# CSC413 Report: Siamese Neural Networks for Facial Recognition

**Andrew Magnuson**
Department of Applied Science & Engineering
University of Toronto
Toronto, ON, M5R 0A3
andrew.magnuson@mail.utoronto.ca

**Carol Meng**
Department of Computer Science
University of Toronto
Toronto, ON, M5R 0A3
carol.meng@mail.utoronto.ca

**Efe Tascioglu**
Department of Applied Science & Engineering
University of Toronto
Toronto, ON, M5R 0A3
efe.tascioglu@mail.utoronto.ca

## Abstract

Learning facial recognition at scale presents serious challenges for traditional machine learning approaches due to the need for dynamic class expansion and the limited amount of data available for any one class. One-shot or few-shot learning [5] is a standard example of this problem, and is explored in detail in this paper. We thoroughly review the application Siamese Neural Networks (SNNs) in the task of facial recognition, comparing their performance against a baseline CNN model on the CelebA dataset [6]. SNNs, with their shared weights and ability to create a semantically rich embedding space, as particularly suited to the task of one-shot and few-shot learning. This presents a critical advantage over CNNs, which require extensive fine-tuning for new identities. Our approach involves a Siamese architecture over a triplet loss function in embedding space, which, via thorough data augmentation and evaluation, allows us to demonstrate the adaptability and scalability of SNNs to entirely novel sets of identities from the training set. The results underscore the potential of SNNs for scalable, efficient facial recognition systems in environments with frequently updating identity pools.

## 1 Introduction

Facial recognition is the task of differentiating individuals using images of their face and is an important topic with a variety of uses, particularly surrounding security. Due to the variety of lighting conditions, expressions, and angles present for any one image of a face, machine learning has proven itself an effective tool for tackling such a task. Siamese neural networks (SNNs) in particular have found great success, as their ability to identify faces from few images of a reference affords the end user a flexible system without sacrificing accuracy. Their unique architecture, designed around coupled networks and a Euclidean distance metric, allows them to capture discriminative features that are effective over novel identities of individuals. SNNs present a distinct advantage over traditional CNN architectures, which are fixed in their output space, and thus require extensive fine-tuning in order to adapt to a new desired set of identities. In this paper, we demonstrate the utility of siamese neural networks on the task of facial recognition over the CelebA dataset [6], outperforming a baseline traditional CNN model on a standardized accuracy measure from Schroff et al. [7]. We also show

37th Conference on Neural Information Processing Systems (NeurIPS 2023).

how each model addresses the challenges of dynamic class expansion, and offer a practical solution for environments requiring frequent updates to the pool of individuals.

## 2 Background & Related Work

Siamese networks were initially devised for signature verification [2]. They process pairs of inputs through subnetworks with identical architectures and weights, with a loss function that encourages the formation of a semantically meaningful output space of embeddings. They prove highly effective for tasks like face recognition and few-shot learning, as they are able to distill a few examples into a context-rich embedding for later classification without suffering the typical losses in accuracy that come from a small dataset [5]. This makes them good at rapid adaptation to new data without extensive retraining or dependency on large labeled datasets.

In the FaceNet paper, Schroff et al. [7] used a Siamese variant to achieve groundbreaking performance in facial recognition by mapping images into a space where distances reflect facial similarities. This adaptability demonstrates scalability, ability to learn from limited data, and enhanced model generalizability to new examples. Triplet Networks are "deep architectures" in which the loss of the model is defined through the comparison of the outputs of multiple networks [3] [7].

Koch et al. [5] propose a method for learning Siamese networks that outperformed other deep models with state-of-the-art performance on one-shot classification tasks. Their model consists of two identical components which are fed into an $L_1$ distance layer with a sigmoid activation. This allows their model to create a logistic prediction, $p$, of whether the two examples are the same or different. This is then able to generalize to unseen categories based on the model's learned feature mappings.

## 3 Data Processing

For our project, we utilized the CelebA dataset, which comprises over 200,000 face images and 10,177 identities. Each image features a cropped and aligned celebrity face, labeled with an integer to denote the celebrity's identity [6]. To organize the images into training, validation, and test sets, we relied on the provided 'list_eval_partition.csv'. This file specifies the designated set for each image. Following this initial division, we employed the 'identity_CelebA.txt' file from the dataset to further categorize the images within each set (training, validation, test) based on identity, ensuring that images of the same individual did not cross over into multiple sets. This step was critical to prevent the formation of empty classes and to maintain the integrity of our model's evaluation.

Table 1: Distribution of Identities Across Different Data Partitions

| Dataset | Number of Identities |
|---|---|
| Train Identities | 8192 |
| Validation Identities | 985 |
| Test Identities | 1000 |
| Total Identities | 10177 |

Subsequently, we applied data augmentation techniques to the images in the training set to enhance model robustness, including a random horizontal flip, random rotation of up to 20 degrees, random shear of up to 10 degrees, and normalization to the mean and standard deviation of ImageNet. We then structured the training data into tuples of three tensors: an 'anchor' (a tensor of an image), a 'positive' (a tensor of another image of the same identity as the anchor), and a 'negative' (a tensor of an image from a different identity). Given the nature of this triplet formation, the number of negative samples exceeded the positives. Finally, we loaded these tuples into a PyTorch DataLoader, preparing them for batch processing in our model.

## 4 Model Architecture

Our model architecture (Figure 2) is a Siamese network framework using a pretrained PyTorch ResNet50, which is a standard residual convolutional neural network introduced in He et al. [4]. We


Figure 1: Example Triple Generated from our Training Set. Note the augmentation visible in the form of rotation.

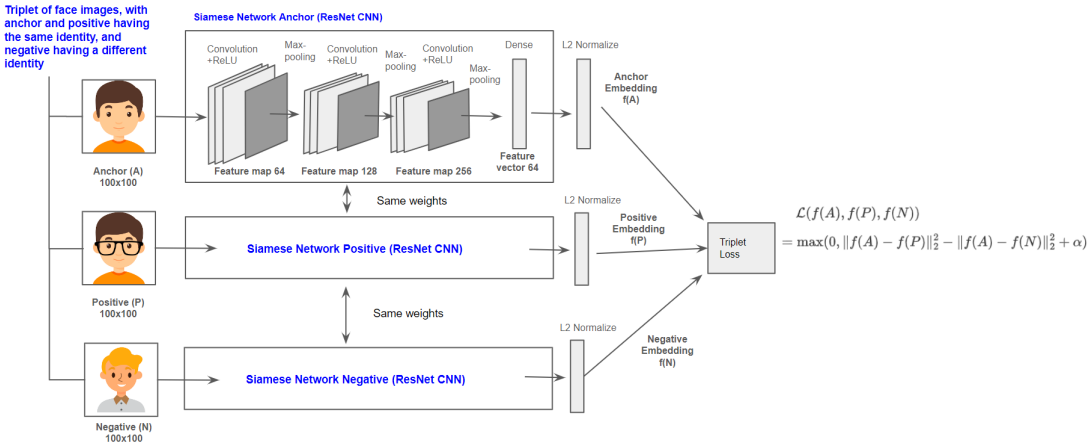


Figure 2: Model Architecture: A Siamese Network consisting of a Deep ResNet following by $L_2$ normalization, then triplet loss

have modified the ResNet50 by not passing the input through the fully connected classification layer at the end in the forward function, allowing the network to output feature vectors rather than class predictions. Keeping in line with Siamese neural network practice, we pass an image $x_i^a$ (anchor), an image of the same class $x_i^p$ (positive), and an image of a different class $x_i^n$ (negative) through the same modified ResNet50 weights to obtain their respective feature vectors in $d$-dimensional Euclidean space. We used 256-dimensional embedding vectors for this implementation. For training and accuracy calculation, we use `torch.nn.functional.pairwise_distance` to assess the Euclidean distances between the feature vectors, $f(x) \in \mathbb{R}^d$, of the anchor and positive images, and between the anchor and negative images. We utilize the triplet margin loss function, which encourages the model to maintain a margin, $\alpha = 1$, between the distances for positive and negative pairs in all triplets, $\mathcal{T}$ [7]:

$$\|f(x_i^a) - f(x_i^p)\|_2^2 + \alpha < \|f(x_i^a) - f(x_i^n)\|_2^2 \ \forall (f(x_i^a), f(x_i^p), f(x_i^n)) \in \mathcal{T} \tag{1}$$

$$\implies \mathcal{L} = \sum_i^N \left[ \|f(x_i^a) - f(x_i^p)\|_2^2 - \|f(x_i^a) - f(x_i^n)\|_2^2 + \alpha \right] \tag{2}$$

Accuracy is determined by comparing how frequently the distance between the anchor and positive images is less than the distance between the anchor and negative images across all features. This metric is used by Schroff et al. [7] and Koch et al. [5].

Our dataset handling involves the creation of image triples (anchor, positive, negative), where the positive images share the same identity as the anchor, and the negative images have a different identity. Data augmentation is applied selectively to enhance the variability and robustness of our model. The augmented data and the triples are managed through a custom `TripletCelebADataset` class, which handles the augmentation logic and indexing of identities to ensure accurate triplet formation.

This structured dataset is then loaded into PyTorch DataLoaders, facilitating efficient batch processing during training, validation, and testing phases. The training process involves adjusting the learning rate, applying triplet loss, and periodically evaluating the model's performance on the validation


Figure 3: The Triplet Loss minimizes the distance between an *anchor* and a *positive*, both of which have the same identity, and maximizes the distance between the *anchor* and a *negative* of a different identity [7]

set to monitor improvements and make necessary adjustments. The architecture's effectiveness is underscored by its capacity to learn discriminative features that robustly distinguish between different identities, which is critical for facial recognition tasks.

## 5 Baseline Model

The baseline model that we are using to evaluate our SNN is a standard CNN network utilizing the same pretrained ResNet50 architecture. Instead of returning an embedding, like the SNN, the baseline CNN has a final layer that creates a vector of logits over the class space of the model. This allows the baseline CNN to express a categorical distribution over the identities, which can have its argmax taken as the predicted identity. The biggest downside with this network is that it isn't easy to scale up with new individuals, as the output of the network needs to be re-scaled, and the network needs to retrain for all the new individuals.

## 6 Quantitative Results

Table 2: Positive and Negative Distances Validation Results on Siamese

| Metric | Value |
|---|---|
| Positive $<$ Negative | 86.3% |
| Positive $<$ 1 | 1.7% |
| Positive Distance Mean | 2.592 |
| Positive Distance Standard Deviation | 0.979 |
| Negative Distance Mean | 4.779 |
| Negative Distance Standard Deviation | 1.726 |

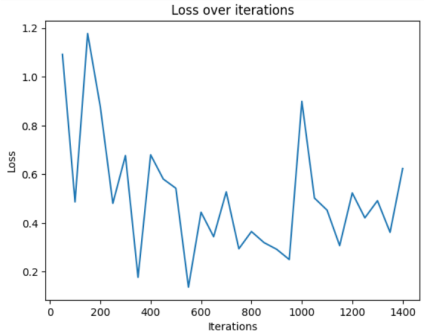


Figure 4: Loss of our Siamese Network trained on 1400 iterations, batch size 48

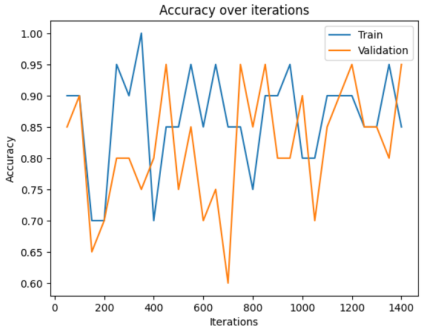


Figure 5: Accuracy of our Siamese Network trained on 1400 iterations, batch size 48


Figure 6: Loss of Baseline Model (CNN) (iterations 6000-8000)

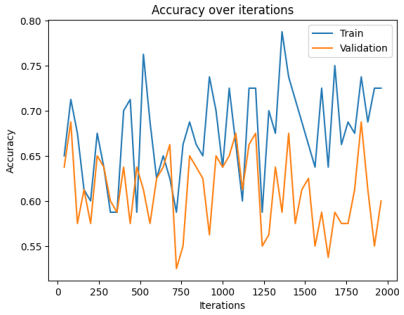


Figure 7: Accuracy of Baseline Model (CNN) (iterations 6000-8000)

Figures 4 and 5 present the results of our Siamese experiment, with a lowest loss of 0.2496 and validation accuracy of 86.3%. Figures 6 and 7 present the results from our baseline CNN experiment, with a lowest loss of 0.4561 and validation accuracy of 68.5%. [1] Table 2 presents the results of the mean and standard deviations of the positive and negative pair distances over the validation set for the Siamese model.

[1] It should be noted that the loss measures are not equivalent between the two models, but we do use an accuracy metric that is identical to that of Schroff et al. [7] and Koch et al. [5].

## 7 Qualitative Results


Figure 8: Anchor and Positive Pair Example: Distance 2.71 < 3.78, images are similar.

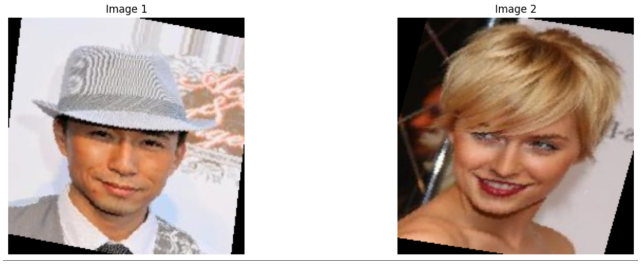


Figure 9: Anchor and Negative Pair Example: Distance 5.97 > 3.78, images are not similar.

When training our Siamese Network, we observed considerable volatility in the loss values, as evidenced by Figure 4. This nature of the loss curve may arise from several factors, such as the triplet loss function we used being hard to train and balance to achieve stability. Despite this, the network only converged slightly within the span of 1400 iterations, with Figure 5 illustrating fluctuations in accuracy for both training and validation datasets. Interestingly, the Siamese Network showed a relative improvement over time, maintaining approximately 85%-90% accuracy by 1400 iterations, suggesting a potential for faster convergence compared to the CNN model. This may be due to the network's ability to discern discriminative features that aid in individual separation.

Figures 8 and 9 present a selection of image pairs and their distance metrics, showcasing instances where the network was successful and unsuccessful. Notably, the Siamese network is good at recognizing large scale differentiating features, such as hair colour and skin tone, and structural elements like glasses or facial shape.

## 8 Discussion

Overall, we see a clear greater accuracy of the Siamese model over the CNN baseline, alongside an evidenced ability to produce differing distance means for positive and negative novel identities from the validation set. The standard deviations on these distances are, for what compute ability and network size we are working with, quite reasonable, indicating that they can relatively simply be put to use in inference using a simple maximum likelihood estimate hypothesis test setup. For Siamese, the bound ends up being a distance of 3.78.

Due to the high number of identities in the dataset, both networks faced struggles differentiating between faces. Both the standard CNN and Siamese Network had an accuracy of < 1% when trying to pick out a specific name to associate with the given image from the entire dataset. For Siamese networks, this is because, despite having a high success rate when *comparing* images, the chances that it gets this comparison correct for many thousands of faces is quite low. Due to the sheer quantity of classes that exist in the CelebA dataset, it is generally quite hard for a model to identify an individual, especially in a one or few-shot learning context. For the compute ability we worked with, we find the performance of the transfer-learned Siamese network quite promising, and indicative of potential success of the architecture in at-scale applications.

Despite these hurdles, the Siamese Network demonstrated a comparative advantage. By focusing on learning discriminative features rather than individual recognition, the SNN showed enhanced performance. However, the difficulty associated with implementing the triplet loss, which is vital for the SNN's learning process, cannot be understated. This complexity arises from the necessity to select appropriate and informative triplets during training [7], which is essential for the model to learn meaningful distinctions between different individuals' faces. This process of triplet mining took considerable additional research to implement efficiently and without overloading the GPU's memory, and a separate extension of our `ipynb` had to be created to accommodate the change. We thought it more appropriate to submit our preliminary results without triplet mining, but would like to revisit the topic in the future, as it would allow the model to focus on the few cases where distinguishing faces is difficult, hence boosting performance.

In conclusion, while neither model achieved convergence, the Siamese Network's tendency toward faster convergence relative to the CNN, coupled with its capacity to handle images it has not previously encountered, underscores its suitability for tasks demanding high-level feature discrimination, even in the face of substantial class imbalance and limited training data availability.

## 9 Ethical Considerations

There are many ethical considerations when dealing with use of automated facial recognition, primarily centered around the two central pillars of security and privacy. Firstly, facial recognition technologies have been used can be used for protection purposes. From locking phones, to identifying criminals, to similar technologies being used to detect fraudulent signatures, this technology has a large capacity to increase the security we feel in our daily lives. However, there are many privacy concerns when using this technology. Facial recognition models require large amounts of data, which incentivizes the scraping of images to train them. Furthermore, facial recognition can be used rapidly identify individuals, which has very large consequences. According to Almeida et al. [1], facial recognition opens the door to surveillance methods that pose great danger to human rights, privacy and data protection, especially surrounding their questionable use by law enforcement and government agencies. These technologies must be closely monitored and new legislation passed to reflect these new use cases. Overall, this technology has advantages and disadvantages, but must be used very carefully due to the big dangers it poses.

## 10 Contributions

The team consists of the three authors listed on the front page alongside OpenAI's ChatGPT 4. All three human authors contributed equally to the writing of the paper, with various authors drafting, revising, and polishing sections of the report. There were no specific assignments to sections for writing. ChatGPT 4 was used as an assistive tool for LaTeX formatting of tables, but *did not* contribute to large-scale ideas or detailed writing for the report. ChatGPT 4 contributed to the notebook by writing initial code for showing images of the triples and initial code for the triplet dataset class. All GPT 4 code was thoroughly reviewed, touched up, and commented before further use. Carol worked on the `ipynb` structure, laying out the entire pipeline, starter SNN model, and train script. She also did extensive work on preprocessing and setting up the CelebA dataset to load quickly and consistently into the notebook. Andrew scanned the notebook for bugs, refactored code for readability, added markdown to code cells, and wrote code to visualize images and results. Andrew also adjusted the SNN mechanism from Carol's starter model. Efe adjusted the dataloading and transformations, while implementing several versions of a baseline CNN, eventually settling on what is seen in the final notebook. He wrote custom training, testing, and models, extending from Carol's notebook structure. All three members worked on a separate notebook to implement triplet mining as per Schroff et al. [7], but could not get the notebook in working order for the final submission.

## References

[1] D. Almeida, K. Shmarko, and E. Lomas. The ethics of facial recognition technologies, surveillance, and accountability in an age of artificial intelligence: a comparative analysis of us, eu, and uk regulatory frameworks. *AI and Ethics*, 2(3):377–387, July 2021. ISSN 2730-5961. doi: 10.1007/s43681-021-00077-w. URL `http://dx.doi.org/10.1007/s43681-021-00077-w`.

[2] J. BROMLEY, J. W. BENTZ, L. BOTTOU, I. GUYON, Y. LECUN, C. MOORE, E. SÄCKINGER, and R. SHAH. Signature verification using a "siamese" time delay neural network. *International Journal of Pattern Recognition and Artificial Intelligence*, 07(04):669–688, Aug. 1993. ISSN 1793-6381. doi: 10.1142/s0218001493000339. URL http://dx.doi.org/10.1142/s0218001493000339.

[3] T. Chen, S. Kornblith, M. Norouzi, and G. Hinton. A simple framework for contrastive learning of visual representations, 2020. URL https://arxiv.org/abs/2002.05709.

[4] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition, 2015. URL https://arxiv.org/abs/1512.03385.

[5] G. Koch, R. Zemel, and R. Salakhutdinov. Siamese neural networks for one-shot image recognition. In *Proceedings of the 32nd International Conference on Machine Learning*, volume 37 of *JMLR: Workshop and Conference Proceedings*, Lille, France, 2015. JMLR.org. Copyright 2015 by the author(s).

[6] Z. Liu, P. Luo, X. Wang, and X. Tang. Deep learning face attributes in the wild, 2014. URL https://arxiv.org/abs/1411.7766.

[7] F. Schroff, D. Kalenichenko, and J. Philbin. Facenet: A unified embedding for face recognition and clustering. In *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, June 2015. doi: 10.1109/cvpr.2015.7298682. URL http://dx.doi.org/10.1109/cvpr.2015.7298682.