VU VRIJE UNIVERSITEIT AMSTERDAM | FACULTY OF SCIENCE

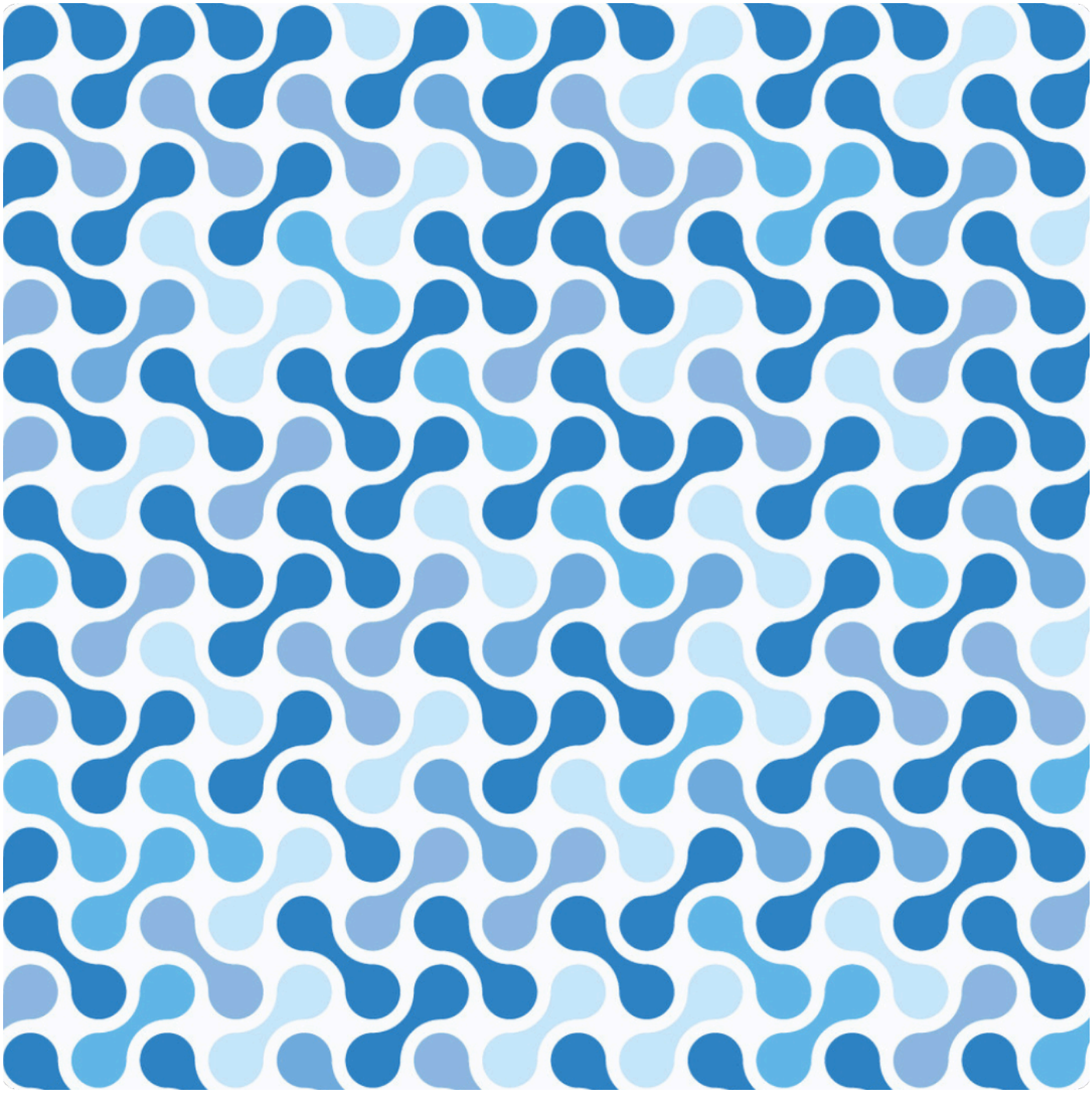
Kinship Recognition

Recognizing familial relationships through video clips

Yossi Hartman

Faculty of Science
Department of Mathematics

Master of Science (MSc) in Computational Intelligence — July 2025



External Supervisor:	Britt Van Leeuwen	Centrum Wiskunde & Informatica (CWI) Stochastics Group
First Reader:	Mark Hoogedorn	Vrije Universiteit Amsterdam Faculty of Science Department of Mathematics
Second Reader:	Rob van der Mei	Centrum Wiskunde & Informatica (CWI) Stochastics Group - Vrije Universiteit Amsterdam Faculty of Science Department of Mathematics

Abstract

Facial kinship recognition aims to determine whether two individuals are biologically related based on visual input, such as videos or images. While deep learning has advanced the field, progress is often constrained by limited labeled data and the challenge of extracting meaningful kinship cues from visual information. This thesis investigates how pre-trained models can be leveraged to improve kinship classification performance, with a focus on video-based inputs.

Two modeling strategies are evaluated: a Siamese-style baseline model (KinFusionNet) and a supervised contrastive learning approach (Family Label Aware Contrastive Learning, FLACL). Experiments are conducted using both video sequences and still image inputs. For video, we employ the MARLIN backbone, a masked autoencoder trained via self-supervised learning. For image-based experiments, we use ResNet50 trained with AdaFace, a quality-aware supervised method.

A central contribution of this work is the introduction of frame filtering strategies for video data, which selectively remove low-quality or poorly posed frames prior to training. This technique led to the largest single performance gain among all experiments. In particular, applying frame filtering to image-based KinFusionNet improved classification accuracy from 65.0% to 69.5%. The best overall result was achieved by the image-based FLACL model, which reached 70.1% accuracy.

Contrary to expectations, using full video sequences did not result in improved performance. These findings suggest that, under current model constraints, high-quality frames paired with robust pre-trained backbones offer a more effective path forward for kinship verification than leveraging raw temporal data. The results also underscore the importance of data quality, careful input selection, and the alignment between pretraining tasks and target objectives in facial kinship research.

List of Figures

List of Tables

Table of Content

1 Introduction

Determining kinship between individuals has traditionally relied on DNA testing. While this method provides the highest accuracy, its high costs in both acquisition and execution make it impractical for many situations. Moreover, DNA-based verification is not always feasible. In such cases, automatic visual kinship recognition offers a potential alternative, particularly in contexts where genetic testing is not an option. Visual kinship recognition which has been the subject of study across various disciplines including cognitive psychology and behavioral analysis. Research has demonstrated that individuals who share genetic relationships often exhibit similarities in facial features (Burch and Gallup, 2000; DeBruine *et al.*, 2009; Kaminski *et al.*, 2009). Inspired by those insights, Fang *et al.* (2010) introduced the first attempt at automatic kinship recognition from facial images.

Facial Kinship Recognition (FKR) is an automated process that determines biological familial relationships between individuals using facial images or videos. The problem is typically approached as a binary classification task, determining the presence or absence of a kinship relationship. FKR also encompasses the identification of specific relationship types, which can be categorized into: direct relationships (without intermediate family members) such as mother-daughter or father-son, and indirect relationships (involving intermediate connections) such as grandfather-granddaughter or uncle-nephew. Additionally, one could also clasify which family a pair of individuals belongs to (Fang *et al.*, 2013).

The field of FKR has experienced significant growth within computer vision over the past decade, driven by its diverse applications. In humanitarian contexts, FKR facilitates family reunification efforts for refugees displaced by conflicts or natural disasters (Robinson *et al.*, 2018). Law enforcement agencies utilize FKR in criminal investigations, as demonstrated during the Boston Marathon Bombing case (Klontz and Jain, 2013). The technology also assists in locating missing children and enhancing border control verification processes. Beyond security applications, FKR proves valuable in social media and personal data management, enabling efficient organization of family photo albums and large-scale image databases (Stone, Zickler and Darrell, 2010).

There remains significant challenges in creating reliable recognition systems. These challenges stem from two main sources: practical limitations in data collection and fundamental complexities in the nature of kinship recognition itself.

The data collection challenges appear in various forms of quality issues. Real-world facial images and videos often have inconsistent lighting, different camera angles, varying distances, and partial face obstructions. Secondly, most available datasets are limited in size and diversity and are often collected in controlled environments. These constraints increase the risk of overfitting and can hinder models from fully learning the subtle nuances of visual attributes in different environments.

At its core, FKR faces two main challenges: large intra-class variations and small inter-class variations (Wu *et al.*, 2022; Wang *et al.*, 2023). Intra-class variations arise from differences in sex, age, ethnicity, and expressions within kin pairs, while inter-class variations stem from unrelated individuals who may look alike and relatives who share only subtle features. The key challenge is identifying the most relevant facial landmarks to accurately assess similarity despite these overlaps.

Recent advances in transfer learning and fine tuning techniques have shown promise in addressing data limitations by using knowledge from related areas where more data is available (Zhao *et al.*, 2024). At the same time, self-supervised learning has seen remarkable success in computer vision. Modern techniques like SimCLR (T. Chen *et al.*, 2020), BYOL (Grill *et al.*, 2020), and MARLIN (Cai *et al.*, 2023) have shown impressive abilities to learn from images and videos without extensive labeled data. These approaches are on par and often beat traditional supervised learning approaches, including on benchmark datasets like ImageNet.

Despite significant advances in machine learning, the methods of self-supervised learning and transfer learning remains largely unexplored in kinship recognition. This research aims to bridge this gap by investigating how modern techniques can improve FKR systems, particularly when working with limited video data. Our main research question addresses a core problem in computer vision.

To what extent can pre-trained models improve the performance of video-based facial kinship recognition models when fine-tuned on limited data?

To answer this question, we apply two different knowledge transfer strategies:

The first strategy, KinFusionNet, is a Siamese-style model that processes a pair of facial images through a shared encoder. The resulting feature embeddings are fused and passed through a small MLP to predict whether the individuals are related. This lightweight architecture enables direct evaluation of the encoder's ability to capture kinship-relevant features.

The second strategy, Family-label Aware Contrastive Learning (FLACL), builds on contrastive learning principles inspired by SimCLR (T. Chen *et al.*, 2020). Rather than training a classifier, FLACL learns

to organize the feature space by pulling embeddings of related individuals closer together while pushing unrelated ones apart, using family labels as supervision. Kinship is then determined based on the similarity between embeddings, using a learned threshold.

To account for the limited size and quality of the dataset, we evaluate these models across three experimental setups. In the first experiment, clips are sampled from the videos without frame selection, allowing for maximal data usage. The second experiment introduces filtering by removing low-quality frames and those where the subject's pose obscures its visual attributes. Finally, in the third experiment, we extract only the best frame per video, aiming to determine whether a single, well-aligned image can outperform full video sequences.

The rest of the paper is structured as followed: the next sections first provide the necessary background and technical context. Section 2 reviews key research in visual representation learning, self-supervised learning, and facial kinship recognition. Next, Section 3 introduces the dataset and details the preprocessing pipeline. Afterwards, we will explain the training setup, backbones and the model architectures. Thereafter, in Section 4 we go over the experimental setup. The subsequent sections present the results, followed by a discussion of the findings and concluding remarks.

2
Literature Review

This chapter provides an overview of the most relevant literature for this research. It begins with the early work on facial kinship recognition and describes how the field has developed over time. After that, the focus shifts to visual representation learning, including methods such as transfer learning and self-supervised learning. These techniques are explained in more detail, showing how they help models learn useful features from visual data, especially when there is little labeled data available.

2.1 Kinship Recognition

2.1.1 Human Visual Kinship Recognition

Before the development of automated systems, visual kinship recognition has been widely studied across disciplines including genalogy, neuroscience, psychology and behavioral analsysis (Burch and Gallup, 2000; Dal Martello and Maloney, 2006; Zebrowitz and Montepare, 2008; DeBruine *et al.*, 2009; Kaminski *et al.*, 2009). Research focused on understanding how humans recognize family relationships through facial features. Dal Martello and Maloney (2006) conducted a comprehensive study with 220 participants to identify which facial regions are most crucial for kinship recognition. By systematically masking different facial areas, they found that the upper face region slightly improved recognition accuracy, while surprisingly, masking the mouth region led to the highest accuracy. However, these findings were later challenged by Gao *et al.* (2019), who demonstrated that mouth regions play a key role in identifying parent-child relationships. Adding to this complexity, DeBruine *et al.* (2009) found that facial similarity patterns vary significantly between genders.

A comprehensive study by Bordallo Lopez *et al.* (2018) involving 304 participants provided deeper insights into the challenges shared by both human and machine recognition systems. Their research revealed that age differences significantly impact recognition accuracy, as facial features undergo substantial changes throughout life. Additionally, cross-gender kinship pairs proved more challenging to identify, with both humans and machines performing best at recognizing same-gender sibling relationships compared to parent-child pairs.

2.1.2 Background

The first proposals for automated FKR relied exclusively on handcrafted feature descriptors, extracting facial attributes such as color and geometric distances between landmarks (Fang *et al.*, 2010; Guo and Wang, 2012; Xia, Shao and Fu, 2012). These extracted features served as inputs for traditional machine learning models like k-Nearest Neighbors (KNN) and Support Vector Machines (SVM). Subsequently, researchers enhanced performance by incorporating additional handcrafted feature descriptors from image processing and texture analysis. (Zhou *et al.*, 2011; Kostinger *et al.*, 2012; Lu *et al.*, 2014; Yan *et al.*, 2014; Yan, Lu and Zhou, 2015).

Following this, metric learning approaches emerged as an alternative to traditional classifiers (Xia, Shao and Fu, 2010; Shao, Xia and Fu, 2011; Xia *et al.*, 2012; Lu *et al.*, 2014; Yan *et al.*, 2014; Yan, Lu and Zhou, 2015). Instead of directly predicting labels, metric learning trains models to measure the similarity between samples by learning a distance function. The goal is to pull related pairs closer in the learned feature space while pushing unrelated pairs further apart, improving the model's ability to distinguish meaningful relationships.

At the same time, Dibeklioglu, Salah and Gevers (2013) introduced the first video-based FKR solution, relying solely on handcrafted features. In addition to the previously mentioned descriptors, they manually tracked facial landmark displacements to capture temporal motion. Their findings demonstrated that incorporating spatio-temporal features from short video sequences emhances FKR performance.

The shift toward deep learning in FKR began with Zhang *et al.* (2015), introducing the first CNN-based approach. This model also improved the previous score by 10%. This marked a turning point, leading to increasingly sophisticated architectures such as Siamese CNN networks (Li *et al.*, 2016), Graph Neural Networks (Liang *et al.*, 2019; Li *et al.*, 2021), Attention Mechanisms (Yan and Wang, 2019), and GANs (Wang, Ding and Fu, 2019; L. Zhang *et al.*, 2021; Liu *et al.*, 2022). As a result, recent research predominantly relies on deep learning and learned feature representations. Even within metric learning, researchers now integrate handcrafted and deep learning features, and shown promissing results as shown in Table 2.1 (Laiadi *et al.*, 2020; Serraoui *et al.*, 2022; Wang, Chen and Hu, 2022; Belabbaci *et al.*, 2023; Ramazankhani, Yazdian-Dehkord and Rezaeian, 2023).

Year	Model	Approach	Accuracy
2014	NRML (Lu *et al.*, 2014)	Metric + shallow	76.5%
2014	DMML (Yan *et al.*, 2014)	Metric + shallow	78.25%
2015	CNN-points (Zhang *et al.*, 2015)	Deep	88.4%
2019	Attention Network (Yan and Wang, 2019)	Deep	92%
2020	Graph Network (Liang *et al.*, 2019)	Deep	90.6%
2021	Relational Network (Yan and Song, 2021)	Deep + Metric	88.8%
2021	AdvKin (L. Zhang *et al.*, 2021)	Deep + Metric	88.8%
2022	TXQEDA+WCCN (Serraoui *et al.*, 2022)	Deep + Metric	90.3%

Table 2.1 — Overview of the performance between prior and recent work. The scores are based on the KinFaceW-II dataset.

This paradigm shift also extended to video-based FKR. Inspired by Dibekliŏglu, Salah and Gevers (2012), Boutellaa *et al.* (2017) leveraged deep feature extraction with the VGG-Face model (Parkhi, Vedaldi and Zisserman, 2015), significantly improving accuracy to 90% and surpassing the score 67% of Dibekliŏglu, Salah and Gevers (2012). More recently, Kohli *et al.* (2019) achieved state-of-the-art performance using a supervised mixed-norm autoencoder architecture, further advancing video-based kinship recognition.

The most recent developments explore multimodal approaches by combining audio with visual data. Three datasets support this direction: Families in the Wild with Multimedia (FIW-MM), KAN-AV, and TALKIN-Family (Robinson *et al.*, 2022; Kefalas *et al.*, 2023; Wu *et al.*, 2024).

2.2 Visual Representation Learning

In computer vision, models rarely process raw image data directly. Instead, they rely on learned visual representations, compact, informative features that capture relevant patterns from visual input. This process, known as visual representation learning, is a fundamental step in enabling machines to understand, compare, and classify images or videos.

In the context of facial kinship recognition, learning effective representations is especially challenging due to the subtle and often non-obvious visual similarities between family members (Wang *et al.*, 2023). These representations must be robust to changes in lighting, pose, expression, and age, while still capturing the nuanced features that may indicate a biological relationship.

As described in Section 2.1.2, in early work, visual representations were hand-crafted using descriptors like HOG (Histogram of Oriented Gradients) or LBP (Local Binary Patterns) (Kostinger *et al.*, 2012; Lu *et al.*, 2014; Yan, Lu and Zhou, 2015). However, the recent models has largely replaced these manual methods, enabling deep learning to learn features automatically through data-driven training.

In general most deep learning approaches rely on supervised learning, where models were trained from scratch on large labeled datasets to perform specific tasks, aiming to generalize well to unseen data. However, achieving satisfactory performance often requires millions of training samples. Insufficient training data can lead to poor prediction performance, limited generalization capability, and rapid overfitting.

Among the various approaches to address deep learning's data scarcity problem, pre-training has emerged as a central technique. This method focuses on learning effective representations from large source datasets and transferring this knowledge to target domains, accelerating learning in new tasks (Zhao *et al.*, 2024).

Transfer Learning (TL), also known as knowledge transfer, involves training a baseline model on a large dataset before fine-tuning it on a smaller target dataset to achieve optimal performance (Iman, Arabnia and Rasheed, 2023; Zhao *et al.*, 2024). This approach draws inspiration from human learning patterns, where previously acquired knowledge can be applied to learn new skills more efficiently (Pan and Yang, 2010). For instance, someone who already plays a musical instrument typically learns another instrument more quickly than a complete beginner.

Transfer learning can be broadly categorized into transductive, inductive, and unsupervised learning, depending on how the source and target domains and tasks are related (Pan and Yang, 2010; Iman, Arabnia and Rasheed, 2023).

In transductive transfer learning, the source and target tasks are the same, but the data distributions differ. A common example is a face recognition model trained on high-quality studio portraits being deployed on surveillance footage, where differences in lighting, resolution, and noise introduce a domain shift. While labeled data exists in the source domain, the target domain remains unlabeled.

Inductive transfer learning, on the other hand, involves a change in task between source and target domains. In this setting, the target domain includes labeled data, which allows the model to fine-tune its parameters for the new task. A typical example would be fine-tuning a model trained for image classification on ImageNet for a different task, such as action recognition in video clips. Despite the shared visual domain, the objective changes, from identifying static objects to interpreting dynamic motion.

A comprehensive study by Mensink *et al.* (2021) examined the impact of transfer learning through 1,200 experiments across 20 datasets. These spanned a wide range of image domains, including consumer, driving, aerial, underwater, indoor, synthetic, and close-ups, and covered task types such as semantic segmentation, object detection, depth estimation, and keypoint detection. The study yielded several important findings.

First, models using transfer learning consistently outperformed those trained from scratch. Second, the alignment between source and target domains (i.e., within-domain transfer) was more critical to performance than the sheer size of the pre-training dataset. In other words, domain similarity matters more than dataset scale. Third, while multi-source models, pre-trained on data from several domains, performed reasonably well, they did not surpass the effectiveness of large, well-aligned single-source models. Additionally, transfer learning was shown to be especially beneficial for small target datasets, where it enabled more accurate model selection even under limited data conditions.

These insights highlight the importance of choosing both the pre-training method and the pre-training data carefully. For our application in facial kinship recognition, this means selecting models trained on facial data that is both representative of the target domain and diverse in content, in order to maximize generalization.

In kinship recognition, transfer learning has become a standard practice since the early application of Zhang *et al.* (2015). Today, most deep learning-based kinship studies begin with a pre-trained model as their foundation. Where the majority use these pre-trained models purely as feature extractors. Among the architectures used, VGG has been particularly prominent Parkhi, Vedaldi and Zisserman (2015), with multiple studies demonstrating its effectiveness in kinship verification tasks as shown in Table 2.1. (Dornaika, Arganda-Carreras and Serradilla, 2020; Laiadi *et al.*, 2020; L. Zhang *et al.*, 2021; Liu *et al.*, 2022; Serraoui *et al.*, 2022; Li and Jiang, 2023).

These backbones are typically pre-trained on large-scale face recognition datasets such as VGGFace2 (Cao *et al.*, 2018), CASIA-WebFace (Yi *et al.*, 2014), and LFW (Huang *et al.*, 2008). Studies have consistently shown that the representations learned through training on such datasets can be effectively reused for kinship verification.

Boutellaa *et al.* (2017) was among the first to apply VGG as a feature extractor for kinship recognition. Their video-based approach extracted feature vectors from VGG for each frame and averaged these representations over time before comparing them between individuals. This method significantly improved accuracy to 90%, surpassing previous solutions. Expanding on this, Serraoui *et al.* (2022) introduced multi-source transfer learning, combining feature representations from VGG-F, VGG-M, and VGG-S to achieve performance comparable to state-of-the-art methods. Meanwhile, Kohli *et al.* (2017) pre-trained their own backbone on specific facial regions using Stacked Fully Connected Deep Belief Networks. These learned representations were then fine-tuned, leading to further advancements in kinship recognition.

The limitations of transfer learning can be grouped into three main categories: data scarcity, label shift, and incorrect model focus (Zhao *et al.*, 2024). Data scarcity is relatively straightforward, it remains difficult to acquire large, high-quality datasets, even for pre-training. The other two limitations are more subtle but equally important. Label shift occurs when there is a significant mismatch

between the labels used during pre-training and those in the fine-tuning task, which can lead to negative transfer. Incorrect model focus refers to situations where the model learns to rely on irrelevant features instead of the intended targets. For instance, in the FIW dataset (Robinson *et al.*, 2018), faces are extracted from family photographs. As shown by (Leeuwen *et al.*, 2022), models trained on this data sometimes rely on background similarities or contextual cues, like lighting, rather than facial features themselves, resulting in unreliable predictions.

2.3 Self-Supervised Visual Representation Learning

Self-supervised learning (SSL) offers a promising direction for advancing machine learning beyond the limitations imposed by labeled data. Traditional supervised learning methods rely heavily on large-scale, annotated datasets, which can be costly, time-consuming, or even infeasible to obtain. In contrast, SSL leverages the structure and redundancy inherent in the data itself to learn useful representations without relying on manual labeling (Zhao *et al.*, 2024). This is particularly impactful in computer vision, where raw image and video data is abundant but labels are scarce. SSL addresses this challenge by using various data augmentation techniques, such as random cropping, rotation, color jittering, and blurring, to create alternate "views" of the same data. The model is then trained to solve an auxiliary task, using these augmented views as a form of synthetic supervision. These tasks, known as pretext tasks, guide the model to learn general features that capture the underlying structure of the data. As a result, SSL enables models to extract meaningful visual patterns and representations, making it a valuable tool in settings where labeled data is limited or unavailable (Ericsson *et al.*, 2022).

Much like transfer learning, SSL begins with a pre-training phase where the model solves a pretext task to learn general, transferable features. The design of this pretext task is crucial, as it guides the type of information the model will prioritize and encode. The type of pretext task is usually categorised into three groups: contrastive, predictive, and generative learning (Zhao *et al.*, 2024). Common examples of pretext tasks in computer vision include predicting the original order of shuffled frames, identifying whether two augmented views come from the same image, or reconstructing masked parts of an input image or video, as done in masked autoencoding approaches. The choice of pretext task directly influences the quality of the learned features and determines how well the model will perform when fine-tuned on downstream tasks like kinship verification (Wu *et al.*, 2023).

Contrastive learning aims to learn representations by maximizing the similarity between positive pairs (samples that share semantic similarity) while minimizing the similarity with negative pairs (samples that do not). In practice, this creates a structured feature space where similar examples are pulled closer together and dissimilar ones are pushed apart. Many state-of-the-art self-supervised

learning models are built on the principles of contrastive learning, though they differ in how they implement and overcome specific limitations of the framework.

SimCLR (T. Chen *et al.*, 2020) is one of the most prominent contrastive learning methods. It maximizes agreement between differently augmented views of the same image using a contrastive loss. This is achieved through strong data augmentations, such as random cropping and color distortion, followed by passing the representations through a projection head into a latent space where the loss is applied. However, a key limitation of SimCLR is its reliance on large batch sizes to provide a sufficient number of negative samples within each batch.

MoCo (He *et al.*, 2020) addresses this limitation by introducing a momentum encoder and a queue-based memory bank for negative sampling. Instead of requiring large batches, MoCo maintains a dynamic queue of previously encoded representations, allowing more stable and efficient contrastive learning across batches. Its successor, MoCo-v2 (X. Chen *et al.*, 2020), incorporates SimCLR's improved augmentation pipeline and a projection head, resulting in stronger performance.

Beyond contrastive methods, another major category within SSL is generative learning, where the goal is to reconstruct missing or corrupted parts of the input. This strategy forces the model to learn meaningful global representations by capturing structural dependencies in the data (Gao and Patras, 2024). One of the most well-known examples of this is masked modeling, which originated in natural language processing with models like BERT (Devlin *et al.*, 2019). In this framework, parts of the input are deliberately masked, and the model learns to predict the missing content using context.

A notable example of masked modeling applied to video is the MARLIN model (Cai *et al.*, 2023). MARLIN uses a Masked Autoencoder (MAE) framework to reconstruct masked regions in facial video frames. The model consists of an encoder, which processes visible portions of the input, and a decoder, which reconstructs the masked areas. By hiding a significant portion of the input during training, the model is forced to learn abstract, high-level features instead of relying on low-level cues. MARLIN has achieved strong results across multiple facial analysis tasks, including attribute recognition, expression classification, and lip-sync generation, making it a strong candidate for transfer learning in kinship recognition.

To date, self-supervised pre-training has not, to our knowledge, been applied to facial kinship recognition. However, a core principle of SSL, contrastive learning, has appeared in related studies. Early work by Li *et al.* (2016) applied a Siamese-style network to kinship recognition. Their model was trained on positive and negative image pairs, with the objective of maximizing similarity for related pairs and minimizing it for unrelated ones. Kinship was then determined using a predefined similarity threshold. Building on this approach, Dibeklioglu (2017) developed a video-based method that introduced a Siamese-like autoencoder with cross-weight sharing. Their model was trained to

learn transformations between parent-child pairs, effectively modeling the facial transition from one individual to the other. This transformation learning served as a form of pretraining. After the decoder was removed, a similarity-based classifier, similar to that used by Li *et al.* (2016) , was employed to determine kinship. To further improve performance, their method also included expression alignment, ensuring that both individuals in a pair shared the same facial expression (e.g., smiling) during comparison.

Zhao *et al.* (2024) identifies four key limitations of SSL: weak supervision signals, limited pretext tasks, high computational costs, and potential label inaccuracies. In computer vision, data augmentation helps mitigate the first issue, but video-based SSL still struggles with a lack of diverse pretext tasks. A major challenge is SSL's reliance on pseudo-labels, which act as a bridge between weak and strong supervision. Without sufficient manual supervision, inappropriate signals may degrade model performance. Additionally, SSL requires significant computational resources, making it less accessible for resource-constrained environments. Lastly, pseudo-label accuracy depends on proper loss function selection and augmentation strategies, errors in these areas can negatively impact learning, posing challenges for effective SSL implementation.

3
Methodology

3.1 Problem Description & Formulation

Facial Kinship Recognition refers to the automated process of determining whether two or more individuals, represented by facial images or videos, share a biological familial relationship. In this thesis, we will consider the classification problem of determining kinship, as well as the type of relation. Typically, a sample can be formulated as:

$$\mathrm{X} = \left\{ x_i \in \mathbb{R}^{f \times c \times h \times w} \mid i = 1, 2, ..., N \right\} \tag{3.1}$$

Where i refer to the i-th sample in our set N. f denotes the number of frames, with $f = 1$ for images and $f > 1$ for videos. w and h represent the image width and height in pixels, respectively. Lastly, c refers to the number of color channels, typically $c = 3$ for RGB. Following the notation from Wu *et al.* (2022), the binary classification problem can be formulated as:

$$z = f\big(g(x_i), g(x_j)\big)) \quad \mathrm{z} \in \{0, 1\} \quad \text{for } i \neq j \tag{3.2}$$

Where, $g(\cdot)$ is the encoder, responsible for mapping input images or videos to a feature space and $f(\cdot)$ is a classifier, which predicts whether a kinship relation exists ($z = 1$) or not ($z = 0$).

3.2 Dataset & Preparation

3.2.1 KAN-AV

For this research, we utilize the audio-visual KAN-AV dataset (Kefalas *et al.*, 2023), which comprises audio and video recordings extracted from publicly available YouTube videos captured in unconstrained environments. The the dataset features 970 persons of interest from diverse backgrounds, including actors, TV presenters, musicians, politicians, and athletes. Each individual is represented across multiple videos, categorized according to their age at the time of recording. A small subset

of these individuals is illustrated in Figure 3.1. The complete dataset encompasses 26,112 samples, with 93.2% of subjects falling within the 20–80 age range. A overview of the distribution is illustrated in Figure 3.2

Figure 3.1 — A subset of the KAN-AV dataset demonstrating the natural variation in lighting conditions, camera angles, facial occlusions, and subject distance typical of unconstrained environments.

Beyond identity and age annotations, the dataset documents familial connections for 645 individuals, each having at least one kinship relation. Specifically, it contains 532 pair-wise, first-degree kinship relations distributed across seven relationship types: Brother-Brother (B-B), Brother-Sister (B-S), Sister-Sister (S-S), Father-Son (F-S), Father-Daughter (F-D), Mother-Son (M-S), and Mother-Daughter (M-D).

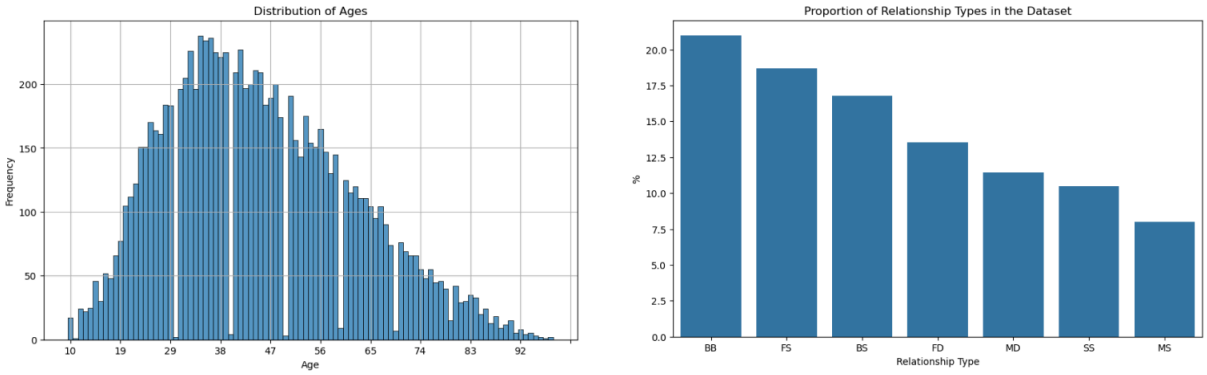


Figure 3.2 — illustration of the distribution of ages and the relationship types in the dataset

3.2.2 Preprocessing

The preprocessing pipeline begins by extracting and cropping each individual's face to match the input size expected by the model backbone. Each video sample is accompanied by a bounding box file, generated using a YOLOv3-based face detector (Redmon and Farhadi, 2018), and manually verified by the authors of the dataset (Kefalas *et al.*, 2023). This bounding box provides the initial region of interest for each face, as shown in Figure 3.3.

Figure 3.3 — The detected face and bounding box from the face detection model by (Redmon and Farhadi, 2018)

The detected boxes are typically tight and often exclude parts of the forehead, hair, ears, or neck, regions that can be informative for kinship verification. To preserve these facial cues, a margin is added around each bounding box before cropping. The final extracted region is shown in Figure 3.4.

Figure 3.4 — The extracted area, including the added margin.

Additionally, to reduce variation caused by head tilt, each face is aligned so that the eyes are horizontally leveled. This normalization removes the roll component of head rotation while keeping all other facial attributes intact. Although aligning faces may slightly reduce the natural appearance of head movements, this step does not alter or remove any facial features themselves. Since our task focuses on facial appearance rather than motion dynamics, standardizing orientation helps the model focus on consistent structural features that are relevant for kinship recognition. In this context, removing rotational variation is a practical way to reduce noise without losing meaningful information.

3.2.3 Data splitting

To build a balanced and representative dataset for classification, we followed a structured pairing strategy. The first step involved constructing a family tree to clearly map the relationships between individuals. This was crucial, as a single person can occupy multiple roles within a family, for example, being both a father and a son. As a result, we were able to extract family IDs, which can be utilized during training.

Out of the many possible strategies for creating kinship sample pairs, we followed the approach used by (Kefalas *et al.*, 2023). This choice allows for a more direct and fair comparison of results. In this method, the algorithm loops through all annotated relationships in the dataset and randomly links one video sample to another within the same relationship type. Once a pair is formed, both samples are excluded from further pairing. This ensures that no sample appears in more than one pair, making it easier to create clean training, validation, and test splits without duplicated samples. While this strategy helps reduce data leakage, it does not eliminate it completely, some individuals may still appear in both training and test sets, though at different ages or in different videos.

For negative pairs, where individuals have no known familial connection, we randomly sampled from unrelated individuals within each split. We ensured that the number of negative samples matched the number of positive ones, resulting in a completely balanced dataset across all subsets.

3.3 Training Paradigm

3.3.1 Pre-training dataset

As described in Section 2.2, Mensink *et al.* (2021) findings emphasize that selecting a pre-trained backbone closely aligned with the target task is critical for effective transfer. This not only applies to the pretext task used during pre-training but also to the nature and composition of the dataset on which the backbone was trained. In facial kinship recognition, where subtle visual cues are essential, both dataset quality and diversity of facial attributes significantly influence transfer effectiveness.

A variety of image facial recognition datasets have been developed over the years, each differing in scale, source, and visual diversity. Among the most commonly used are CASIA-WebFace (Yi *et al.*, 2014), LFW (Huang *et al.*, 2008), VGGFace2 (Cao *et al.*, 2018), and MS1MV2 (Deng *et al.*, 2022). These datasets are composed of images scraped from platforms such as YouTube, Google Search, and academic datasets, and vary widely in terms of resolution, number of identities, and pose diversity. Table 3.2 provides an comparison overview of the datasets.

For the video-based model, we use MARLIN, which is described in detail in Section 3.4.1. MARLIN is pre-trained on the YouTube Faces Database (YTF) (Lior, Tal and Itay, 2011), a dataset comprising short video clips of celebrities collected from YouTube. The dataset is relatively small but offers dynamic facial variations and real-world noise, making it suitable for temporal modeling in video.

For the image-based model, we selected a backbone (ResNet) pre-trained on MS1MV1, a large-scale face recognition dataset. MS1MV1 offers substantial variability in identity, pose, age, and lighting conditions, which makes it highly suitable for learning generalized facial representations. Its size and diversity make it particularly well-suited for transfer to kinship verification tasks, where subtle facial similarities must be captured across varying conditions.

author	Dataset	Type	Size	Identities	Variability	Quality	Source
(Yi *et al.*, 2014)	CASIA-WebFace	Image	500K	10.5K	Medium	Medium	News Articles
(Huang *et al.*, 2008)	LFW	Image	13K,	5.7K	Medium	Medium	IMDb
(Cao *et al.*, 2018)	VGGFace2	Image	3.3M	9.1K	High	High	Google Image Search
(Deng *et al.*, 2022)	MS1MV1	Image	5.8M	85K	Very High	High	Bing
(Lior, Tal and Itay, 2011)	YTF	Video	3.4K	1.6K	High	Medium	YouTube

Table 3.2 — Overview of types of available facial datasets

3.3.2 Pretext task

The backbones used in this study are pre-trained using different approaches, each tailored to their respective data modalities (video vs. image). These training methods influence how the models learn

to extract useful visual features. The choice of pretext task was not based on a deliberate comparison between learning strategies but instead followed the standard design of each backbone as proposed in their original studies.

The MARLIN backbone (used for video inputs) is pre-trained using a self-supervised generative learning approach. In this method, large portions of each input frame are masked, and the model learns to reconstruct the missing regions based on the visible parts. This task encourages the encoder to learn high-level representations that capture both spatial structure and temporal facial dynamics, without relying on identity labels. The goal is to understand facial content across time using unlabeled video data. A simplified overview of this process is shown in Figure 3.5.

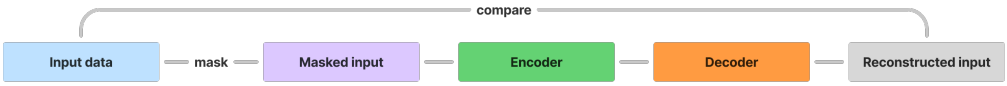


Figure 3.5 — High-level overview of the self-supervised masked modeling strategy used in MARLIN

In contrast, the image-based backbone (ResNet50) is pre-trained using a supervised learning approach. The model is trained to classify each image based on the person's identity. This approach teaches the network to focus on facial features that help differentiate between individuals. Although effective for tasks like face recognition, it relies on labeled datasets and can inherit dataset-specific biases. A simplified illustration of this supervised training process is shown in Figure 3.6.

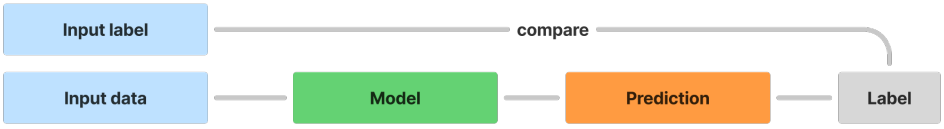


Figure 3.6 — High-level overview of the supervised training process used in ResNet50

3.3.3 Knowledge transfer

In this research, we adopt an inductive transfer learning approach. The source task face regonition (ResNet50), face reconstruction (MARLIN), differs from our target task of classifying kinship relationships between individuals. While both tasks operate within the visual domain and rely on facial features, their objectives are not the same. Identity recognition focuses on distinguishing individuals, whereas kinship recognition involves detecting subtle traits shared between related individuals, such as facial proportions, bone structure, or eye spacing.

Within the inductive transfer learning framework, several transfer strategies can be applied depending on what is being transferred from the source to the target task (Pan and Yang, 2010). These include instance-based transfer, feature representation transfer, parameter transfer, and relational transfer. In this work, we focus on feature representation transfer, where a pre-trained model is used to extract general-purpose features from input data. These features are expected to capture useful patterns that can be reused for the target task.

3.3.4 Fine-tuning

The effectiveness of transfer learning depends not only on what is transferred, but also on how the transfer process is applied to the new task. In this research, we explore two fine-tuning strategies tailored to different model architectures and training objectives.

In the first strategy, the backbone is kept completely frozen, and only new layers are added on top. These layers are trained to classify whether two feature embeddings correspond to a related pair. This method allows us to assess the strength of the pre-trained features on their own, without altering the internal representations learned during pretraining. It serves as a lightweight and interpretable approach, particularly useful when the target dataset is small or the risk of overfitting is high.

The second strategy involves gradual fine-tuning of the backbone itself. Here, we begin by unfreezing the final layers of the encoder and training them using a contrastive learning objective. This setup encourages the model to adjust its representations in a way that better reflects familial similarity, rather than identity. By allowing parts of the backbone to adapt, the model can fine-tune its focus from identity-specific traits to more general patterns shared among related individuals.

3.4 Backbones

3.4.1 MARLIN

A suitable candidate for this research is MARLIN (Cai *et al.*, 2023), which stands for Masked Autoencoder for facial video Representation LearnINg. MARLIN employs a masked autoencoding

framework, where parts of the input data are deliberately masked, requiring the model to reconstruct the missing regions using only the visible portions.

MARLIN is trained in a fully self-supervised manner, meaning it does not require any information about the individuals in the training data, such as age, ethnicity, or identity. As a result, it becomes possible to train on much larger and more diverse datasets, since additional video material can be included without the need for time-consuming and costly manual labeling.

The model builds on VideoMAE (Tong *et al.*, 2022), a masked autoencoder architecture designed to learn compact and generalizable facial representations from videos. One of its key innovations is the introduction of Facial Region-Guided Tube Masking (Fasking), a masking strategy that uses a pre-trained face parser to identify and prioritize semantically important facial regions, such as the eyes, nose, and mouth. Let $x \in \mathbb{R}^{\{T \times H \times W \times C\}}$ represent a video clip consisting of T frames. The model randomly masks approximately 90% of the spatiotemporal patches in x, but with priority given to the previously parsed key regions. Unlike standard frame-wise masking, the same facial region is masked across all T frames (i.e., a temporal tube), which prevents the model from reconstructing a missing region in one frame by relying on visible information in neighboring frames.

The unmasked patches are then passed through the encoder to produce a latent representation z, which is fed into a lightweight decoder. The decoder attempts to reconstruct the full input $\hat{x}$, including the masked areas. This reconstruction task forces the encoder to learn high-level spatiotemporal features that capture the appearance and structure of facial components over time. As shown in Figure 3.7, this process enables the model to focus on meaningful facial traits that are stable across frames, features that are particularly important for kinship verification. Prior research has shown that these specific facial regions play a significant role in human kinship perception (Dal Martello and Maloney, 2006; Kaminski *et al.*, 2009; DeBruine *et al.*, 2009).

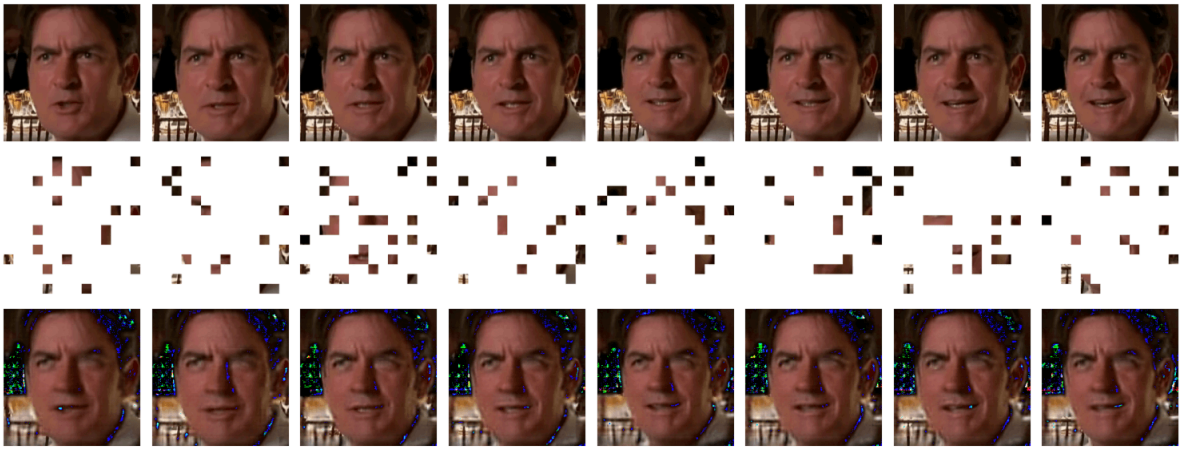


Figure 3.7 — Visualization of the MARLIN model's video reconstruction process. The first row shows the ground truth. The second row depicts the input to the model, where 90% of the data is masked, leaving only 10% visible. The final row presents the MARLIN model's reconstruction of the frames.

At inference time, the encoder processes unmasked video frames, allowing it to leverage complete visual information for generating feature representations. For the purposes of this research, we are only interested in these feature representations. Therefore, we remove the decoder and use only the pre-trained encoder as a feature extractor.

3.4.2 ResNet

We also require a backbone that processes individual images. A well-established architecture for image-based tasks is ResNet (Residual Network) (He *et al.*, 2015).

ResNet introduces the concept of residual connections, which help the model train deeper networks more effectively. In a regular neural network, each layer learns a function $H(x)$ that maps an input x to an output. In ResNet, the network instead learns a residual function, which means it learns the difference between the input and the output, written as $F(x) = H(x) - x$. This can be rearranged to $H(x) = F(x) + x$, where x is directly added back to the output of the layer. See Figure 3.8

This shortcut connection, adding the input x back to the output, makes it easier for the network to learn changes or small adjustments, rather than having to learn everything from scratch. As a result, it becomes possible to train very deep networks without running into common problems like vanishing gradients, where the learning signal gets too small to update earlier layers. These residual connections help the model remain stable and efficient during training.

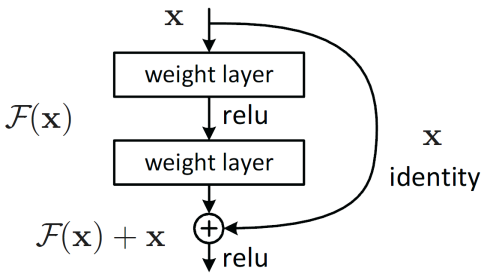


Figure 3.8 — Diagram of the residual connection

In our setup, ResNet is pre-trained using AdaFace (Kim, Jain and Liu, 2022), a method that enhances face recognition performance by incorporating image quality awareness into the training process. While traditional margin-based losses such as CosFace (Wang *et al.*, 2018) and ArcFace (Deng *et al.*, 2022) enforce angular separation between features, they apply equal emphasis to all training samples regardless of quality.

AdaFace addresses this limitation by adjusting the learning signal based on image quality, estimated by the l_2 of the embedding $\|f(x)\|$. High-quality samples (with larger norms) receive stronger supervision via a larger adaptive margin, while low-quality images (smaller norms) are updated more conservatively. This stabilizes training and prevents noisy or blurry samples from dominating the learning process.

Because AdaFace trains the model to focus on high-confidence visual signals, the resulting representations are less sensitive to distortions and noise. This makes the model more robust at inference time, especially when dealing with low-quality inputs, since it has not overfitted to unreliable training examples.

3.5 Model Design

3.5.1 KinFusionNet

For the first model, we adopt a Siamese-style network. The primary goal of this model is to assess how well learned facial representations, extracted from a backbone encoder, can be leveraged to determine the presence or absence of kinship. The core idea is to process two input samples independently through a shared feature extractor, then combine the resulting embeddings and use a shallow classifier to predict kinship. The flow of this network is visualized in Figure 3.9.

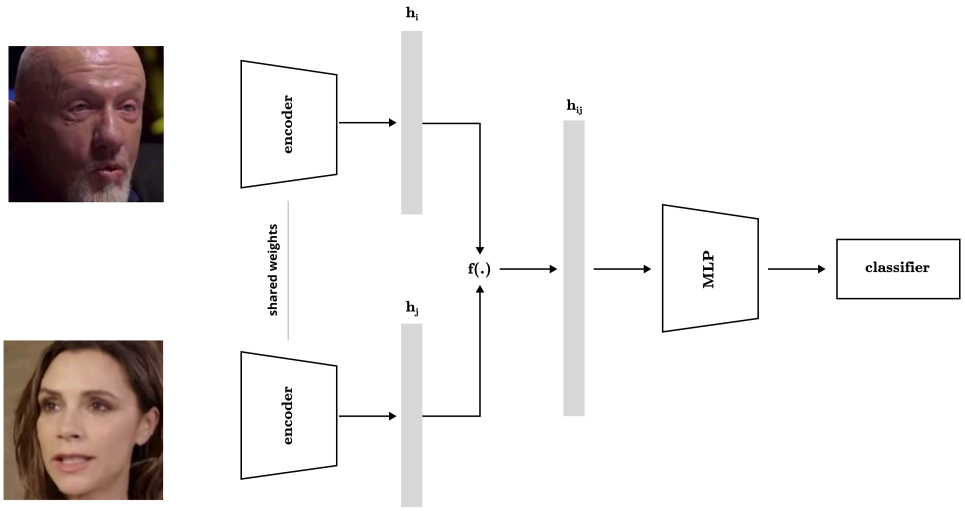


Figure 3.9 — A diagram that illustrates the architecture of the siamese network

The model receives as input a pair of facial samples, each corresponding to a different individual in a candidate kinship pair. These inputs are independently passed through a shared encoder which returns a representation for each individual (h_i and h_j). By sharing weights across both branches, the backbone ensures consistent feature extraction, allowing both inputs to be embedded in the same representational space.

Next, the two feature vectors are combined into a single joint representation, h_{ij}. Previous research by Li and Jiang (2023) systematically evaluated various fusion strategies for kinship verification, including element-wise addition $(x + y)$, subtraction $(x - y)$, dot product $(x \cdot y)$, and concatenation. Their findings suggest that combining multiple operations leads to stronger performance than using any single method in isolation. Based on this, we concatenate several fusion variants into a single composite vector that reflects both shared and contrasting facial features between the input pair.

This fused representation is then passed to a Multi-Layer Perceptron (MLP), which serves as the classifier. The MLP consists of a small number of fully connected layers with non-linear activations (e.g., ReLU), and maps the input to a scalar output representing the predicted probability of kinship. Importantly, the backbone remains frozen during training; only the MLP is fine-tuned on the kinship task. This allows us to directly assess how well the pre-trained features support kinship classification, without modifying the original encoder.

The model is trained using Binary Cross-Entropy (BCE) loss, which is appropriate for binary classification tasks and provides a clear objective for learning discriminative patterns. This network is intentionally kept simple to isolate the contribution of the learned representations from the backbone.

3.5.2 Family Label Aware Contrastive Learning

The second model architecture explored in this research builds on a more sophisticated learning strategy which is largely inspired by the SimCLR (T. Chen *et al.*, 2020) network. Unlike KinFusionNet, which relies on a simple classifier to evaluate fused feature vectors, this approach focuses on refining the encoder itself using a contrastive learning objective. The main goal is to improve the quality of the learned feature space by encouraging the model to pull together embeddings from similar (positive) pairs and push apart embeddings from dissimilar (negative) pairs. Over time, this causes samples with similar semantics to form tight clusters in the high-dimensional representation space, improving downstream classification performance.

Once the encoder has been trained using this contrastive objective, kinship is not predicted through a learned classifier but rather through a similarity metric (e.g., cosine similarity or Euclidean distance) applied to the resulting embeddings. A predefined threshold is then used to determine whether a given pair exhibits kinship or not. This setup shifts the focus away from explicit supervision and instead prioritizes learning a structured feature space where kin-related samples naturally cluster together.

3.5.2.1 SimCLR

To better understand our adaptation, it is useful to first look at the original SimCLR architecture. SimCLR (Simple Framework for Contrastive Learning of Visual Representations) is a self-supervised learning method that aims to learn high-quality feature representations without relying on manual labels. Introduced by T. Chen *et al.* (2020), SimCLR leverages contrastive learning to train a neural network to distinguish between similar and dissimilar data points by comparing their representations in a learned embedding space.

At the core of SimCLR is a simple yet powerful idea: given an input image, generate two different augmented views and treat them as a positive pair, while treating all other augmented images in the batch as negatives. This forces the model to learn representations that are invariant to transformations while still being discriminative across different images.

The SimCLR architecture is composed of three main components that work together to facilitate contrastive learning. First, each image in a batch is augmented twice using a series of random transformations T, such as cropping, color jittering, and flipping. These two augmented views form a positive pair, while all other images in the batch act as negative examples. Each view is then passed through a shared encoder network which extracts a high-dimensional feature representation (h_j & h_i). These features are subsequently processed by a projection head, a small multilayer perceptron (MLP) that maps them into a latent space specifically optimized for contrastive learning (z_j & z_i). The role of the projection head allows the encoder to focus on learning general-purpose visual features,

while the projection head learns to select a specific "subspace" of the features that most effectively minimises the contrastive loss (Gupta *et al.*, 2022). Figure 3.10 illustrates an example.

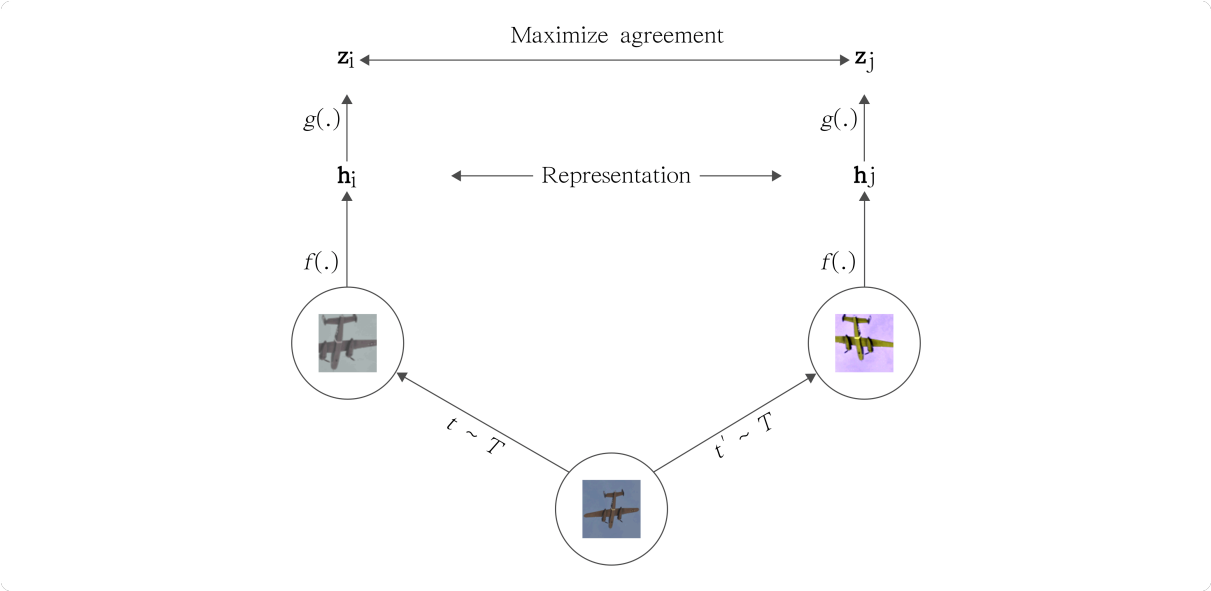


Figure 3.10 — A diagram that illustrates the architecture of the original SimCLR Network. $f(\cdot)$ represents the encoder, whereas $g(\cdot)$ the projection head

Once training is complete, the projection head is discarded, and the encoder is used as a standalone feature extractor for downstream tasks. The representations learned through this process have been shown to rival or outperform supervised learning on a wide range of visual benchmarks.

3.5.2.2 Normalized Temperature-scaled Cross Entropy loss

To train a contrastive learning model such as SimCLR, a specialized loss function is required, one that encourages the network to draw similar data points closer in the embedding space while pushing dissimilar ones further apart. SimCLR adopts the Normalized Temperature-scaled Cross Entropy loss (NT-Xent), introduced by Sohn (2016).

The loss is designed to maximize the similarity between the positive pair and simultaneously minimize the similarity between the anchor and all other negative samples. Cosine similarity is typically used to measure how close two vectors are in the representation space. These similarity scores are then passed through a softmax function to normalize them into a probability distribution. The model is penalized when negative pairs have higher similarity than the positive pair, and rewarded when the positive pair stands out as the most similar, effectively guiding the model to learn meaningful and discriminative representations.

Figure 3.11 illustrates the contrastive learning process using the NT-Xent loss. The first image shows a batch containing several individuals. In the second image, for each kinship pair, the model maximizes the similarity between related individuals while minimizing similarity with all other individuals in the batch. This effectively pushes unrelated samples apart in the feature space. The third image highlights the result for a single kinship pair, where the embeddings move closer together. When this process is repeated across the entire batch, the final image shows the ideal outcome: a well-structured feature space where related individuals form tight clusters.

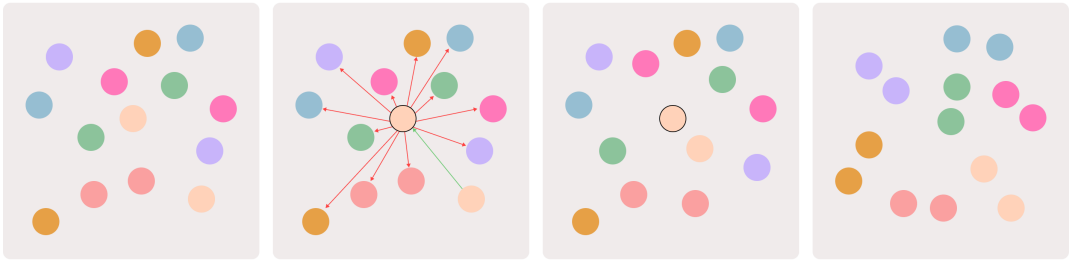


Figure 3.11 — Illustration of the contrastive training process using NT-Xent loss. Positive pairs are pulled together, while negatives are pushed apart, resulting in a structured feature space.

Equation 3.3 shows the mathematical formulation of the NT-Xent loss. In this formulation, let (z_i, z_j) be a positive pair, i.e., the projected embeddings of two augmented views of the same sample. The goal is to make z_i similar to z_j, while dissimilar to the remaining $2N-2$ samples in the batch. Here, z_k denotes any other embedding in the batch.

The similarity function $\mathrm{sim}(z_i, z_j)$ is defined as the cosine similarity between the two l_2-normalized vectors:

A crucial component of this loss is the temperature parameter τ. This parameter controls the sharpness of the softmax distribution: lower temperatures make the model focus more on the hardest negatives (those that are most similar to the anchor), while higher temperatures result in a smoother distribution and a more balanced treatment of all negatives.

$$l_{i,j} = -\log \frac{\exp\big(\mathrm{sim}(z_i, z_j)/\tau\big)}{\sum_{k=1}^{2N} \mathbb{1}\{k \neq i\} \exp(\mathrm{sim}(z_i, z_k)/\tau)} \qquad \mathrm{sim}\big(z_i, z_j\big) = \frac{z_i \cdot z_j}{\|z_i\| \cdot \|z_j\|} \tag{3.3}$$

3.5.2.3 Adaptation for label aware learning

The primary distinction between our approach and the original SimCLR framework lies in the construction of positive pairs. SimCLR operates in a self-supervised setting, where positive pairs are created by applying different augmentations to the same image. This formulation assumes no

access to labels and relies on data transformations to simulate semantic similarity. In contrast, our method leverages known kinship labels and family relationships to form positive pairs directly. Specifically, we use pairs of images or video frames of two biologically related individuals, converting the framework into a supervised contrastive learning setting (X. Zhang *et al.*, 2021). This modification aligns the training objective more closely with the downstream task of kinship verification, where actual familial relationships serve as supervision signals.

One notable limitation of the original SimCLR framework is the possibility of false negatives. Since all other examples in the batch are treated as negative samples, embeddings of semantically similar or related individuals may be inadvertently pushed apart. This is especially problematic in kinship verification tasks, where multiple members of the same family could appear in a single batch. To reduce this risk, SimCLR relies on extremely large batch sizes (e.g., 4096) to lower the ratio of related individuals being treated as negatives. However, such large batches significantly increase the computational burden.

Our method addresses this issue by enforcing a constraint during training: each batch contains samples from only one unique family. This ensures that all other individuals in the batch, aside from the positive pair, are truly unrelated and can be safely treated as negatives. By structuring the batches this way, we preserve the reliability of the contrastive loss while avoiding the need for large batch sizes.

To operationalize this strategy, we design a custom batch sampler inspired by (Bendib, 2023). This sampler is invoked at the beginning of each training epoch and dynamically constructs batches by randomly selecting individuals from different families. To promote diversity and prevent memorization, care is taken to avoid grouping the same individuals together repeatedly across epochs. This variability helps the model generalize more effectively to unseen kinship configurations.

4
Experimental Setup

In this section, we outline the procedures used to prepare and tune our models, followed by a description of how the data is adjusted across the three experimental conditions.

4.1 Model Preparation

Each of the two proposed models requires its own configuration to achieve optimal performance, particularly due to differences in the backbone architectures. This applies to both the tuning of hyperparameters and architectural components.

To ensure robust evaluation and minimize the influence of data variance, we employ 10-fold cross-validation during the tuning phase. This approach provides more reliable performance estimates and reduces the risk of overfitting to specific data splits.

As described in Section 3.2.3, the data is partitioned such that no video samples or identities overlap between the training, validation, or test sets. This guarantees that folds are created without the risk of data leakage. Each fold contains either different individuals or different age versions of individuals, ensuring clean separation.

We begin with the configuration of KinFusionNet. Each input pair is processed through a shared encoder, resulting in two fixed-dimensional feature vectors. These vectors are fused using a combination of the following element-wise operations absolute difference: $|x - y|$, average: $\frac{1}{2}(x + y)$ and multiplicative interaction: $x \cdot y$ This fusion strategy is motivated by the findings of Li and Jiang (2023), who demonstrated that combining multiple fusion operations yields better performance than any single strategy.

Next, we investigate the effect of different MLP configurations. Various architectures are evaluated to determine which best exploits the fused feature vector, including variations in network depth (number of layers) and width (number of units per layer). We also test the effect of excluding hidden

layers altogether. During this phase, all models are trained using the Adam optimizer with a learning rate of $4e-3$ and a batch size of 32, hyperparameters known to work effectively with Adam.

A total of 10 MLP architectures are evaluated as shown in Table 4.3

name	dimension
deep-0-hidden-layers	[] (no hidden layers)
deep-1-hidden-layer	[2 × input-dim]
deep-2-hidden-layers	[2 × input-dim, input-dim]
deep-3-hidden-layers	[2 × input-dim, input-dim, 256]
wide-1-hidden-layer	[3 × input-dim]
wide-2-hidden-layers	[3 × input-dim, 3 × input-dim]
wide-3-hidden-layers	[3 × input-dim, 3 × input-dim, 3 × input-dim]
extra-wide-1-hidden-layer	[5 × input-dim]
extra-wide-2-hidden-layers	[5 × input-dim, 3 × input-dim]
extra-wide-3-hidden-layers	[5 × input-dim, 3 × input-dim, 1 × input-dim]

Table 4.3 — Experiments to test effect of different MLP setups

Before proceeding with hyperparameter tuning, we first isolate and evaluate the effect of the MLP architecture on performance. Performing architecture search and hyperparameter tuning simultaneously would result in an unmanageable number of configurations. By identifying the best-performing MLP structure first, we reduce the complexity of the search space.

Once the optimal MLP architecture is selected, we continue with a focused hyperparameter tuning phase. In this step, we explore different values for the number of training epochs, learning rate, weight decay, and batch size to further refine the model's performance as shown in Table 4.4

parameter	options
batch size	[32, 64, 128]
epochs	[10, 20, 30]
learning rate	[0.0001, 0.0005, 0.001, 0.005]
weight decay	[0.0, 0.0001, 0.001]
momentum	[0.95, 0.90, 0.85]
optimizer	[SGD, Adam]

Table 4.4 — Search space for hyperparameter tuning KinFusionNet

d metrics: Accuracy, Precision, Recall, and F1 Score. Since the dataset is balanced, we use accuracy as the primary criterion for model selection.

For the FLACL model, no architectural modifications are needed, allowing us to proceed directly with hyperparameter tuning. In addition to standard parameters such as learning rate and batch size, we also tune the learning rate decay and the temperature parameter (τ) used in the NT-Xent loss, as these can significantly affect the quality of the learned embeddings.

parameter	options
batch size	[32, 64]
epochs	[30, 50, 80]
learning rate	[0.0001, 0.0005, 0.001, 0.005]
weight decay	[0.0, 0.0001, 0.001]
momentum	[0.95, 0.90, 0.85]
optimizer	[SGD, Adam]
steps learning rate decay	[10, 20, 30]
gamma learning rate decay	[0.1, 0.2, 0.5]
τ	[0.07, 0.7, 1.7]

Table 4.5 — Search space for hyperparameter tuning FLACL

Because FLACL relies on similarity scores rather than a binary classifier, we evaluate its performance using the Receiver Operating Characteristic (ROC) curve, specifically focusing on the Area Under the Curve (AUC). The final classification threshold is selected based on the point on the ROC curve that yields the highest accuracy.

4.2 Experiment 1: Full Data Usage

In the first experiment, we use the complete dataset without applying any additional frame filtering, aside from the standard preprocessing steps described in Section 3.2.2. This approach ensures that the model is exposed to the full range of natural variation present in the video data, including frames with suboptimal conditions such as motion blur or off-angle views.

4.3 Experiment 2: Frame Quality Filtering

In the second experiment, we introduce frame filtering strategies to improve the quality of the training data. Specifically, we focus on two factors: facial orientation and image quality.

During a video sequence, individuals naturally move their heads, occasionally resulting in extreme angles where the face is only partially visible or entirely obscured. To address this, we estimate the head orientation for each frame using the model proposed by Hempel, Abdelrahman and Al-Hamadi (2022). This model estimates three key pose metrics: pitch, yaw, and roll. Based on manual inspection, we define the following thresholds for acceptable orientation: yaw within ±30° and pitch

within ±15°. Frames exceeding these bounds are discarded. The roll is already corrected during preprocessing by aligning the eyes horizontally, as described in Section 3.2.2.

In addition to pose filtering, we apply a no-reference image quality assessment using the BRISQUE score (Mittal, Moorthy and Bovik, 2012). This metric assigns a score between 1 and 100 to each image, where lower values indicate better visual quality. We discard any frame with a BRISQUE score above 40, as these are typically blurry or distorted and may degrade model performance.

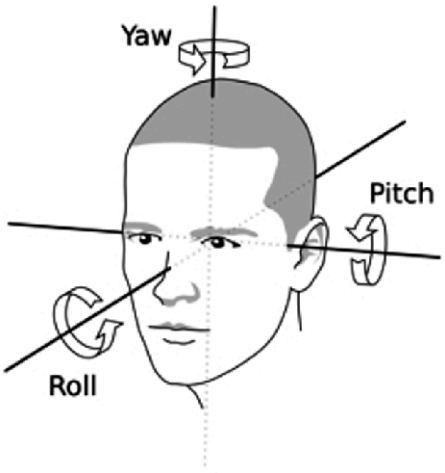


Figure 4.12 — Illustration of facial orientation using pitch, yaw, and roll angles.

4.4 Experiment 3: Best Frame Selection

In the final experiment, we shift from video-based to image-based kinship recognition by selecting the best-quality frame from each video. The chosen frame is the one where the individual's head orientation is closest to a neutral position, that is, the pitch and yaw angles are nearest to 0 degrees. This selection ensures that each sample represents a clear, forward-facing view of the subject, which has been shown to be optimal for facial recognition tasks.

Due to the change in input format, we also switch the model backbone from MARLIN to ResNet-50. Since this experiment introduces a different backbone and data format, the hyperparameter tuning process must be repeated. All key parameters, such as learning rate, batch size, weight decay, and fine-tuning strategy, are re-evaluated to ensure optimal performance for the new configuration.

5 Results

In the following sections, we present the results of our experiments. We begin with the baseline model, discussing its performance across the three experimental setups. Afterwards, we analyze the results of the contrastive learning approach. Lastly, we compare and analyse the results.

5.1 KinFusionNet

We first identify the optimal MLP architecture, tuned separately for the MARLIN (video-based) and ResNet (image-based) backbones. As shown in Table 5.6, a single wide hidden layer consistently yielded the best results, indicating that increased capacity helps the model better interpret the fused embeddings.

Backbone	input-dim	hidden-layers-dim	Accuracy	Precision	Recall	F1
Video	3072	5120	0.61616	0.61409	0.63390	0.62165
Image	1536	2560	0.64138	0.65190	0.61008	0.62954

Table 5.6 — Average of the mlp architecture tuning phase for the KinFusionNet models on the validations folds

With this architecture in place, we conducted hyperparameter tuning, adjusting epochs, batch size, learning rate, and weight decay. As shown in Table 5.7, the performance gains from this phase were modest, suggesting the architecture choice had a greater impact than fine-tuning these parameters.

Backbone	epochs	batch-size	learning rate	weight decay	Accuracy	F1
Video	20	128	0.0001	0.0001	0.602145	0.600326
Image	30	32	0.00001	0.01	0.66041	0.65975

Table 5.7 — Averages of the hyperparameter-tuning for the KinFusionNet models on the validation folds

5.1.1 Test results

Final test evaluations were conducted using the optimal MLP configuration and tuned hyperparameters for each model. The classification threshold was not fixed at 0.5; instead, it was selected based on the ROC curve. Specifically, we determined the threshold that maximized accuracy by finding the point where the true positive rate (TPR) and false positive rate (FPR) yielded the highest classification performance. The ROC curves for all experiments are shown in Figure 5.13.

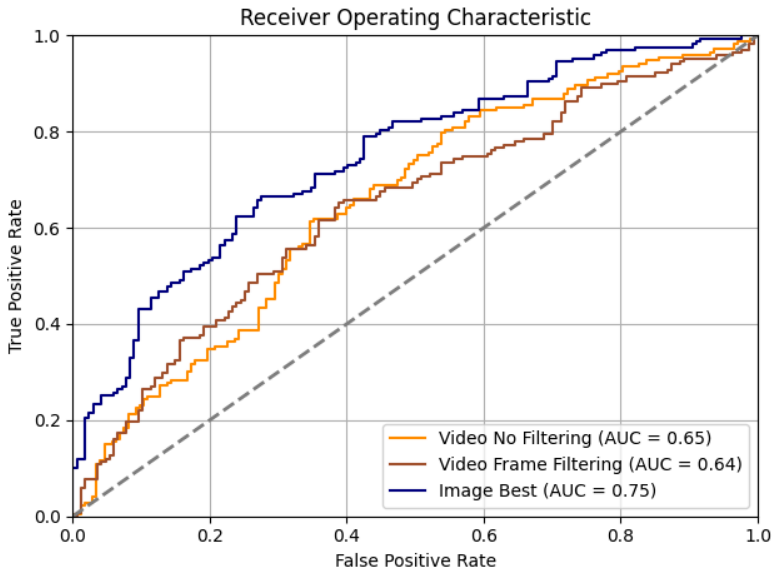


Figure 5.13 — ROC curves for the baseline models across all three experiments, including AUC scores.

As seen in Figure 5.13, frame filtering shows no noticeable performance benefit. This is also reflected in the accuracy curves across the 10-fold cross-validation in Figure 5.14, which remain consistent regardless of filtering strategy. One explanation may be that filtering removes frames that, despite being imperfect, still contain valuable variation in facial features. Alternatively, it may suggest that the model has limited capacity to extract fine-grained discriminative features, making it relatively insensitive to moderate improvements in input quality.

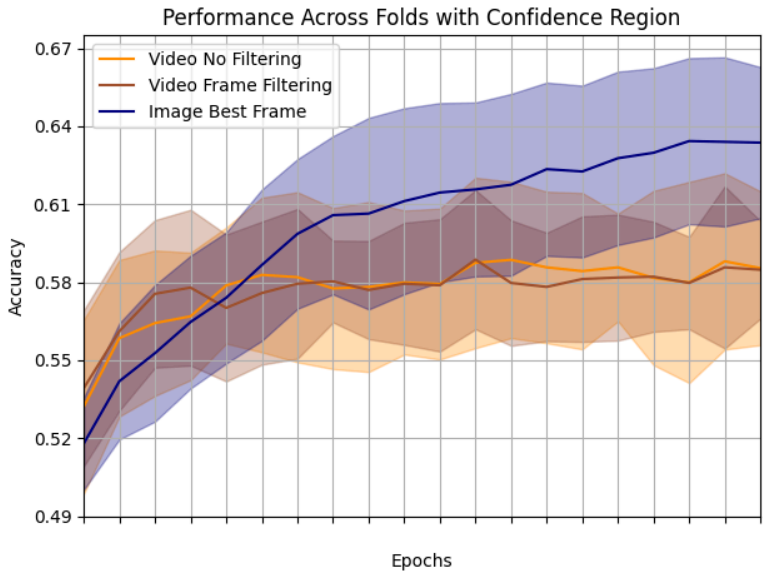


Figure 5.14 — Accuracy confidence interval of KinFusionNet models over 10-fold cross validation

These results show that selecting the best frame from each video yields the strongest performance among the baseline setups. Several factors may explain this. First, the ResNet model benefits from pre-training on a much larger and more diverse dataset than the MARLIN video model. This gives it a substantial advantage in generalizing to new data. Second, building a representation from video requires handling more variation, such as changes in facial expression and head pose, whereas a single, well-aligned image may offer a more stable and informative input for the task.

5.2 Family Label Aware Constrastive Learning (FLACL)

We now turn to the results of the Family Label Aware Contrastive Learning (FLACL) model. This model introduces several additional hyperparameters compared to KinFusionNet, including factors specific to contrastive learning such as the temperature parameter τ in the NT-Xent loss. The results of the tuning phase are summarized in Table 5.8. The metrics shown are averages over validation folds and include: BS (Batch Size), Epo (Epochs), LR (Learning Rate), WD (Weight Decay), Mom (Momentum), Wa (Warm-Up Epochs), LRSS (Learning Rate Step Size), Gam (Step Size Gamma), τ (Temperature), and AUC (Area Under the Curve).

Backbone	BS	Epo	LR	WD	Mom	Wa	LRSS	Gam	τ	AUC
Video	16	100	0.003	0	0.95	50	10	0.8	0.1	0.55
Image	32	50	0.001	0.0005	0.95	10	10	0.8	0.1	0.77

Table 5.8 — Averages of the hyperparameter-tuning for the FLACL models on the validation folds

5.2.1 Test results

With the tuned parameters, the image-based FLACL model achieves strong performance, reaching an average AUC of 0.77 on the validation folds. This suggests that the contrastive objective, guided by kinship labels, effectively improves the feature space. In contrast, the video-based model performs only marginally above chance (AUC = 0.55), despite a stable decrease in training loss. This gap indicates a difficulty in generalizing from video inputs, potentially due to weaker pretraining or the added complexity of modeling temporal variation.

To gain insight into how training reshapes the learned representation space, we plot the cosine similarity distributions before and after training. As shown in Figure 5.15, the trained image-based model pushes positive pair similarities further to the right, separating them more clearly from non-kin pairs. This shift confirms that the contrastive loss function succeeds in pulling kinship pairs closer together in the embedding space, improving the model's ability to distinguish between related and unrelated individuals.

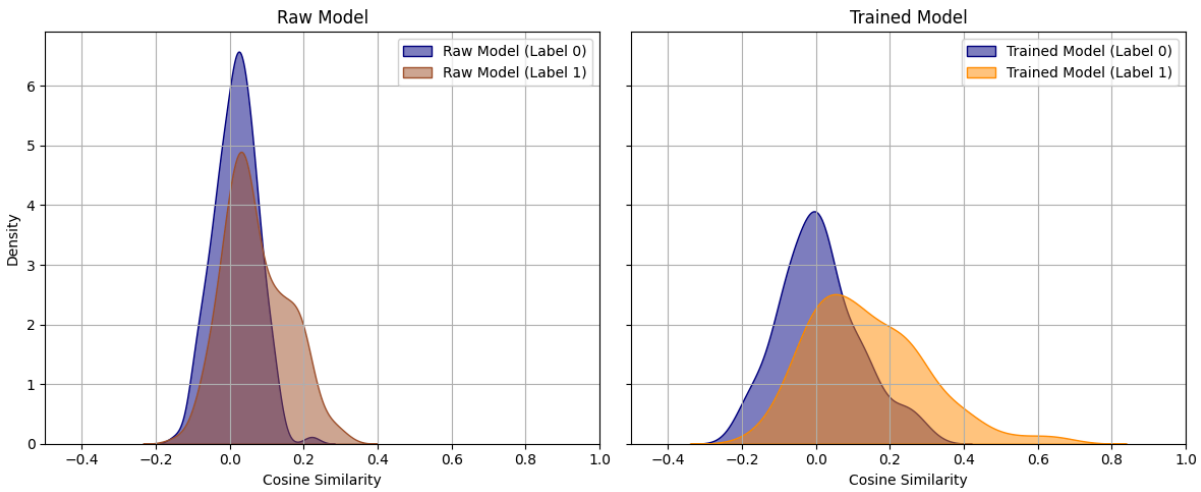


Figure 5.15 — Distribution difference between the untrained and trained image backbone

5.3 Model Comparison & Analysis

In this section, we compare the performance of all proposed models and benchmark them against results reported in (Kefalas *et al.*, 2023). The scores shown reflect the performance of each model after cross-validation and threshold optimization via the ROC curve.

To better understand how each model handles specific relationship types, we include per-label accuracy scores. The final column reports accuracy on non-related (NR) pairs. The distribution of relationship types within the KAN-AV dataset is shown in Table 5.9.

To clarify the label abbreviations used throughout the results: *NR* (Non-Related), *FS* (Father–Son), *BB* (Brother–Brother), *FD* (Father–Daughter), *MD* (Mother–Daughter), *BS* (Brother–Sister), *SS* (Sister–Sister), and *MS* (Mother–Son).

Labels	NR	FS	BB	FD	MD	BS	SS	MS
Distribution	0.50	0.101	0.101	0.075	0.066	0.060	0.060	0.034

Table 5.9 — The distribution of kinship relationships in the KAN-AV Dataset

Across all models, the FLACL model using the image backbone achieves the highest overall accuracy at 70.1% ± 5.24. However, given the overlapping confidence intervals, it is not possible to claim a statistically significant improvement over the next-best model, KinFusionNet with frame filtering, which achieved 69.5% ± 4.82. These two results are therefore best interpreted as comparable under current experimental conditions

Compared to prior work by Kefalas *et al.* (2023), several of our models, especially those using image inputs, reach or exceed previously reported scores. While the FLACL model shows promising performance, further validation would be required to assert a clear state-of-the-art advantage, particularly given the confidence interval overlap and the variation across relationship categories.

Input	Model	All	BB	SS	BS	FD	FS	MD	MS	NR
Video	Triplet Loss (Kefalas *et al.*, 2023)	**66.8**	70.0	74.2	60.9	58.2	67.6	73.3	64.8	-
Image	Triplet Loss (Kefalas *et al.*, 2023)	**69.1**	74.5	76.0	63.0	58.2	71.9	75.7	63.7	-
Video	KinFusionNet	**61.8** ± *5.12*	77.1	81.0	71.4	61.5	68.6	56.5	41.7	56.1
Video	KinFusionNet with Frame Filtering	**63.2** ± *5.08*	64.7	70.0	40.0	45.8	50.0	65.2	50.0	70.7
Video	FACL	**55.4** ± *5.03*	76.5	85.0	75.0	54.2	79.4	73.9	75.0	36.5
Image	KinFusionNet	**65.0** ± *4.85*	45.7	23.8	33.3	38.5	42.9	56.5	50.0	88.4
Image	KinFusionNet with Frame Filtering	**69.5** ± *4.82*	67.6	65.0	55.0	58.3	70.6	82.6	66.7	71.9
Image	FACL	**70.1** ± *5.24*	82.4	80.0	60.0	37.5	73.5	56.5	33.3	76.0

Table 5.10 — Comparison of model performance across relationship types. Results from (Kefalas *et al.*, 2023) are included for reference.

Among the video-based models, frame filtering does not lead to consistent improvements. Although filtering slightly increases the overall accuracy over the unfiltered baseline (63.2% vs. 61.8%), it decreases performance across most relationship types. The exception is the non-related category, where filtering results in a significant boost. The video-based FLACL model shows strong performance on specific relationship types, particularly BB, SS, BS, FD, and FS, but suffers from poor performance on NR pairs, leading to a much lower overall score (55.4%). This result highlights a critical limitation: the model may overfit to kinship cues while failing to distinguish unrelated individuals, which is essential for real-world kinship verification.

In contrast, the image-based models benefit more clearly from filtering. The baseline model with frame filtering improves across nearly all relationship types compared to the unfiltered version. This supports the idea that carefully selected frames, particularly those with good frontal pose and clarity, provide more consistent inputs for learning kinship features.

To further explore the impact of frame selection, we tested the image-based FLACL model using randomly sampled frames (from the first five in each video) instead of the best frame. This led to a significant drop in performance, with the overall accuracy falling to just 60%. This confirms that frame selection plays a critical role in the success of image-based kinship verification and reinforces the value of pre-processing strategies that emphasize quality and alignment.

5.3.1 Finetuned encoder

As described earlier, the FLACL model refines its encoder through supervised contrastive learning, optimizing the feature space by drawing embeddings of related individuals closer together and pushing apart those of non-related individuals. This training encourages the model to structure its internal representation space in a way that captures subtle familial similarities more effectively.

To assess whether this improved representation can also benefit simpler models, we conducted an experiment where the FLACL-refined backbone was reused in the KinFusionNet architecture. In this setup, the backbone weights are frozen, and only the MLP classifier is trained. The rest of the model pipeline, including the fusion strategy and training procedure, remains unchanged. This setup isolates the effect of the contrastively fine-tuned embedding space on downstream kinship classification, without introducing any changes to the classifier itself.

The results, shown in Table 5.11, demonstrate an overall accuracy of 72.9%, representing a modest 2% absolute gain over the previous best KinFusionNet setup. Notably, performance improvements are observed across several relationship categories, including non-related (NR) pairs. These findings suggest that contrastive learning not only helps in structuring the feature space for similarity-based inference (as in FLACL) but also enhances the discriminative capacity of standard classifiers like MLPs. In practice, this shows the potential of reusing refined embeddings in simpler architectures to improve both accuracy and generalization.

All	BB	SS	BS	FD	FS	MD	MS	NR
72.9	94.1	80.0	50.0	62.5	58.8	82.6	75.0	71.3

Table 5.11 — Performance of the baseline model using the fine-tuned FLACL backbone.

6
Discussion

This thesis investigated the impact of pre-trained models on facial kinship classification, particularly in settings where labeled data is limited. To answer this, we designed and evaluated two fine-tuning strategies, KinFusionNet, a Siamese-style classification model, and FLACL, a supervised contrastive learning framework. Both were tested using video-based and image-based inputs across three experiments, allowing us to analyze the influence of data modality, pretraining strategy, and model design.

The first major observation is that using video data for kinship classification proved far more difficult than anticipated. In Experiment 1, KinFusionNet with the MARLIN video backbone achieved just over 60% accuracy, despite using the full dataset. To improve this, we introduced frame quality and pose filtering in Experiment 2, but this yielded only marginal gains. This suggests that the challenge is not merely in data noise, but in the model's ability to extract stable, discriminative features from temporally dynamic inputs. The MARLIN backbone, while designed for facial video reconstruction, may lack the pretraining scale or diversity needed to generalize across the subtle variations required for kinship recognition.

FLACL showed more promise, particularly in specific kinship types such as father-daughter and brother-sister. However, its inability to consistently identify non-related pairs led to a drop in overall accuracy. The design choice use small batches, structured per family-id helped reduce false negatives, but may have inadvertently limited the diversity of non-kin pairings, which are crucial for contrastive learning.

Experiment 3 brought a turning point. Switching to image-based inputs and selecting the best frame from each video resulted in significant performance improvements. Both KinFusionNet and FLACL achieved their highest scores in this setup, with FLACL reaching 70.1% accuracy. This suggests that, contrary to common assumptions, a single high-quality frame can outperform full video sequences, at least under current modeling strategies. The image-based ResNet backbone benefited from extensive pretraining on large-scale identity datasets, which likely contributed to its superior generalization and discriminative power.

Looking back, several choices in the experimental setup may have influenced the final results. First, we only evaluated a single backbone per modality, MARLIN for video and ResNet for images. Testing additional architectures could have helped determine whether the challenges observed, particularly in the video-based models, stemmed from the task of kinship recognition itself or from limitations in model design or pretraining. Exploring other video backbones, especially those trained on larger or more diverse datasets, might have provided valuable insights.

Second, in the frame filtering experiment, we focused on removing low-quality or poorly posed frames but did not attempt to improve the quality of the remaining data. Given that many video frames were low-resolution or blurred, applying super-resolution techniques before resizing and cropping could have enhanced facial detail and improved the model's ability to extract discriminative features.

6.1 Future Work

Building on the findings of this research, several directions could be explored to advance facial kinship recognition. One of the most immediate challenges is the limited availability of high-quality video datasets. Acquiring or constructing a larger, more diverse kinship video dataset would enable a more comprehensive evaluation of video-based models and potentially allow architectures like MARLIN to realize their full potential.

In addition to data improvements, future work could also focus on enhancing model interpretability. Deep learning methods often offer little transparency into which facial features are being used to determine kinship, making it difficult to verify or trust model predictions. Incorporating explainability techniques, such as saliency maps or attention-based visualizations, could help reveal which facial regions the model relies on, providing insight into both its strengths and limitations.

Another promising research direction lies in further optimizing the contrastive learning framework. This might include combining it with supervised classification objectives or expanding it to support multi-positive training, where the model learns from multiple related individuals within the same family. These extensions could improve the model's ability to generalize and better capture the subtle visual patterns that define familial relationships.

7
Conclusion

This thesis examined the effectiveness of pre-trained models in facial kinship recognition under limited supervision. Specifically, it evaluated two fine-tuning strategies, KinFusionNet and Family Label Aware Contrastive Learning, across both video and image-based inputs. The aim was to assess how different fine tuning strategies and data modalities affect kinship classification performance.

The results show that image-based approaches consistently outperform video-based models in terms of accuracy and generalization. The best-performing model was the image-based FLACL network, which achieved an overall accuracy of 70.1% on the KAN-AV dataset. This performance was closely followed by the KinFusionNet image model using frame filtering. These outcomes suggest that, under current modeling conditions, a single, high-quality frame may offer more informative and stable input for kinship verification than full video sequences.

By contrast, video-based models, despite leveraging temporally rich data, did not provide consistent improvements. The video-based FLACL model performed well on specific relationship types but struggled with non-kin classification, leading to lower overall accuracy. Moreover, frame filtering strategies did not significantly improve performance, possibly due to limitations in video quality and the capacity of the MARLIN backbone to model complex temporal dependencies.

Several factors likely contributed to these outcomes. The ResNet backbone was pre-trained on a larger and more diverse dataset than the MARLIN encoder, which may have influenced the generalization capability of the respective models. Additionally, contrastive learning showed some benefit in refining feature representations, as evidenced by improved performance when the fine-tuned backbone from FLACL was reused in a baseline classifier.

In conclusion, this study finds that high-quality image inputs, combined with strong pre-trained models and appropriate fine-tuning strategies, provide a reliable basis for kinship verification. While video offers richer temporal information, further work is needed to develop models and pretraining strategies that can fully leverage this modality.

Bibliography

Belabbaci, E.O. *et al.* (2023) “High-Order Knowledge-Based Discriminant Features for Kinship Verification,” *Pattern Recognition Letters*, 175, pp. 30–37. Available at: https://doi.org/10.1016/j.patrec.2023.09.008.

Bendib, N. (2023) *Supervised Contrastive Learning and Feature Fusion for Improved Kinship Verification*. Available at: https://arxiv.org/abs/2302.09556.

Bordallo Lopez, M. *et al.* (2018) “Kinship Verification from Facial Images and Videos: Human versus Machine,” *Machine Vision and Applications*, 29, pp. 873–890. Available at: https://doi.org/10.1007/s00138-018-0943-x.

Boutellaa, E. *et al.* (2017) “Kinship Verification from Videos Using Spatio-Temporal Texture Features and Deep Learning.” arXiv. Available at: https://doi.org/10.48550/arXiv.1708.04069.

Burch, R.L. and Gallup, G.G. (2000) “Perceptions of Paternal Resemblance Predict Family Violence,” *Evolution and Human Behavior*, 21, pp. 429–435. Available at: https://doi.org/10.1016/S1090-5138(00)00056-8.

Cai, Z. *et al.* (2023) “MARLIN: Masked Autoencoder for Facial Video Representation LearnINg,” in *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. Vancouver, BC, Canada: IEEE, pp. 1493–1504. Available at: https://doi.org/10.1109/CVPR52729.2023.00150.

Cao, Q. *et al.* (2018) *VGGFace2: A dataset for recognising faces across pose and age*. Available at: https://arxiv.org/abs/1710.08092.

Chen, T. *et al.* (2020) “A Simple Framework for Contrastive Learning of Visual Representations.” arXiv. Available at: https://doi.org/10.48550/arXiv.2002.05709.

Chen, X. *et al.* (2020) “Improved Baselines with Momentum Contrastive Learning.” arXiv. Available at: https://doi.org/10.48550/arXiv.2003.04297.

DeBruine, L.M. *et al.* (2009) “Kin Recognition Signals in Adult Faces,” *Vision Research*, 49, pp. 38–43. Available at: https://doi.org/10.1016/j.visres.2008.09.025.

Deng, J. *et al.* (2022) “ArcFace: Additive Angular Margin Loss for Deep Face Recognition,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(10), pp. 5962–5979. Available at: https://doi.org/10.1109/tpami.2021.3087709.

Devlin, J. *et al.* (2019) "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding." arXiv. Available at: https://doi.org/10.48550/arXiv.1810.04805.

Dibeklioglu, H. (2017) "Visual Transformation Aided Contrastive Learning for Video-Based Kinship Verification," in *2017 IEEE International Conference on Computer Vision (ICCV)*. Venice: IEEE, pp. 2478–2487. Available at: https://doi.org/10.1109/ICCV.2017.269.

Dibeklioglu, H., Salah, A.A. and Gevers, T. (2013) "Like Father, Like Son: Facial Expression Dynamics for Kinship Verification," in *2013 IEEE International Conference on Computer Vision*. Sydney, Australia: IEEE, pp. 1497–1504. Available at: https://doi.org/10.1109/ICCV.2013.189.

Dibekliŏglu, H., Salah, A.A. and Gevers, T. (2012) "Are You Really Smiling at Me? Spontaneous versus Posed Enjoyment Smiles," *Computer Vision – ECCV 2012*. Berlin, Heidelberg: Springer Berlin Heidelberg. Available at: https://doi.org/10.1007/978-3-642-33712-3_38.

Dornaika, F., Arganda-Carreras, I. and Serradilla, O. (2020) "Transfer Learning and Feature Fusion for Kinship Verification," *Neural Computing and Applications*, 32, pp. 7139–7151. Available at: https://doi.org/10.1007/s00521-019-04201-0.

Ericsson, L. *et al.* (2022) "Self-Supervised Representation Learning: Introduction, Advances, and Challenges," *IEEE Signal Processing Magazine*, 39, pp. 42–62. Available at: https://doi.org/10.1109/MSP.2021.3134634.

Fang, R. *et al.* (2013) "Kinship Classification by Modeling Facial Feature Heredity," in *2013 IEEE International Conference on Image Processing*. Melbourne, Australia: IEEE, pp. 2983–2987. Available at: https://doi.org/10.1109/ICIP.2013.6738614.

Fang, R. *et al.* (2010) "Towards Computational Models of Kinship Verification," in *2010 IEEE International Conference on Image Processing*, pp. 1577–1580. Available at: https://doi.org/10.1109/ICIP.2010.5652590.

Gao, P. *et al.* (2019) "What Will Your Child Look Like? DNA-Net: Age and Gender Aware Kin Face Synthesizer." arXiv. Available at: https://doi.org/10.48550/arXiv.1911.07014.

Gao, Z. and Patras, I. (2024) "Self-Supervised Facial Representation Learning with Facial Region Awareness." arXiv. Available at: https://doi.org/10.48550/arXiv.2403.02138.

Grill, J.-B. *et al.* (2020) "Bootstrap Your Own Latent: A New Approach to Self-Supervised Learning." arXiv. Available at: https://doi.org/10.48550/arXiv.2006.07733.

Guo, G. and Wang, X. (2012) “Kinship Measurement on Salient Facial Features,” *IEEE Transactions on Instrumentation and Measurement*, 61, pp. 2322–2325. Available at: https://doi.org/10.1109/TIM.2012.2187468.

Gupta, K. *et al.* (2022) *Understanding and Improving the Role of Projection Head in Self-Supervised Learning*. Available at: https://arxiv.org/abs/2212.11491.

He, K. *et al.* (2020) “Momentum Contrast for Unsupervised Visual Representation Learning.” arXiv. Available at: https://doi.org/10.48550/arXiv.1911.05722.

He, K. *et al.* (2015) *Deep Residual Learning for Image Recognition*. Available at: https://arxiv.org/abs/1512.03385.

Hempel, T., Abdelrahman, A.A. and Al-Hamadi, A. (2022) “6d Rotation Representation For Unconstrained Head Pose Estimation,” in *2022 IEEE International Conference on Image Processing (ICIP)*, pp. 2496–2500. Available at: https://doi.org/10.1109/ICIP46576.2022.9897219.

Huang, G. *et al.* (2008) “Labeled Faces in the Wild: A Database forStudying Face Recognition in Unconstrained Environments,” *Tech. rep.*, p. .

Iman, M., Arabnia, H.R. and Rasheed, K. (2023) “A Review of Deep Transfer Learning and Recent Advancements,” *Technologies*, 11, p. 40. Available at: https://doi.org/10.3390/technologies11020040.

Kaminski, G. *et al.* (2009) “Human Ability to Detect Kinship in Strangers' Faces: Effects of the Degree of Relatedness,” *Proceedings of the Royal Society B: Biological Sciences*, 276, pp. 3193–3200. Available at: https://doi.org/10.1098/rspb.2009.0677.

Kefalas, T. *et al.* (2023) “KAN-AV Dataset for Audio-Visual Face and Speech Analysis in the Wild,” *Image and Vision Computing*, 140, p. 104839. Available at: https://doi.org/10.1016/j.imavis.2023.104839.

Kim, M., Jain, A.K. and Liu, X. (2022) “AdaFace: Quality Adaptive Margin for Face Recognition,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*.

Klontz, J.C. and Jain, A.K. (2013) “A Case Study of Automated Face Recognition: The Boston Marathon Bombings Suspects,” *Computer*, 46, pp. 91–94. Available at: https://doi.org/10.1109/MC.2013.377.

Kohli, N. *et al.* (2017) “Hierarchical Representation Learning for Kinship Verification,” *IEEE Transactions on Image Processing*, 26, pp. 289–302. Available at: https://doi.org/10.1109/TIP.2016.2609811.

Kohli, N. *et al.* (2019) “Supervised Mixed Norm Autoencoder for Kinship Verification in Unconstrained Videos,” *IEEE Transactions on Image Processing*, 28, pp. 1329–1341. Available at: https://doi.org/10.1109/TIP.2018.2840880.

Kostinger, M. *et al.* (2012) "Large Scale Metric Learning from Equivalence Constraints," in *2012 IEEE Conference on Computer Vision and Pattern Recognition*. Providence, RI: IEEE, pp. 2288–2295. Available at: https://doi.org/10.1109/CVPR.2012.6247939.

Laiadi, O. *et al.* (2020) "Tensor Cross-View Quadratic Discriminant Analysis for Kinship Verification in the Wild," *Neurocomputing*, 377, pp. 286–300. Available at: https://doi.org/10.1016/j.neucom.2019.10.055.

Leeuwen, B. van *et al.* (2022) "Explainable kinship: The importance of facial features in kinship recognition," in *IARIA Congress 2022: The 2022 IARIA Annual Congress on Frontiers in Science, Technology, Services, and Applications*, pp. 54–60.

Li, D. and Jiang, X. (2023) "Kinship Verification Method of Face Image Deep Feature Fusion," *Academic Journal of Science and Technology*, 5, pp. 57–62. Available at: https://doi.org/10.54097/ajst.v5i1.5348.

Li, L. *et al.* (2016) "Kinship Verification from Faces via Similarity Metric Based Convolutional Neural Network," in *Image Analysis and Recognition*. Cham: Springer International Publishing, pp. 539–548. Available at: https://doi.org/10.1007/978-3-319-41501-7_60.

Li, W. *et al.* (2021) "Reasoning Graph Networks for Kinship Verification: From Star-Shaped to Hierarchical," *IEEE Transactions on Image Processing*, 30, pp. 4947–4961. Available at: https://doi.org/10.1109/TIP.2021.3077111.

Liang, J. *et al.* (2019) "Weighted Graph Embedding-Based Metric Learning for Kinship Verification," *IEEE Transactions on Image Processing*, 28, pp. 1149–1162. Available at: https://doi.org/10.1109/TIP.2018.2875346.

Lior, W., Tal, H. and Itay, M. (2011) "Face recognition in unconstrained videos with matched background similarity," in *CVPR 2011*, pp. 529–534. Available at: https://doi.org/10.1109/CVPR.2011.5995566.

Liu, F. *et al.* (2022) "Age-Invariant Adversarial Feature Learning for Kinship Verification," *Mathematics*, 10, p. 480. Available at: https://doi.org/10.3390/math10030480.

Lu, J. *et al.* (2014) "Neighborhood Repulsed Metric Learning for Kinship Verification," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 36, pp. 331–345. Available at: https://doi.org/10.1109/TPAMI.2013.134.

Dal Martello, M.F. and Maloney, L.T. (2006) "Where Are Kin Recognition Signals in the Human Face?," *Journal of Vision*, 6, p. 2. Available at: https://doi.org/10.1167/6.12.2.

Mensink, T. *et al.* (2021) "Factors of Influence for Transfer Learning across Diverse Appearance Domains and Task Types." arXiv. Available at: https://doi.org/10.48550/arXiv.2103.13318.

Mittal, A., Moorthy, A.K. and Bovik, A.C. (2012) "No-Reference Image Quality Assessment in the Spatial Domain," *IEEE Transactions on Image Processing*, 21(12), pp. 4695–4708. Available at: https://doi.org/10.1109/TIP.2012.2214050.

Pan, S.J. and Yang, Q. (2010) "A Survey on Transfer Learning," *IEEE Transactions on Knowledge and Data Engineering*, 22, pp. 1345–1359. Available at: https://doi.org/10.1109/TKDE.2009.191.

Parkhi, O.M., Vedaldi, A. and Zisserman, A. (2015) "Deep Face Recognition," in *Procedings of the British Machine Vision Conference 2015*. Swansea: British Machine Vision Association, p. 41. Available at: https://doi.org/10.5244/C.29.41.

Ramazankhani, F., Yazdian-Dehkord, M. and Rezaeian, M. (2023) "Feature Fusion and NRML Metric Learning for Facial Kinship Verification," *JUCS - Journal of Universal Computer Science*, 29, pp. 326–348. Available at: https://doi.org/10.3897/jucs.89254.

Redmon, J. and Farhadi, A. (2018) *YOLOv3: An Incremental Improvement*. Available at: https://arxiv.org/abs/1804.02767.

Robinson, J.P. *et al.* (2022) "Families in Wild Multimedia: A Multimodal Database for Recognizing Kinship," *IEEE Transactions on Multimedia*, 24, pp. 3582–3594. Available at: https://doi.org/10.1109/TMM.2021.3103074.

Robinson, J.P. *et al.* (2018) "Visual Kinship Recognition of Families in the Wild," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40, pp. 2624–2637. Available at: https://doi.org/10.1109/TPAMI.2018.2826549.

Serraoui, I. *et al.* (2022) "Knowledge-Based Tensor Subspace Analysis System for Kinship Verification," *Neural Networks*, 151, pp. 222–237. Available at: https://doi.org/10.1016/j.neunet.2022.03.020.

Shao, M., Xia, S. and Fu, Y. (2011) "Genealogical Face Recognition Based on UB KinFace Database," in *CVPR 2011 WORKSHOPS*, pp. 60–65. Available at: https://doi.org/10.1109/CVPRW.2011.5981801.

Sohn, K. (2016) "Improved Deep Metric Learning with Multi-class N-pair Loss Objective," in D. Lee et al. (eds.) *Advances in Neural Information Processing Systems*. Curran Associates, Inc., p. . Available at: https://proceedings.neurips.cc/paper_files/paper/2016/file/6b180037abbebea991d8b1232f8a8ca9-Paper.pdf.

Stone, Z., Zickler, T. and Darrell, T. (2010) "Toward Large-Scale Face Recognition Using Social Network Context," *Proceedings of the IEEE*, 98, pp. 1408–1415. Available at: https://doi.org/10.1109/JPROC.2010.2044551.

Tong, Z. *et al.* (2022) "VideoMAE: Masked Autoencoders are Data-Efficient Learners for Self-Supervised Video Pre-Training," in *Advances in Neural Information Processing Systems*.

Wang, H. *et al.* (2018) *CosFace: Large Margin Cosine Loss for Deep Face Recognition*. Available at: https://arxiv.org/abs/1801.09414.

Wang, S., Ding, Z. and Fu, Y. (2019) "Cross-Generation Kinship Verification with Sparse Discriminative Metric," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 41, pp. 2783–2790. Available at: https://doi.org/10.1109/TPAMI.2018.2861871.

Wang, W. *et al.* (2023) "A Survey on Kinship Verification," *Neurocomputing*, 525, pp. 1–28. Available at: https://doi.org/10.1016/j.neucom.2022.12.031.

Wang, Z., Chen, J. and Hu, J. (2022) "Multi-View Cosine Similarity Learning with Application to Face Verification," *Mathematics*, 10, p. 1800. Available at: https://doi.org/10.3390/math10111800.

Wu, L. *et al.* (2023) "Self-Supervised Learning on Graphs: Contrastive, Generative, or Predictive," *IEEE Transactions on Knowledge and Data Engineering*, 35, pp. 4216–4235. Available at: https://doi.org/10.1109/TKDE.2021.3131584.

Wu, X. *et al.* (2022) "Facial Kinship Verification: A Comprehensive Review and Outlook," *International Journal of Computer Vision*, 130, pp. 1494–1525. Available at: https://doi.org/10.1007/s11263-022-01605-9.

Wu, X. *et al.* (2024) "Audio-Visual Kinship Verification: A New Dataset and a Unified Adaptive Adversarial Multimodal Learning Approach," *IEEE Transactions on Cybernetics*, 54, pp. 1523–1536. Available at: https://doi.org/10.1109/TCYB.2022.3220040.

Xia, S., Shao, M. and Fu, Y. (2010) "Kinship Verification through Transfer Learning."

Xia, S., Shao, M. and Fu, Y. (2012) "Toward Kinship Verification Using Visual Attributes," in *Proceedings of the 21st International Conference on Pattern Recognition (ICPR2012)*, pp. 549–552.

Xia, S. *et al.* (2012) "Understanding Kin Relationships in a Photo," *IEEE Transactions on Multimedia*, 14, pp. 1046–1056. Available at: https://doi.org/10.1109/TMM.2012.2187436.

Yan, H. and Song, C. (2021) "Multi-Scale Deep Relational Reasoning for Facial Kinship Verification," *Pattern Recognition*, 110, p. 107541. Available at: https://doi.org/10.1016/j.patcog.2020.107541.

Yan, H. and Wang, S. (2019) “Learning Part-Aware Attention Networks for Kinship Verification,” *Pattern Recognition Letters*, 128, pp. 169–175. Available at: https://doi.org/10.1016/j.patrec.2019.08.023.

Yan, H., Lu, J. and Zhou, X. (2015) “Prototype-Based Discriminative Feature Learning for Kinship Verification,” *IEEE Transactions on Cybernetics*, 45, pp. 2535–2545. Available at: https://doi.org/10.1109/TCYB.2014.2376934.

Yan, H. *et al.* (2014) “Discriminative Multimetric Learning for Kinship Verification,” *IEEE Transactions on Information Forensics and Security*, 9, pp. 1169–1178. Available at: https://doi.org/10.1109/TIFS.2014.2327757.

Yi, D. *et al.* (2014) *Learning Face Representation from Scratch*. Available at: https://arxiv.org/abs/1411.7923.

Zebrowitz, L.A. and Montepare, J.M. (2008) “Social Psychological Face Perception: Why Appearance Matters,” *Social and Personality Psychology Compass*, 2, pp. 1497–1517. Available at: https://doi.org/10.1111/j.1751-9004.2008.00109.x.

Zhang, K. *et al.* (2015) “Kinship Verification with Deep Convolutional Neural Networks,” in *Procedings of the British Machine Vision Conference 2015*. Swansea: British Machine Vision Association, p. 148. Available at: https://doi.org/10.5244/C.29.148.

Zhang, L. *et al.* (2021) “AdvKin: Adversarial Convolutional Network for Kinship Verification,” *IEEE Transactions on Cybernetics*, 51, pp. 5883–5896. Available at: https://doi.org/10.1109/TCYB.2019.2959403.

Zhang, X. *et al.* (2021) “Supervised contrastive learning for facial kinship recognition,” in *2021 16th IEEE International Conference on Automatic Face and Gesture Recognition (FG 2021)*, pp. 1–5.

Zhao, Z. *et al.* (2024) “A Comparison Review of Transfer Learning and Self-Supervised Learning: Definitions, Applications, Advantages and Limitations,” *Expert Systems with Applications*, 242, p. 122807. Available at: https://doi.org/10.1016/j.eswa.2023.122807.

Zhou, X. *et al.* (2011) “Kinship Verification from Facial Images under Uncontrolled Conditions,” in *Proceedings of the 19th ACM International Conference on Multimedia*. Scottsdale Arizona USA: ACM, pp. 953–956. Available at: https://doi.org/10.1145/2072298.2071911.