch, optimizer are tuned to obtained optimal results from the model. After series of iterations and tuning these hyper parameters the difference between the training loss and value loss is decreased.

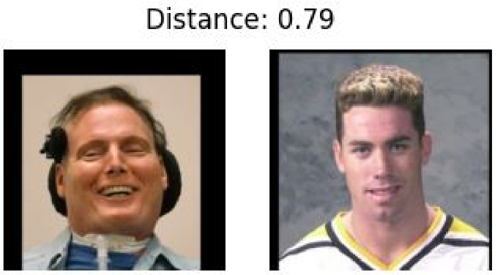
The graph displayed above shows that the values for training loss and value loss are approximately same, indicating that the predicted result is very close to the actual result. This means that the trained model is more reliable and accurate in predicting outcomes, making it more useful and effective for real world applications. Overall, by carefully tuning hyper parameters and reducing the difference between training and value loss, we can build a model that is more robust and accurate in its predictions.

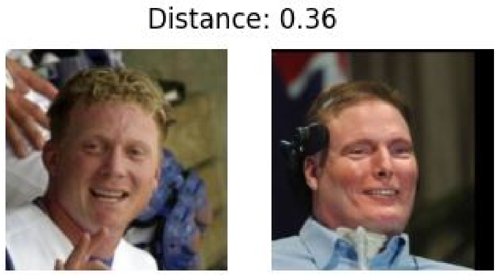
4.4. Testing the trained network

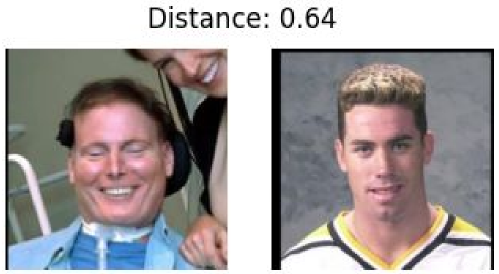
After training the model, it is important to test and validate its performance using a separate set of data that was not used during the training. This helps us to ensure that the model can generalize well to new, unseen data. Here, a set of input images are provided to the model and compared its output to the correct labels.

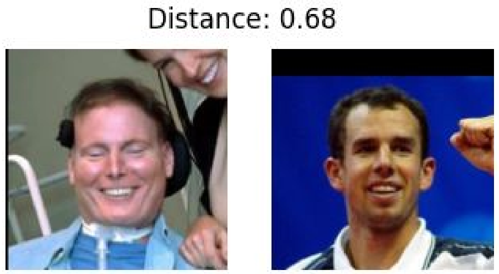
The test function takes three inputs that are anchor, positive and negative images and compares them using the euclidean distance. The goal of the function is to compare the three images and determine whether the images belong to the same individual. To do this, the function increases the distance between the anchor and positive images, and decreases distance between the anchor and negative images. By comparing the distance between the anchor and positive image to the distance between the anchor and negative images, the function can determine whether the model is correctly recognizing the similarity between the two images of the same person.

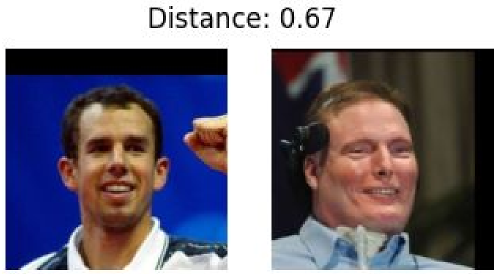
By executing the test function, we can observe in the output that produces the distance between the images that are compared as shown below.

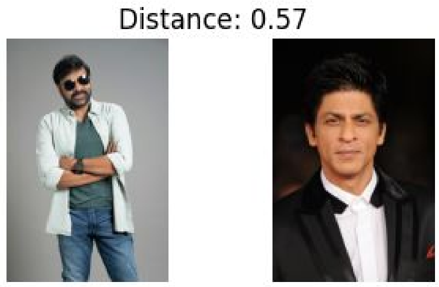




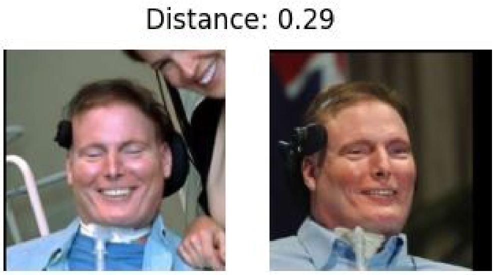








In the below output we can observe that if the images are different but belong to the same person then the distance between the two images is very less which is the value of threshold alpha.



If the images are same then we could observe that the distance is 0 as shown below.

5. Conclusion

The triplet network architecture demonstrated promising results in accurately verifying the identities. Moreover in this project, we explored various strategies for training the triplet network, including optimization techniques, loss functions, and hyperparameter tuning. These experiments revealed that fine-tuning the network and carefully selecting the suitable parameters significantly contributed to improving the overall performance and convergence speed of the model. The triplet network demonstrated a certain level of resilience to these challenges, indicating its potential applicability in real-world scenarios where such variations are encountered. The Project's findings indicate that the triplet network approach holds great potential for identity verification. By utilizing its ability to learn discriminative embeddings and similarity metrics, the network can effectively distinguish genuine identities.

6. References

[1] Bradford, R.R., Bradford, R.”Introduction to Handwriting Examination and Identification,” Nelson-Hall(1992)

[2] L. Franck and R. Plamondon, ”Automatic Signature Verification: The State of the Art. 1994”.

[3] D. Chen, S. Ren, Y. Wei, X. Cao, and J. Sun. Joint cascade face detection and alignment. In Proc. ECCV, 2014

[4] Y. Taigman, M. Yang, M. Ranzato, and L. Wolf. Deepface: Closing the gap to human-level performance in face verification. In IEEE Conf. on CVPR, 2014

[5] M. Lin, Q. Chen, and S. Yan. Network in network. CoRR, abs/1312.4400, 2013

[6] Florian Schroff, Dmitry Kalenichenko, James Philbin, ”FaceNet: A Unified Embedding for Face Recognition and Clustering,” In IEEE Conf. on CVPR, 2015

[7] D. R. Wilson and T. R. Martinez. The general inefficiency of batch training for gradient descent learning. Neural Networks, 16(10):1429–1451, 2003

[8] W. Min, S. Mei, Z. Li and S. Jiang, ”A Two-Stage Triplet Network Training Framework for Image Retrieval,” in IEEE Transactions on Multimedia, vol. 22, no. 12, pp. 3128-3138, Dec. 2020, doi: 10.1109/TMM.2020.2974326.

[9] A. Challa, S. Danda, B. S. D. Sagar and L. Najman, ”Triplet-Watershed for Hyperspectral Image Classification,” in IEEE Transactions on Geoscience and Remote Sensing, vol. 60, pp. 1-14, 2022, Art no. 5515014, doi: 10.1109/TGRS.2021.3113721.