Siamese Networks

What are Siamese Networks

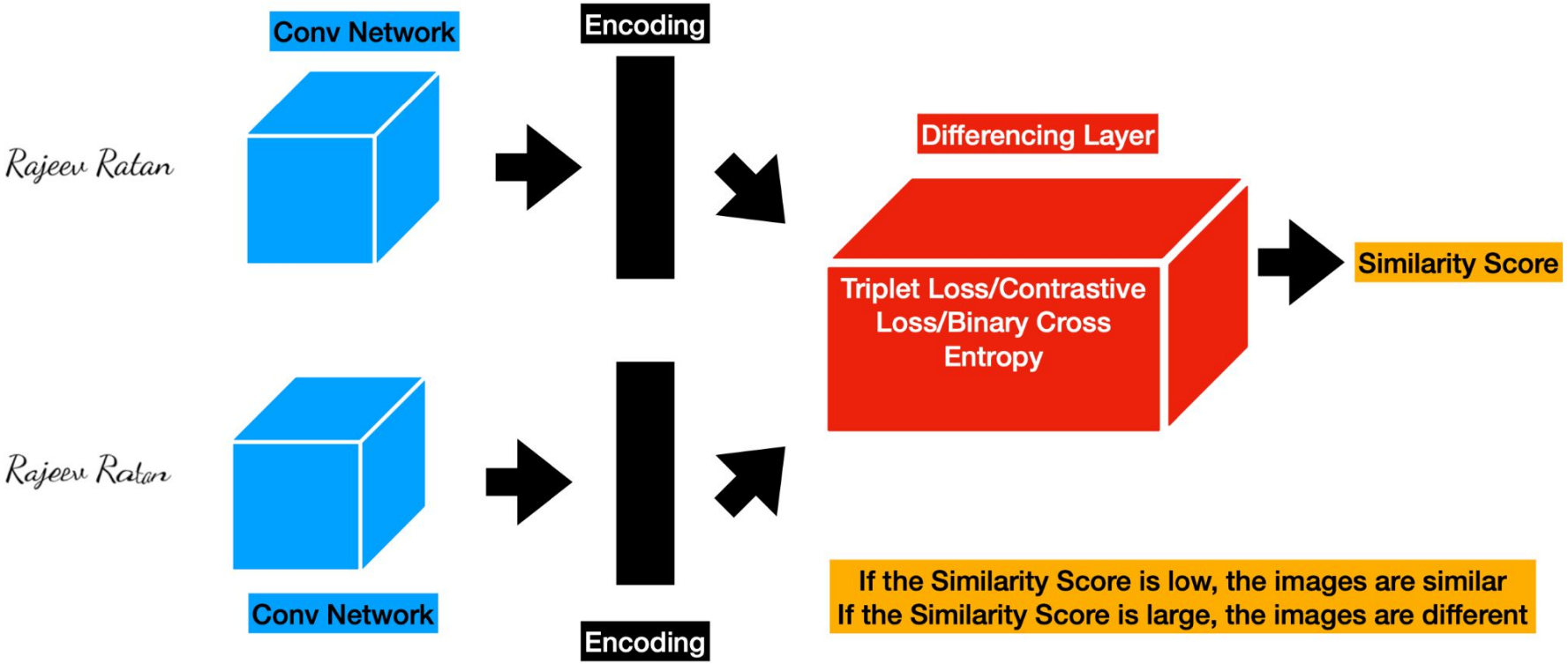
- The term comes from Siamese twins (conjoined twins) which are twins that unfortunately joined together at birth, sometimes sharing organs.
- Siamese Networks are similarly ‘connected’. They consist of:
  - Two or more identical (in architecture) subnetworks (Neural Networks)
  - The subnetworks share the same parameters & weights
  - Parameter updates are mirrored on both networks (i.e. when one updates, the other one updates as well)

https://commons.wikimedia.org/wiki/File:Chang_and_Eng_the_Siamese_twins,_in_a_games_room._Coloured_e_Wellcome_V0007366.jpg

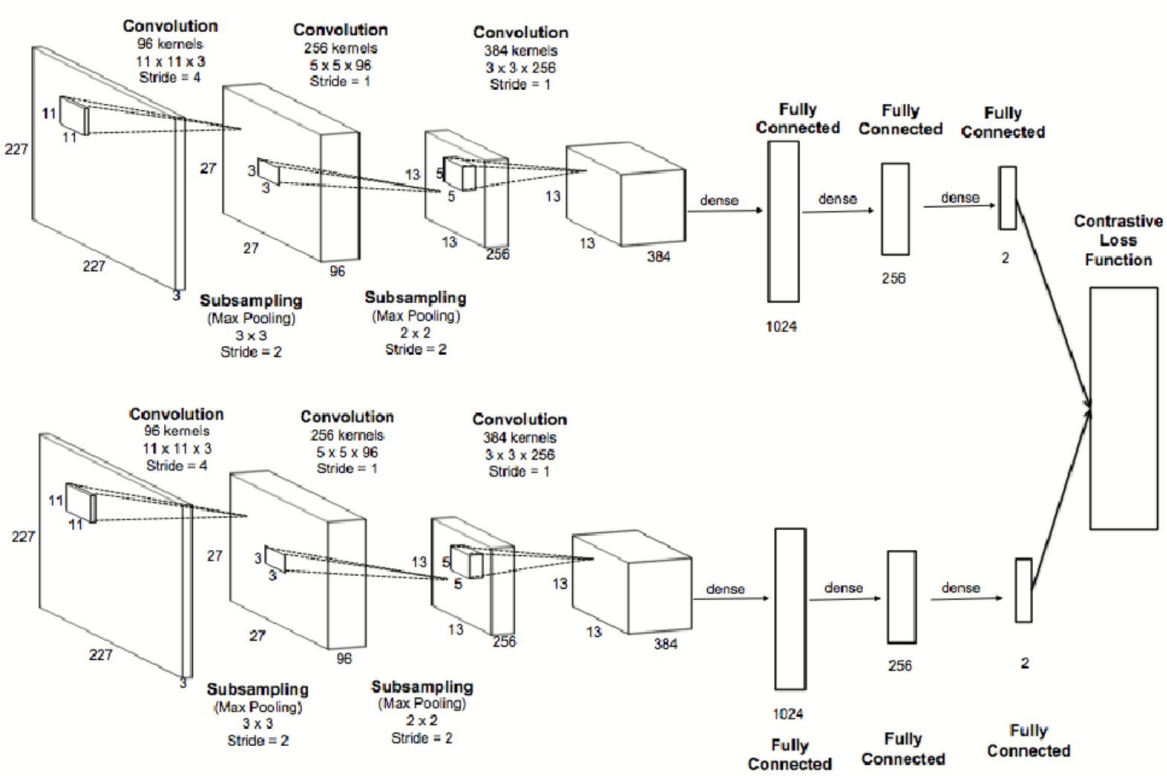
What are Siamese Networks used for?

- Siamese Networks are very useful for comparing images or image similarity tasks.
- Examples of these are Signature Verification, Face Recognition and Finger Print Matching.
- Siamese Networks give a simple binary output, Yes if the images match or No if they do not match.

High Level Diagram of a Siamese Network



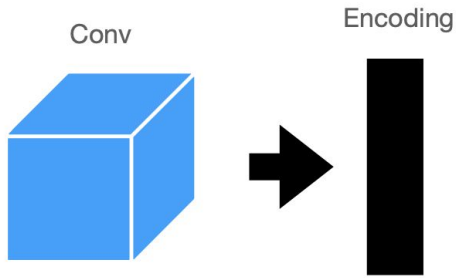
High Level Diagram of a Siamese Network



https://www.semanticscholar.org/paper/A-Deep-Siamese-Neural-Network-Learns-the-Similarity-Rao-Wang/3e16932979250e66cd2cb4d8c9a5411e195273be/figure/0

Siamese Network Architecture

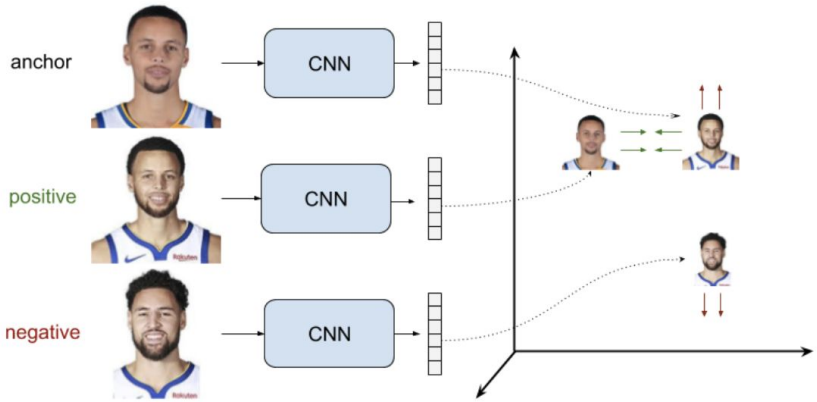
- Notice that the output of Conv Networks in a Siamese Networks are a flat matrix.
- This is the embedding layer or encoding which can be considered the features extracted from that image.
- The Differencing Layer is where we can find the difference using Euclidian or Cosine Difference between the matrices produced by each network.
- We can then use a loss function to assess whether the Siamese Networks made the right decision.



Siamese Networks Loss Functions

Popular Loss functions used when training Siamese Networks are:

- Triplet Loss - where a baseline input is compared to a positive input and a negative input. The distance from the baseline input to the positive input is minimized, and the distance from the baseline input to the negative input is maximized
- Contrastive Loss
- Binary Cross-Entropy

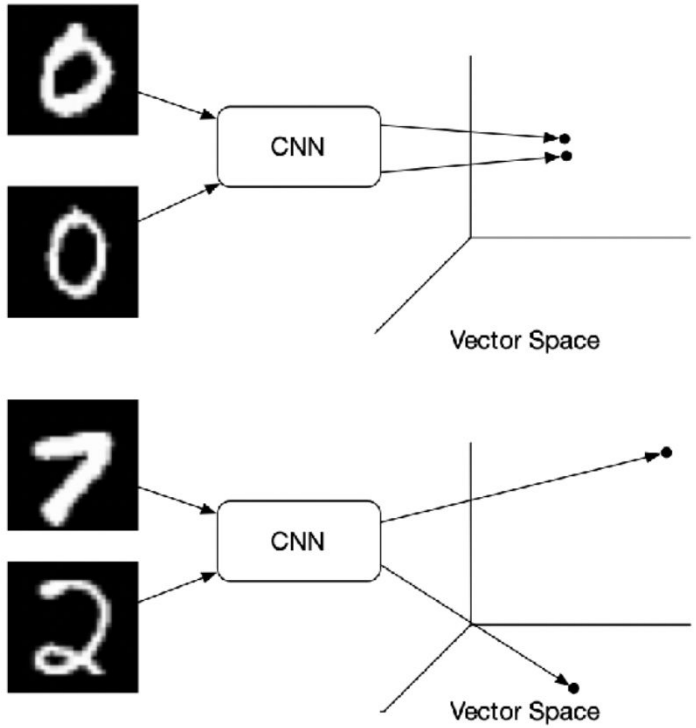


https://commons.wikimedia.org/wiki/File:Triplet_loss.png

Siamese Networks Loss Functions

Popular Loss functions used when training Siamese Networks are:

- Triplet Loss
- Contrastive Loss - The goal of a siamese networks is to differentiate between pairs of images. Contrastive refers to the fact that these losses are computed contrasting two or more data points representations.
- Binary Cross-Entropy



Siamese Networks Loss Functions

Popular Loss functions used when training Siamese Networks are:

- Triplet Loss
- Contrastive Loss
- Binary Cross-Entropy - given the output of our Siamese Model is binary, i.e. similar or dis-similar, using binary cross-entropy loss function is often the obvious default choice.

Training Siamese Networks

Dataset Preparation

- We start the training process by firstly preparing our data by creating Image Pairs.
- We create two types of pairs:
  - Positive data pair is when both the inputs are the same class
  - Negative pair is when the two inputs are difference classes

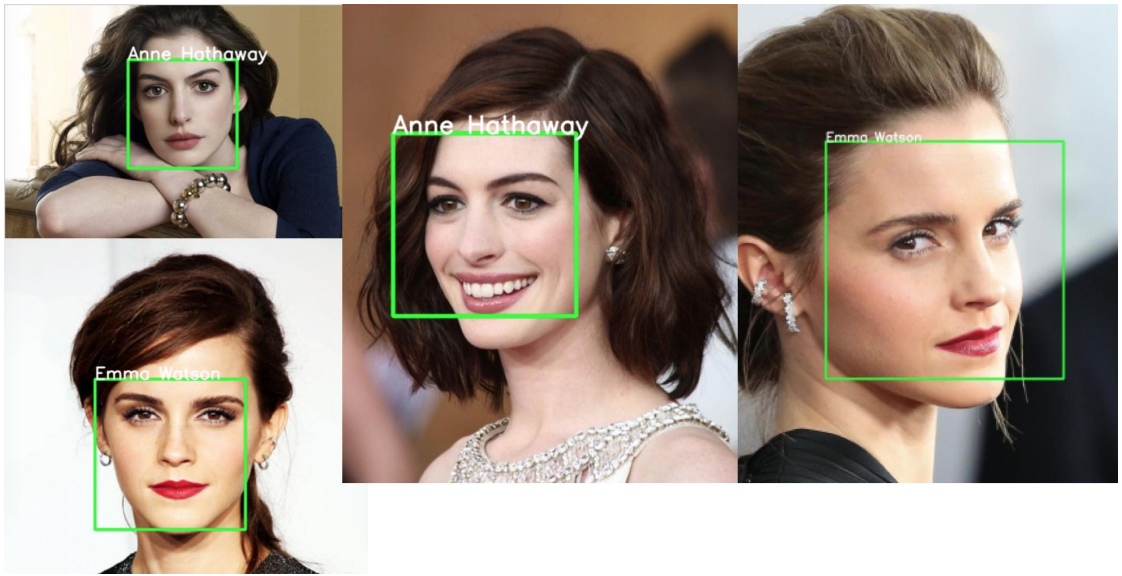
Building the Network

- We build a CNN that outputs the feature encoding or embedding using a fully connected layer.
- We build the sister CNN's will have the same architecture, hyperparameters, and weights.
- We then build the differencing layer to calculate the Euclidian distance between the output of the two CNN subnetworks encoding.
- The final layer is a fully-connected layer with a single node using the sigmoid activation function to output the Similarity score.
- Compile the model using one of the loss functions (Contrastive or Triplet Loss work well).

Facial Recognition

What is Facial Recognition?

- The ability to automatically attach an individual's identity to a face.
- Human is great at this, but can machines do it?



Similar Face…

Similar Face…

Facial Recognition using Deep Learning

- Siamese Networks can be used for Facial Recognition.
- Two popular Deep Learning Facial Recognition Networks:
  - VGGFace
  - FaceNet

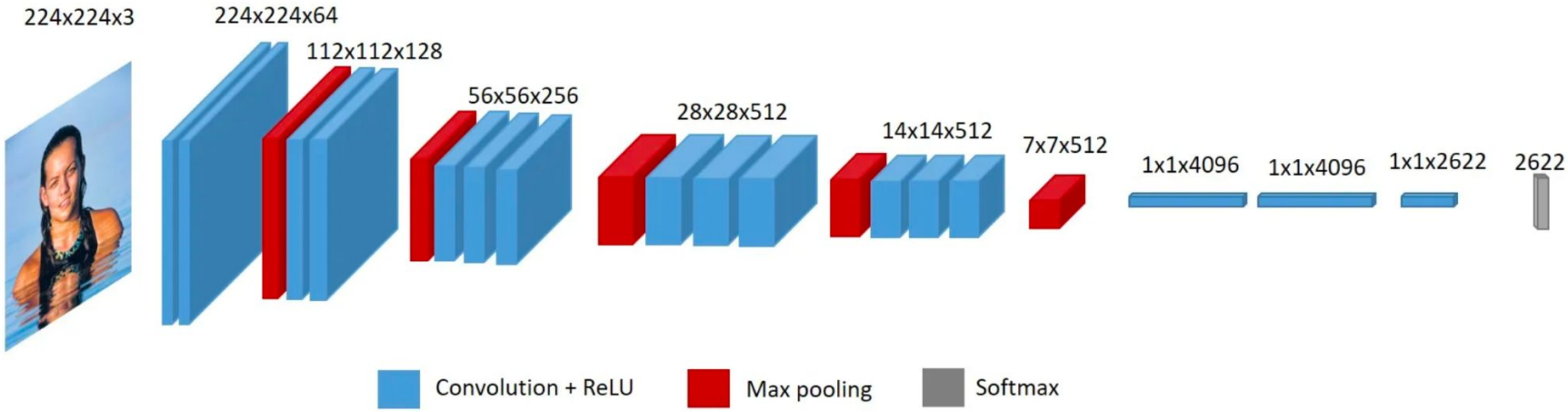
VGGFace

- VGGFace was introduced by Oxford University Researchers Omkar M. Parkhi, Andrea Vedaldi and Andrew Zisserman in their paper titled ‘Deep Face Recognition) in 2015
- They used Triplet Loss and an embedding vector of 2,622 or 1,024 (depending on the configuration) with input image size being 224 x 224

Figure 1: Example images from our dataset for six identities.

https://www.robots.ox.ac.uk/~vgg/publications/2015/Parkhi15/parkhi15.pdf

VGGFace Architechture



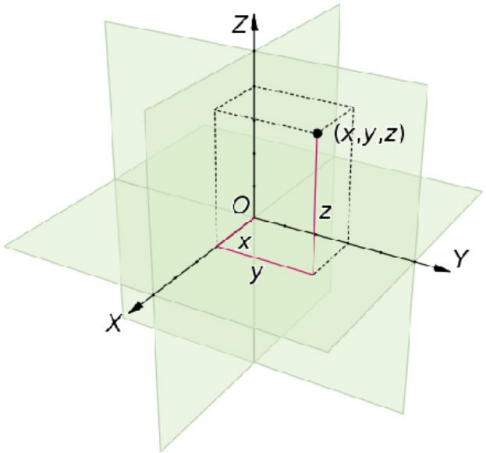
Dataset	Identities	Images
LFW	5,749	13,233
WDRef [4]	2,995	99,773
CelebFaces [25]	10,177	202,599

Dataset	Identities	Images
Ours	2,622	2.6M
FaceBook [29]	4,030	4.4M
Google [17]	8M	200M

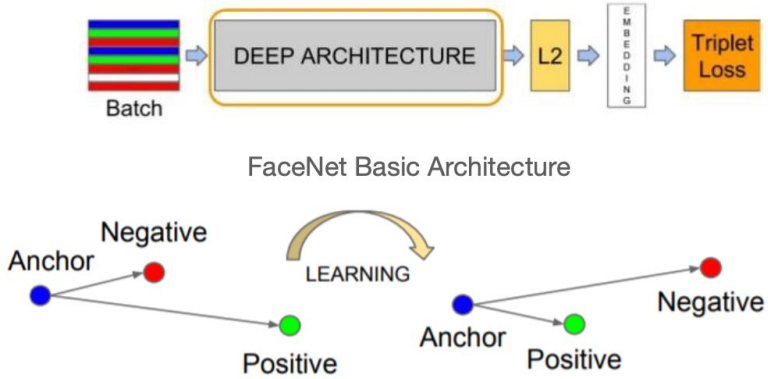
Table 1: **Dataset comparisons:** Our dataset has the largest collection of face images outside industrial datasets by Goole, Facebook, or Baidu, which are not publicly available.

FaceNet

- FaceNet was first introduced by Google in 2015
- It transforms a face into a 128 dimension Euclidian space embedding
- Uses the triplet loss function



3 Dim in euclidian space

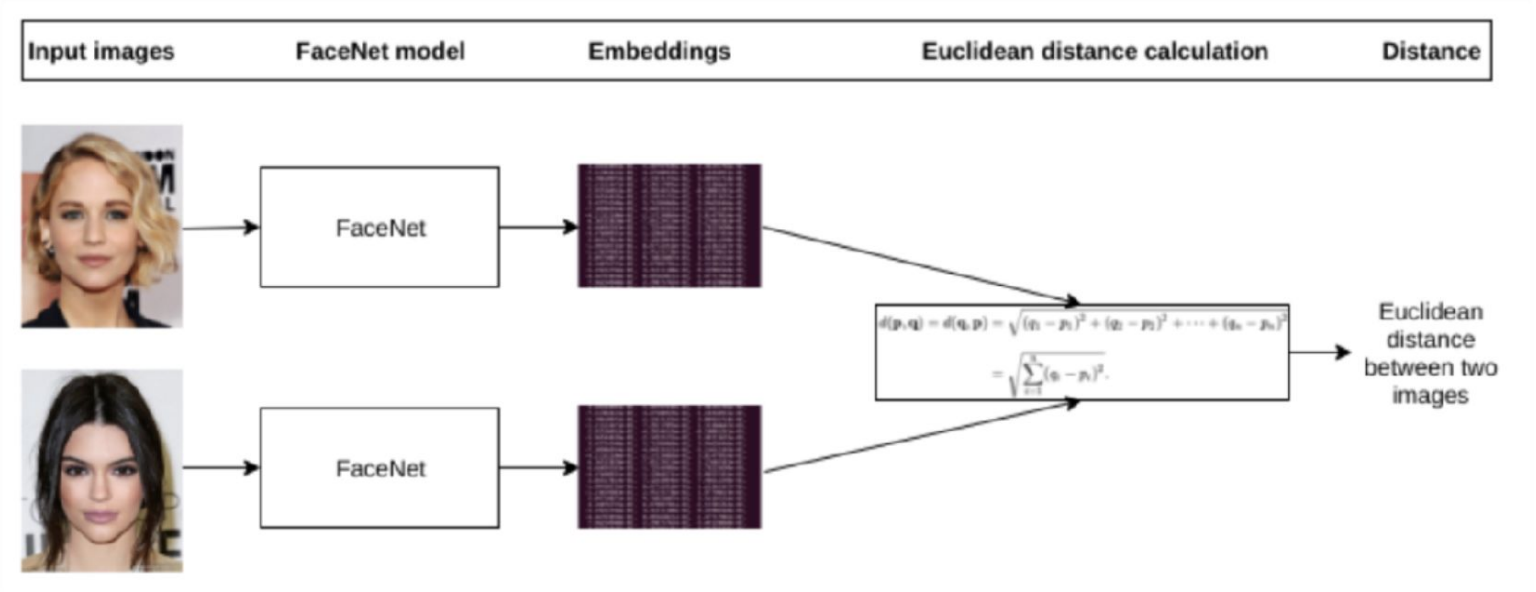


FaceNet Basic Architecture

Triplet Loss Training

https://arxiv.org/pdf/1503.03832.pdf

FaceNet



One shot learning using FaceNet

Implementation

Download notebook at:
https://github.com/albert831229/nchu-computer-vision/tree/main/113/night