

Human Kinship Classification Using Images & Networks

2025-02-20

GitHub Repository

You can find the code for this project on GitHub: [GitHub Repository](#)

Abstract

Today, kinship detection is achieved with high accuracy through DNA-based methods, which allow for the identification of blood relationships between individuals. However, this task is also routinely performed by humans through visual perception, as people often remark that children resemble their parents. Visual perception and classification can similarly be facilitated by neural networks, which have proven effective in image classification and segmentation, particularly with Convolutional Neural Networks (CNNs) that are tailored for working with images. This project explores the potential of simple neural network architectures to visually classify two individuals as relatives based on their facial images. The results from different architectures are presented, highlighting the superiority of using both image embeddings and their similarities, rather than relying on one method alone. Evaluations are provided based on relationship type to explore how they influence the difficulty of detecting kinship. Overall, the findings demonstrate the potential of neural networks for visual kinship detection, with parent-child relationships being the easiest to identify, followed by siblings, and grandparent-grandchild relationships proving the most challenging.

1 Introduction

Human kinship classification, or kinship identification, refers to the ability to determine whether two individuals share a blood relationship. The article "Human Kin Detection" by Lieberman et al. [1] argues that humans have evolved specific mechanisms to identify relatives for survival and other biological reasons. Regardless of the underlying processes, it is widely accepted that people tend to resemble their relatives in both appearance and behavior due to inherited genetic traits.

Today, kinship identification is commonly performed using advanced techniques. The most precise method relies on DNA analysis, which provides unique genetic information for accurate kinship classification [2]. However, a far more common and instinctive approach is visual recognition—a method used by humans daily. It is not uncommon for someone to remark that a child strongly resembles their parents, highlighting how facial features

play a role in perceived kinship.

With the rapid advancements in deep learning over the past few decades, neural networks have become capable of performing complex tasks that were previously beyond the reach of traditional machine learning. In particular, convolutional neural networks (CNNs) have revolutionized image-based tasks such as classification, segmentation, and object detection.

This project evaluates the effectiveness of widely used CNN architectures for visual kinship identification by analyzing facial images. The dataset used for model training and testing is the publicly available Families in the Wild dataset. Comparison between different Siamese architectures are presented. The architectures utilize an Inception-based model pretrained specifically on human faces. The best performing architecture, which combines the image embeddings with similarity calculations is then used to evaluate its performance with kin classification across the different kin relationships present in the dataset. Finally, a ranking of the different kin relationships is presented, illustrating how easily the model can identify kinship based on images of various kin types.

2 Related Work

The resemblance between individuals and their parents has been extensively studied, leading to the development of accurate kinship classification methods using DNA, which reflects genetic similarities inherited from ancestors [2]. However, significant research has also explored our innate ability to detect kin relationships, suggesting that humans have evolved to recognize kin automatically, relying in part on visual features and resemblance [3, 1].

Due to the strong visual similarity between blood-related individuals, image-based methods have been studied and developed for kinship classification. Early approaches focused on extracting and comparing facial features [4]. With the rise of convolutional neural networks (CNNs)—which have demonstrated exceptional performance on image-based tasks—these models have also been applied to kinship detection, yielding promising results [5, 6].

3 Methods

The dataset is divided into training, validation, and test sets for model training, with the validation set used to track improvements in the area under the curve (AUC) score. Three distinct Siamese network architectures are evaluated for kinship classification, all leveraging the

InceptionResNetV1 model, pretrained on the **VGGFace2** dataset, containing large number of human face images. This setup aims to identify the most effective approach for comparing embedded images and achieving accurate kinship classifications.

3.1 Data

The dataset used for training and evaluation is the Families In the Wild (FIW) dataset, see 6 for the dataset link on Kaggle. This dataset includes images of 689 families, each with labeled pairs representing various kinship relationships between family members. The relationships are labeled as follows:

- Brother - Brother (bb)
- Sister - Sister (ss)
- Siblings (sibs)
- Father - Daughter (fd)
- Father - Son (fs)
- Mother - Daughter (md)
- Mother - Son (ms)
- Grandfather - Granddaughter (gfgd)
- Grandfather - Grandson (gfgs)
- Grandmother - Granddaughter (gmgd)
- Grandmother - Grandson (gmgs)

Note that the category "Siblings" (sibs) lacks a detailed explanation in the dataset documentation. It is assumed that this category refers to brother-sister sibling relationships, given the presence of the specific "Brother - Brother" (bb) and "Sister - Sister" (ss) relationships.

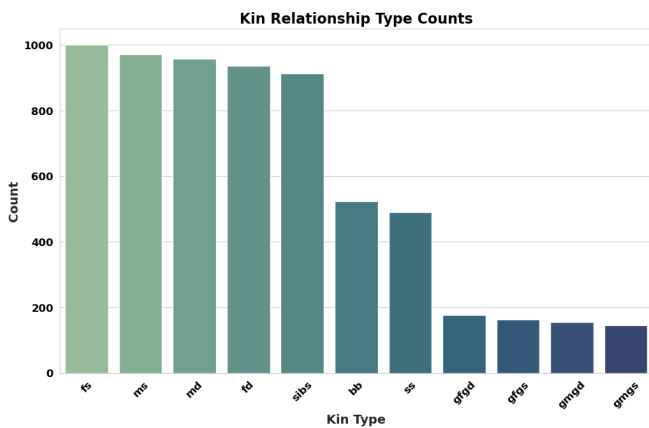


Figure 1: Distribution of kin relationships in the dataset with the count representing the number of individual pairs with that relationship type

Figure 1 illustrates the imbalance in the representation of various kinship relationships within the dataset. Notably, the relationships between grandparents and their grandchildren are the least represented, with the fewest pairs of individuals matching these kinship categories.

3.2 Architectures

Three Siamese architectures are trained separately to evaluate the best approach for combining embedded images for kinship classification. All three architectures use the **InceptionResNetV1** model, pretrained on the **VGGFace2** dataset, to encode face images of two individuals, generating embeddings of size 512. The task remains the same across all models: classifying whether two people are kin-related based on their face images. Additionally, the feedforward network used for classification is identical across all architectures, except for variations in input dimensions.

3.2.1 SEC

The first architecture, SEC (Siamese Embedding Concatenation), shown in Figure 2, concatenates the two embeddings, forming a single vector of size 1024 ($2 \times$ embedding size). This vector is then passed through fully connected layers to produce a final binary classification.

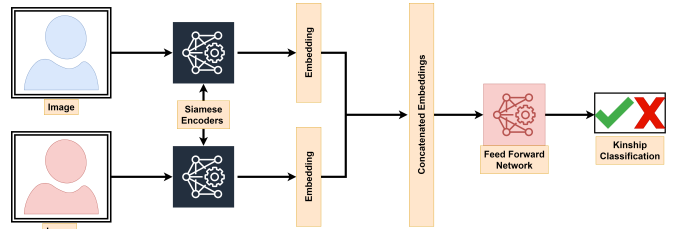


Figure 2: The SEC Architecture

3.2.2 SSC

The second architecture, SSC (Siamese Similarity Concatenation), shown in Figure 3, computes three similarity metrics between the embeddings: cosine similarity, Euclidean distance, and element-wise squared difference. These similarity values are concatenated, forming a vector of size 514 ($2 \times$ embedding size + 2), which is then fed into the fully connected layers for classification.

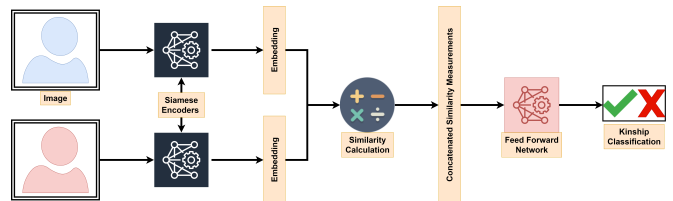


Figure 3: The SSC Architecture

3.2.3 SESC

The third architecture, SESC (Siamese Embedding Similarity Concatenation), shown in Figure 4, combines the approaches of the previous two models. It concatenates both the embeddings and the computed similarities into a single vector of size 1538 ($3 \times$ embedding size + 2),

which is then passed to the fully connected network for classification.

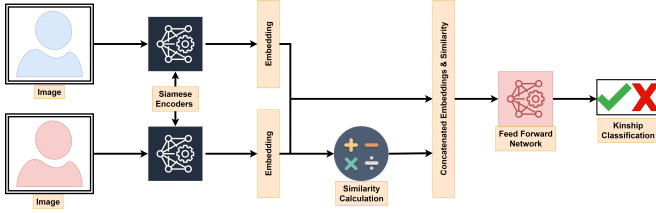


Figure 4: The SESC Architecture

3.2.4 Classifier

The fully connected network used for binary classification follows the structure outlined in Figure 5.

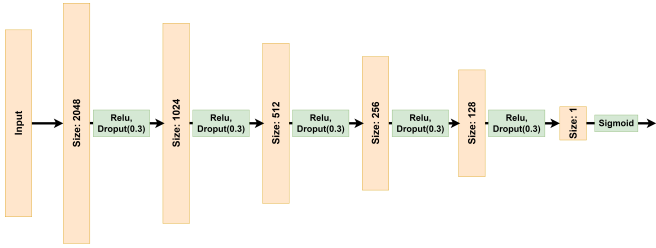


Figure 5: The last feed forward classifier network scheme.

3.3 Data Splitting & Training

The families are split into train, validation and test set as follows:

- Train (80%): 565 Families
- Validation (10%): 62 Families
- Test (10%): 62 Families

In each set, all positive image pairs (pairs of individuals who share a kin relationship) are generated. Specifically, pairs are formed through a Cartesian product of all available images for each individual. Similarly, all negative pairs (pairs of individuals with no kin relationship) are also created using a Cartesian product between people that do not share a kin relationship, but are then randomly sampled to match the number of positive pairs. This ensures that the training, validation, and test sets remain balanced in terms of kin-related examples.

However, the dataset splits are based on families without considering their sizes or the varying number of images available for each member. As a result, the number of positive image pairs and the distribution of kin relationship types vary across splits. This also explains why, despite the validation and test sets containing the same number of families, the number of positive pairs differs, as shown in table 1

| Set | Positive Pairs | Negative Pairs |
|------------|----------------|----------------|
| Train | 195386 | 195386 |
| Validation | 61638 | 61638 |
| Test | 20147 | 20147 |

Table 1: Image pair counts based on splits

4 Experiments

4.1 Train, Validation and Test Sets Evaluation

Before training and evaluating the models, insights about the data and its characteristics across the splits is necessary.

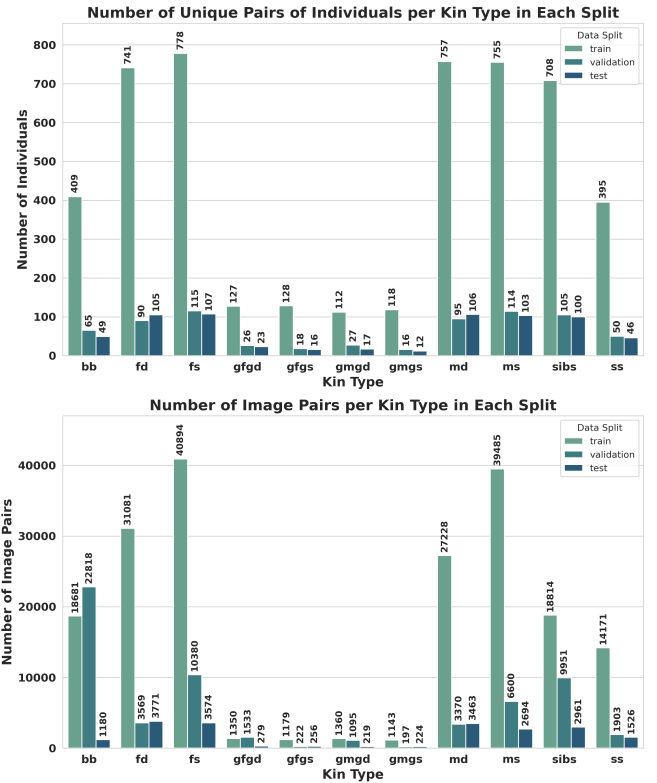


Figure 6: Distribution of kin type across the splits based on unique pairs of people and based on image pairs.

Figure 6 illustrates the distribution of kin types in each split, along with the number of image pairs representing each kin type. The substantial variation in the number of image pairs across splits, particularly the disproportionately high number of bb kin type pairs in the validation set compared to the training set, can be attributed to the varying number of images available for each individual.

Since image pairs are generated through a Cartesian product, individuals with more images contribute significantly more pairs for the kin types they belong to. The large number of bb kin relations in the validation set, for example, is driven by individuals with many images appearing in this split. Additionally, family sizes can vary considerably, further influencing the distribution of kin relationships both within and across splits. This can also drive the number of bb pairs in the validation set to be high, as families which fell in the validation set, has an

increased number of brothers within a single family compared to families in other splits.

Additionally, the plot further highlights the scarcity of grandparent-grandchild pairs in the dataset, making them a minority. As a result, models are less likely to learn to detect kinship in such cases compared to other kin types, where more training examples are available.

4.2 Architecture Comparison

A model using each architecture is trained on the training set, with validation used to monitor performance based on the AUC score. Training continues as long as the validation AUC improves, with early stopping applied using a patience of three epochs. The model achieving the highest AUC on the validation set is then selected for final evaluation on the test set.

The architectures are evaluated on their ability to detect general kin relationship using area under the ROC curve (AUC) and average precision (AP) derived from a precision recall curve. Additionally, their ability to detect general kinship is evaluated by comparing their correct kinship detections for pairs of images of specific kin types to all the negative examples. It is important to note that the evaluation for each specific kin type is significantly imbalanced and skewed, as it compares the results to all negative examples, which do not differentiate between kin types. This is because the models are tasked with classifying the presence of a kin relationship, without needing to identify the specific type of kinship.

Table 2: Evaluation of Different Neural Network Architectures Using AUC and AP Scores

| Kin Type | AUC Scores | | | AP Scores | | |
|----------|------------|--------------|--------------|-----------|--------------|--------------|
| | SEC | SSC | SESC | SEC | SSC | SESC |
| General | 0.789 | 0.797 | 0.856 | 0.750 | 0.787 | 0.835 |
| bb | 0.781 | 0.799 | 0.839 | 0.167 | 0.301 | 0.255 |
| ss | 0.731 | 0.871 | 0.887 | 0.152 | 0.386 | 0.385 |
| sibs | 0.732 | 0.842 | 0.830 | 0.267 | 0.472 | 0.420 |
| fs | 0.829 | 0.809 | 0.875 | 0.431 | 0.463 | 0.548 |
| fd | 0.785 | 0.777 | 0.854 | 0.384 | 0.407 | 0.505 |
| ms | 0.839 | 0.798 | 0.873 | 0.334 | 0.347 | 0.475 |
| md | 0.801 | 0.808 | 0.864 | 0.345 | 0.404 | 0.499 |
| gfgs | 0.679 | 0.391 | 0.726 | 0.018 | 0.010 | 0.025 |
| gfgd | 0.747 | 0.571 | 0.806 | 0.027 | 0.022 | 0.038 |
| gmgs | 0.740 | 0.521 | 0.755 | 0.036 | 0.016 | 0.039 |
| gmgd | 0.808 | 0.640 | 0.823 | 0.092 | 0.117 | 0.095 |

Table 2 shows the superiority of the SESC architecture, especially if focusing on the general kin detection performance which is the task that the models are trained to perform. Thus, its test results will be used for evaluating the ability to perform kin detection.

4.3 Kin Classification Evaluation

Figures 78 Display the ROC and Precision-Recall curves, with the general kin classification highlighted in black. To assess the model’s ability to detect kinship for specific relationships, we use the Average Precision (AP) score.

The Precision-Recall trade-off better reflects the model’s confidence in kin classification across varying thresholds. A higher AP score indicates that the model maintains high precision over a broader range of thresholds.

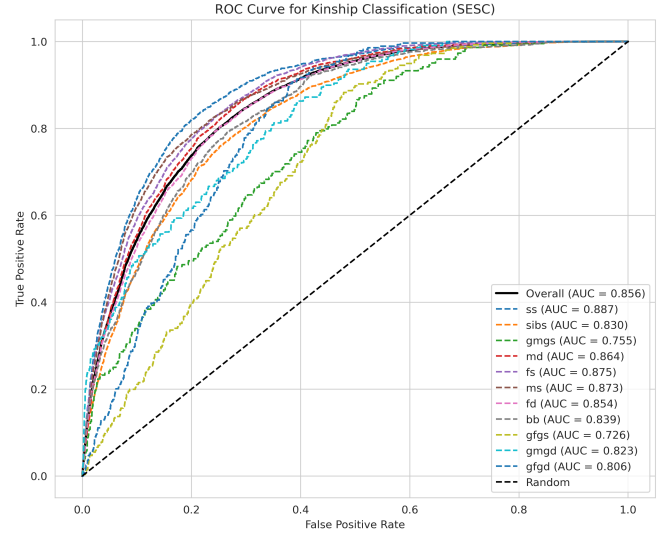


Figure 7: ROC Curve and AUC scores of kin classification using SESC

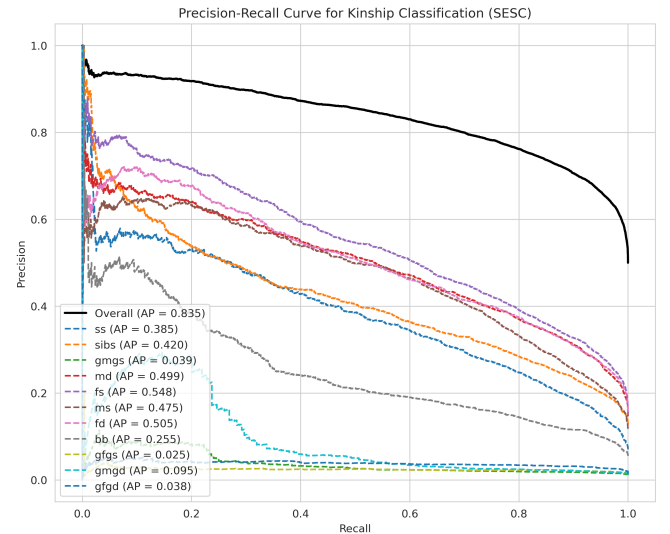


Figure 8: Precision Recall Curve and AP scores of kin classification using SESC

Figure 9 presents the SESC AP scores for each kin type, highlighting the model’s varying performance across relationships. The results indicate that the model most effectively detects kinship in parent-child pairs, followed by siblings, while grandparent-grandchild relationships pose the greatest challenge. This difficulty may be partly due to the smaller amount of data representing this kin type. Additionally, the complexity could stem from the generational gap, as facial features tend to diminish in similarity across generations. While children often strongly resemble their parents, this resemblance becomes less pronounced with grandparents, making it harder for the model to identify kinship.

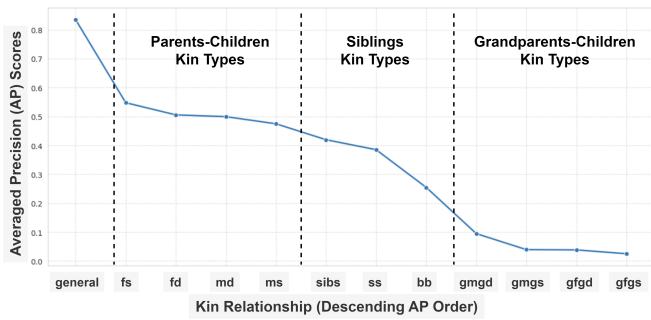


Figure 9: AP Scores for Different Kin Types (Descending Order)

5 Limitations

The data used in this project, while large, is not sufficiently extensive for image-based classification tasks, as neural networks require large datasets to perform effectively—particularly for kinship detection, a task that is inherently challenging to perform visually. Additionally, the dataset is imbalanced, with grandparent-grandchild relationships being underrepresented compared to other kin types.

Additionally, the quality of the images varies significantly, with some being of high quality, while many others are of lower quality, and some are black and white, while others are in color. These variations make the training and testing process more difficult and can sometimes prevent accurate relationship classification. The quality differences are not recorded or labeled, resulting in inconsistencies that hinder optimal and functional results.

6 Acknowledgments

I would like to thank my lecturer in the course "Data Mining in Large Databases," for allowing and inspiring me to undertake this project and conduct the experiments presented.

Additionally, I am grateful to the Northeastern SMILE Lab for making the data publicly available on Kaggle at the following link: [Dataset Source](#)

Finally, I would like to note that ChatGPT was used for assisting with spelling and language checks throughout this project paper, as well as for speeding up some coding steps and reducing the need for extensive web searches.

References

- [1] Paola Bressan and Peter Kramer. Human kin detection. *Wiley Interdisciplinary Reviews: Cognitive Science*, 6(3):299–311, 2015.
- [2] Bruce S Weir, Amy D Anderson, and Amanda B Heppler. Genetic relatedness analysis: modern data and new challenges. *Nature Reviews Genetics*, 7(10):771–780, 2006.
- [3] Debra Lieberman, John Tooby, and Leda Cosmides. The architecture of human kin detection. *Nature*, 445(7129):727–731, 2007.
- [4] Guodong Guo and Xiaolong Wang. Kinship measurement on salient facial features. *IEEE Transactions on Instrumentation and Measurement*, 61(8):2322–2325, 2012.
- [5] Alice Othmani, Duqing Han, Xin Gao, Runpeng Ye, and Abdenour Hadid. Kinship recognition from faces using deep learning with imbalanced data. *Multimedia Tools and Applications*, 82(10):15859–15874, 2023.
- [6] Joseph P Robinson, Ming Shao, Yue Wu, Hongfu Liu, Timothy Gillis, and Yun Fu. Visual kinship recognition of families in the wild. *IEEE Transactions on pattern analysis and machine intelligence*, 40(11):2624–2637, 2018.