

Siamese Neural Networks and Transfer Learning for Kinship Verification from Dermal Palm Images

Mazin H. Aziz

mazin.hazizi@uomosul.edu.iq

Computer Engineering Department, College of Engineering, University of Mosul, Mosul, Iraq

Received: April 26th, 2025 Received in revised form: July 2nd, 2025 Accepted: July 28th, 2025

ABSTRACT

Kinship verification is a crucial research area due to its diverse applications, including paternity tests, family reunions, and criminal investigations. While DNA analysis has been the predominant method, artificial intelligence techniques are still being explored and tested. Facial kinship verification, which involves comparing features between two facial images, has garnered significant research interest. This paper introduces a new approach to kinship verification using hand-palm images. The EfficientNetB0 model was utilized for deep feature extraction through transfer learning. A Siamese neural network architecture was employed to assess similarity. Various experimental scenarios were conducted concerning network architecture, training parameters, and fine-tuning. The Mosul Kinship Hand (MKH) dataset was used to create the palm dermal image dataset, consisting of 7,332 pairs equally divided into related and unrelated categories. The results were promising, achieving approximately 99% validation accuracy, and 77.02 ms average inference time per image pair using a post-training Principal Component Analysis (PCA) technique.

Keywords:

Kinship Verification; Siamese Neural Network; Deep Transfer Learning; EfficientNetB0; Hand-Palm Skin.

This is an open access article under the CC BY 4.0 license (<http://creativecommons.org/licenses/by/4.0/>).

<https://rengj.uomosul.edu.iq>

Email: alrafidain_engjournal3@uomosul.edu.iq

1. INTRODUCTION

Kinship verification (KV) is defined as the automated process of determining whether two or more individuals share a biological relationship, that is, whether they are kin or non-kin. This approach assumes that genetically related individuals exhibit some resemblances, which can be analyzed [1]. KV plays a significant role in paternity disputes, forensic investigations, reunification of families, and the identification of victims of disasters. It is helpful in reuniting families, solving cases in court, and determining biological relationships. Technology has increased its accuracy and speed, making it highly essential in forensic medicine, immigration, and medical research. While deoxyribonucleic acid (DNA) has long been the gold standard for kinship verification, image analysis algorithms offer a lower-cost and time-efficient approach [2]. However, DNA requires several hours or days to produce results and involves significant costs [3]. Thus, it cannot be used for real-time KV

applications [4]. Researchers have employed computer vision and artificial intelligence (AI) techniques for visual kinship verification (VKV), with a particular focus on facial kinship verification (FKV), which has garnered significant attention. VKV is not yet a substitute for DNA tests and requires considerable research, but it holds promise as a valuable complementary tool. Various traits are passed down from parents to their offspring through genetic inheritance. These include characteristics such as eye color, handedness [5], lip print patterns [6] [7], and fingerprint patterns [8][9].

Based on the impact of inherited familial traits, this study explores the potential use of hand palm skin texture for KV, investigating the use of deep transfer learning (DTL), especially the EfficientNetB0 model, and the Siamese neural network (SNN) architecture for image similarity analysis.

The rest of the paper's introduction addresses essential theories related to the core topics of this study and highlights key research

contributions. Subsequent sections feature a literature review, a description of the experimental setup, and a comprehensive research methodology divided into six scenario subsections. The paper proceeds with results and discussion, conclusions, acknowledgements, and references.

1.1. Computer Vision-Based Kinship Verification

Kinship verification involves three basic issues [10]: determining if two persons are related by blood through a kinship relationship, establishing parent-child relationships, and identifying an individual's relatives from a list of potential candidates [11]. KV fundamentally relies on comparing two individuals [12]. FKV is a method of automatically verifying familial relationships based on facial features. [13]. Over the past decade, significant advancements have been made in FKV, mainly due to the integration of deep learning techniques [11]. However, it remains in its early stages, with ongoing research addressing numerous associated challenges [14]. KV remains a vital area of study with significant implications across various domains [15].

1.2. Deep Transfer Learning (DTL)

KV feature learning and extraction methods are typically divided into three main categories: hand-crafted approaches, metric learning methods, and deep learning models [1]. Deep learning (DL) is widely regarded as a highly promising approach for KV due to its capability to automatically classify and extract features. However, to effectively address real-world challenges, there is a need for larger, more balanced datasets and more advanced methods [16]. DL-based approaches, particularly trained models such as VGG-Face and Facenet, have demonstrated superior results for FKV [16] [17]. The EfficientNets model was first introduced by M. Tan and Q. V. Le as a new scaling approach for the Convolutional Neural Network (CNN) ConvNets, based on a compound scaling approach [18]. They introduced a balanced strategy that simultaneously scales up the three dimensions of ConvNets—depth, width, and resolution—leading to significant performance improvements and the

creation of a new model family known as EfficientNets. Remarkably, EfficientNet-B7 achieved an accuracy of 84.3% while being 8.4 times smaller and 6.1 times faster than the most popular ConvNets at the time, such as MobileNets and ResNet.

The architecture of the pre-trained EfficientNetB0 model that was used as the SNN's backbone for feature extraction in this research comprises 237 layers, organized using 16 MBConv blocks combined with squeeze-and-excitation (SE) modules. It also has an initial Conv2D layer along with a final classification head, as shown in Fig. 1[19].

EfficientNets were used to enhance feature extraction capabilities for fingerprint classification, specifically targeting gender identification [20], and to classify the monkeypox skin lesions. It was found that EfficientNet-B0 is a highly effective model for skin lesion classification, outperforming various CNN architectures [21], and that it outperformed other CNN models in terms of precision, recall, and F1-score when used to improve the early detection and classification of brain tumors using MRI images [22] and for the classification of thoracic diseases [23].

1.3. Siamese Neural Network (SNN)

A Siamese Neural Network (SNN) is an effective tool for assessing the similarity between two images. It is designed to compare image pairs and evaluate their likeness. The SNN operates using two or more identical sub-networks that share the same parameters and features. The term "Siamese" highlights the interconnected architecture of these networks, similar to conjoined twins [24][25]. The initial version of SNNs was introduced by Bromley et al. [26] and has since gained significant and growing attention in practical applications [27]. SNNs are a type of deep learning architecture that excels in creating non-linear embeddings for a variety of machine learning tasks, especially those based on similarity [28]. The basic SNN architecture as depicted in Fig. 2, consists of two identical, weight-shared networks that extract features (embeddings) from

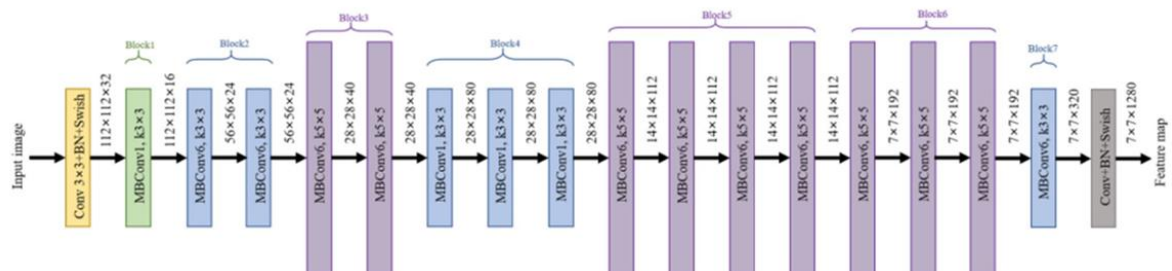


Fig. 1. Architecture of baseline EfficientNetB0.

input images X_1 and X_2 . These embeddings are then compared to evaluate similarity, which is used to classify the inputs as either related or unrelated. Recently, there has been renewed interest in this architecture due to advancements in neural networks, particularly in multimedia applications [29]. SNNs and their variants are highly effective for various computer vision tasks, especially when dealing with a large number of classes and a small number of samples per class [30].

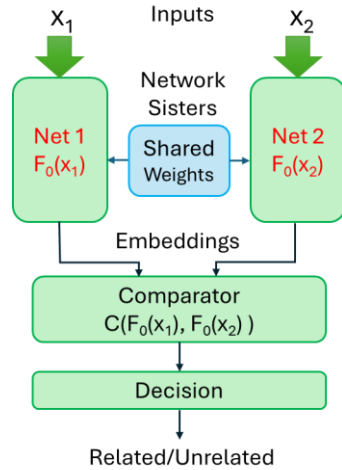


Fig. 2. The basic Siamese Neural Network (SNN) architecture.

SNNs are trained with labeled pairs (related/unrelated) to learn a similarity metric. During training, a feature space is created in a way that related pairs are near and unrelated pairs are distant. A similarity layer then looks at these pairs and classifies them according to a learned threshold [31]. Distance metrics, like cosine similarity (Equation 1) and Euclidean distance (Equation 2) [32], are utilized for measuring the output embeddings produced by twin subnetworks. This architecture proves advantageous in various applications, such as image comparison, object tracking, and face recognition, particularly for limited or imperfect training data sets [33] [34]. SNNs use a contrastive loss function (Equation) that is computed by summing up the losses over similar and dissimilar pairings. Contrastive Loss (L) is the loss function that penalizes dissimilar embeddings for positive pairs and similar embeddings for negative pairs, with a margin M (Equation 3), and (L_M) is the modified Contrastive Loss (Equation 4) designed to handle class imbalance by weighing positive and negative samples differently [32][35]. Through backpropagation, the network adapts its weights to make the distance between similar pairs smaller while keeping the distance between dissimilar pairs as large as some defined margin [31].

SNNs, primarily developed with deep learning frameworks, have only recently become viable for real-world applications due to their high computational demands. [27]. SNNs have recently advanced in terms of architectures, methods, and applications including face recognition, signature verification, gait analysis, tattoo recognition, and pedestrian tracking [30]. Siamese Neural Networks (SNNs) vary by twin network structure: (1) Simple SNNs have two branches with shared weights. (2) Pseudo SNNs feature branches with different weights or structures, ideal for varied input types. (3) Triplet networks use three branches to enhance deep metric learning by comparing an anchor input with a positive and a negative example. (4) Quadruplet and quintuplet networks allow for more complex input comparisons [27].

$$\text{Cosine similarity } (A, B) = \frac{A \cdot B}{\|A\| \cdot \|B\|} \quad \dots 1$$

Where:

$A \cdot B$ is the dot product of vectors A and B .

$\|A\| \cdot \|B\|$ are the magnitudes (norms) of vectors A and B , respectively.

Euclidean Distance: $d(A, B)$

$$= \sqrt{\sum_{i=1}^n (A_i - B_i)^2} \quad \dots 2$$

$$L(y, d) = y \cdot d^2 + (1 - y) \cdot \max(M - d, 0)^2 \quad \dots 3$$

$$L_M(y, d) = y \cdot (d^2 - \omega_p) + (1 - y) \cdot \max(M - d, 0)^2 \quad \dots 4$$

- $y \in \{0, 1\}$ is the label (1 for similar pairs, 0 for dissimilar).
- $\omega_p = \frac{\#negative\ samples}{\#positive\ samples + \epsilon}$ balances class imbalance.
- $M=1.5$ is the margin hyperparameter.

1.4. Contributions

Key contributions of this work include (1) the introduction of a novel palm-skin-based kinship verification (PSKV) method. (2) The effectiveness of EfficientNetB0 for deep feature extraction in skin images via a Siamese Neural Network (SNN). (3) The achievement of state-of-the-art accuracy, surpassing current computer vision techniques in kinship verification. (4) The implementation of an SNN architecture with fully

connected similarity layers leads to significant performance enhancements.

2. LITERATURE REVIEW

This section examines studies utilizing computer vision for kinship verification, delves into related research on Siamese neural networks, and highlights the application of deep transfer learning techniques. The proposed method for kinship verification using palm skin images presents a new approach with no existing equivalent in prior research.

Othmani et al. proposed using ResNet50 to extract features from paired facial images to determine kinship by measuring feature distances. They highlight that unbalanced datasets significantly lower accuracy, stressing the need for balanced training samples [36]. Another study proposes a deep learning approach using an SNN to predict kinship between individuals based on their facial features, achieving a validation accuracy of 65% [37]. A proposed family-aware convolutional neural network (FA-CNN) classifier achieved acceptable performance on the Family in the Wild (FIW) facial dataset for VKV, with an average accuracy of 68.84% [38]. The AdvKin (Adversarial Convolutional Network for Kinship Verification) model is proposed by N. Nader et al. for KV using a family ID-based adversarial convolutional network. Extensive experiments conducted on both small-scale benchmarks and the large-scale Families in the Wild (FIW) dataset demonstrate the superiority of the AdvKin model over existing state-of-the-art approaches in KV tasks [39].

SNN architecture was established by T. Navghare et al. based on deep learning algorithms such as ResNet and VGGNet. The focus was on four kinship relations: father-son, father-daughter, mother-son, and mother-daughter. The proposed model achieved an average similarity score of 72.73% using the created dataset which comprised facial images from 96 families, including 410 images and over 77,000 distinct pairs [40]. FKV using a deep SNN architecture applied to the Families In the Wild (FIW) dataset, indicating that cosine similarity outperforms L1 and L2 norms, achieving higher accuracy across various kinship types.[41]. C. Bisogni and F. Narducci employed SNNs utilizing a VGGFace architecture to conduct experiments to distinguish kinship versus non-kinship and identify specific kinship types, using two established datasets: Faces in the Wild and KinFace-II. In their experiments, the SNNs exhibited a maximum accuracy of 75% for kinship recognition tasks [42]. J. Yu, G. addressed three key tasks: kinship verification, tri-subject kinship

verification, and kinship retrieval, utilizing a deep fusion SNN to achieve these objectives. The authors explore two methods for similarity computation: fully connected similarity and cosine similarity, both of which aid in ranking the similarity scores to identify potential relatives[43]. The study conducted by R. Annisa and B. Soewito explores the effectiveness of MobileNet and SNN in analyzing the M2FRED dataset, which focuses on mobile face recognition under the constraints imposed by the COVID-19 pandemic. The results revealed that MobileNet significantly outperformed SNN across all metrics, achieving an overall accuracy of 99.85%, including 100% accuracy in mask scenarios. In contrast, SNN exhibited an accuracy of only 49.41% [33]. Another approach has adopted transforming facial images of parents and children to a common age range of 15-19 years and using a deep relational network for post-age transformation image processing. A triplet SNN was used to optimize the distances between anchor (parent), positive (child), and negative (other parent) images. The results demonstrate an accuracy rate of 76.38% [44]. A Siamese architecture, GLANet, was proposed, combining the strengths of Transformers and CNNs to enhance the discriminative feature extraction required for accurate KV [45]. A deep fusion SNN model was produced for the tri-subject FKV task. The network calculates the kinship similarity score by combining the individual similarity scores of the father-child and mother-child pairs. The paper demonstrates the transition from traditional, handcrafted, feature-based techniques to contemporary deep learning methods [43]. A deep learning SNN architecture was proposed for family member retrieval from facial images. The approach consisted of two primary components: similarity computation and ranking. The authors experimented with various combinations of backbone networks and training methods to optimize performance. In their findings, the authors highlight that while both fully connected similarity and cosine similarity were utilized during training, cosine similarity yielded better results during inference [46].

Far from using FKV, researchers developed their own dataset named Mosul Kinship Hand (MKH), which contains 648 images from 81 individuals across 14 families. The study employed Google MediaPipe for hand detection and segmentation, subsequently extracting 43 geometric features from the images. A neural network classifier was then designed and trained, achieving a prediction accuracy of 93% [47], and 92.8% using DTL via a ResNet50 model [48]. The findings suggest that hand geometry harbor distinct biometric traits that can effectively indicate



Fig. 3. Samples from the MKH dataset.

kinship. Another study investigates KV using ear images. The authors introduce the KinEar dataset, which consists of 1,477 images from 19 families, totaling 37,282 kinship pairs. The paper employs an SNN architecture, utilizing five advanced deep learning models, including VGG16, ResNet-152, USTC-NELSLIP, Attentional Feature Fusion (AFF), and Contextual Transformer Network (CoTNet). Experimental results indicate that ear images can effectively be used for kinship verification, with four out of five models achieving over 60% in Area Under the Receiver Operating Characteristics (ROC-AUC). Notably, the VGG16 model achieved the highest performance, with an ROC-AUC score of 69.22% [49].

Results from prior studies indicated that, particularly when combined with DTL techniques, SNNs are exceedingly effective for similarity analysis as well as KV. In addition, EfficientNet models were found to be suitable for mining deep features out of skin images. Although facial KV is quite common in these studies, only preliminary works have been reported on other traits, which could be a gap in reviewed research literature. Our goal with this work is to address the gap by taking a DTL-SNN approach and using an EfficientNetB0 model for kinship verification from palm skin images.

3. EXPERIMENTAL SETUP

The proposed image dataset, evaluation metrics, and computer specifications are the three topics covered in this section.

3.1. Dataset

This section details the generation of a skin image dataset from the source data MKH (Mosul Kinship Hand) and its subsequent preparation for SNN implementation.

MKH Dataset: The skin image dataset used in this work was derived from the MKH hand image dataset [47], which contains images from 84 individuals (44 females, 40 males, aged 3-70) spanning 15 families. Each participant contributed eight images: two palm and two dorsal images per

hand, with both open and closed finger poses. Refer to Fig. 3 for samples from the MKH dataset.

Hand-Palm Skin Image Extraction: For this work, only the hand-palm images from the MKH dataset were used. Specifically, four region of interest (ROI) images of the palm from each of the 84 subjects were processed to create palm skin images. The preprocessing involved cropping the ROI to a standard size of 762x762 pixels. If cropping was not possible, the images were resized. Due to an artifact in the original MKH dataset, only family 15 had 16 images instead of the expected 20. The final dataset comprised 332 labelled images, organized by families. Fig. 4 shows samples from the dataset, which was then used to construct the dataset needed for the SNN.

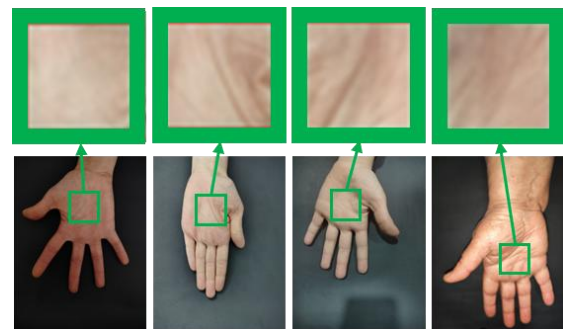


Fig. 4. Palm skin ROI cropping to generate the palm dermal image dataset.

Dataset Preparation for Siamese NN: A Python script was created to generate a labelled palm skin image dataset for training and validating a Siamese Neural Network (SNN). The script produced image pairs categorized as "related" (individuals from the same family, labelled with a value of 1) or "unrelated" (individuals from different families, labeled with a value of 0). The dataset comprised 14,664 images, forming 7,332 pairs. These pairs were evenly distributed, with 3,666 classified as related and 3,666 as unrelated. Related pairs included combinations such as parent-child, father-mother, and siblings, covering both same-gender and opposite-gender siblings. Unrelated pairs were randomly selected from different families. The datasets are available to interested researchers upon request to the author.

3.2. Metrics

The proposed methods were evaluated using standard training and validation metrics, including accuracy, precision, recall, and F1-score [50]. These metrics were calculated based on true positive (TP), true negative (TN), false positive (FP), and false negative (FN) instances, following the equations (5-8). Furthermore, both the training duration and the average inference time across 100 runs were evaluated. High accuracy signifies that

the model correctly predicts a significant portion of instances compared to the total predictions; however, it is insufficient on its own for comprehensive model evaluation. To enhance assessment, the remaining metrics should be employed, where high recall reflects fewer false negatives, while high precision indicates minimal false positives. The F1 score balances precision and recall, providing a more holistic view of performance. Conversely, high training accuracy paired with lower validation accuracy suggests the presence of mild overfitting.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad \dots 5$$

$$Precision = \frac{TP}{TP + FP} \quad \dots 6$$

$$Recall = \frac{TP}{TP + FN} \quad \dots 7$$

$$F1 - Score = 2 * \frac{TPrecision * Recall}{Precision + Recall} \quad \dots 8$$

3.3. Computer specifications

All experiments were conducted on a desktop PC equipped with an Intel® Core™ i5-9600 CPU @ 3.70GHz, 16 GB of RAM, a 64-bit operating system, and an x64-based processor, without a GPU. All experiments were performed on a CPU due to hardware constraints. Future efforts will focus on transitioning the training and inference workflows to cloud platforms or GPU-accelerated systems (e.g., Google Colab, AWS EC2) to enhance scalability and efficiency.

4. METHODOLOGY

This section outlines the methodology used in the study, illustrating the proposed scenarios. The EfficientNetB0 pre-trained model was employed as the foundational architecture for both branches of the SNN. The Siamese structure comprises two identical branches that process input image pairs using shared weights (weight-sharing constraint), ensuring comparable feature extraction for both inputs. To optimize performance, we investigated several DTL strategies, encompassing fine-tuning, alternative loss functions, similarity metrics, Principal Component Analysis (PCA), and modifications to batch size, transfer learning rate, and early stopping criteria. For all operational scenarios, the dataset was partitioned into a training set (80%)

and a validation set (20%). The common batch size was 64 samples, even though several batch sizes were experimented with.

4.1. First Scenario (SNN with Cosine Similarity)

An SNN architecture with cosine similarity was employed. The model architecture is depicted in Fig. 5, where the outputs from the SNN twin arms which are the embeddings from the two input images represent the deep features of each. The two embeddings were fed to a lambda layer that calculates the cosine similarity, then to the fully connected (dense) layer with 256 units and a ReLU activation function to learn a non-linear transformation of the cosine similarity. The next layer is a batch normalization layer, which normalizes the activations of the previous layer (mean = 0, standard deviation = 1) to stabilize and speed up training. It helps reduce internal covariate shift and improves generalization. To prevent the model from relying too heavily on specific neurons, the output is fed to a dropout layer, which randomly sets 50% of the input units to 0 during training. The last layer is a dense layer with 1 neuron and a sigmoid activation function that squashes the output to a value between 0 and 1, which is useful for the binary classification task. All the layers of the EfficientNetB0 were freeze on the pretrained weights and the training were employed on the rest layers. This model was trained for 75 epochs. Adam optimizer with 0.0001 learning rate, and binary cross entropy loss function.

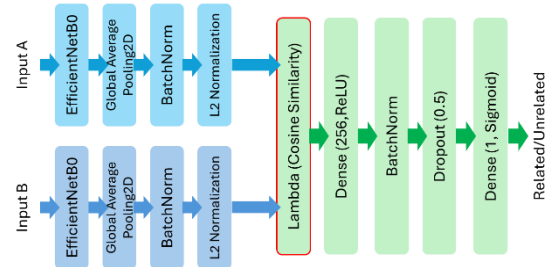


Fig. 5. The proposed SNN architecture for the first scenario. In the second scenario, the only change is replacing the cosine similarity-lambda layer with a Euclidean distance layer.

4.2. Second Scenario (Euclidean distance)

The same architecture used in the first scenario was applied, replacing the cosine similarity layer with a Euclidean distance layer to compute the Euclidean distance between the two embeddings generated by the SNN for each pair of images. All the training parameters were kept as in the first scenario.

4.3. Third Scenario (FC layers)

The SNN architecture was somewhat modified to overcome some of the drawbacks of the previous scenarios. However, instead of using cosine similarity or Euclidean distance, this implementation uses a fully connected (FC) layer to learn the similarity between the embeddings of the two input images. Traditional SNNs often use a distance metric (e.g., L1, cosine) between embeddings, followed by a threshold. However, this architecture replaces the metric with a learnable classifier head. This architecture is illustrated in Fig. 6. The two embeddings were processed independently before being merged through a global average pooling (GAP) layer, followed by a batch normalization layer. The GAP layer reduces spatial dimensions of convolutional features (output of EfficientNetB0) to a 1D vector, preserving channel-wise information. The batch normalization layer stabilizes training by normalizing activations post-GAP. The two outputs were merged via a feature concatenation layer, which merges embeddings from both inputs to form a joint representation for subsequent classification. It concatenates 1280 features' vector from the two embeddings to create a single feature vector of 2560 features. The resultant vector is passed through the following FC layers in sequence: Dense (512), batch normalization, dropout (0.5), dense (256), batch normalization & dropout repeated, and the output layer is a dense layer with sigmoid activation. The concatenated vector (2560-dim) is progressively compressed to 512, then to 256, and finally to 1, thereby avoiding the curse of dimensionality and focusing on task-relevant features. Batch normalization mitigates sensitivity to weight initialization and learning rates. The stacked dense layers act as a similarity comparator, distilling high-dimensional embeddings into a single confidence score.

The first two scenarios utilize a lambda layer to compute similarity (cosine or Euclidean), whereas this one employs a concatenate layer followed by fully connected layers, thereby learning the similarity measure from the data. The training methodology adopted the binary cross-

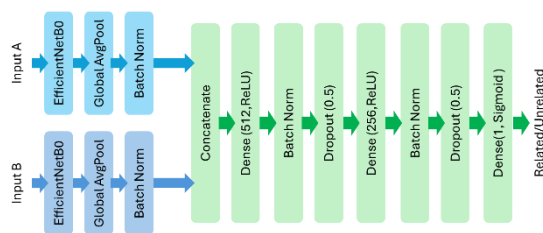


Fig. 6. The proposed SNN architecture for the third scenario features fully connected layers designed to learn similarity through training.

entropy as a loss function and the Adam optimizer with a learning rate of 0.0001, with no fine-tuning.

4.4. Fourth Scenario (Improved FC layers)

The fully connected (FC) layers proposed in the previous architecture were optimized with key enhancements to regularization and training stability. To promote simpler decision boundaries and improve generalization, L2 weight regularization ($\lambda=0.01$) was applied to both the 512-unit and 256-unit dense layers as depicted in Fig. 7. This penalizes excessively large weights, effectively reducing overfitting by encouraging sparser feature utilization. Training dynamics were further refined through early stopping (patience=10 epochs) to halt training upon validation loss convergence, thereby preventing overfitting, and adaptive learning rate reduction (factor=0.2, patience=5 epochs) to stabilize gradient updates. The original loss function (binary cross-entropy) and optimizer (Adam) were retained to maintain consistency in the optimization objectives.

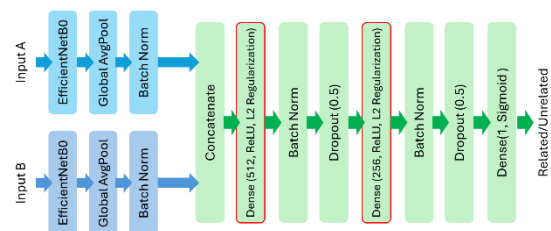


Fig. 7. The proposed architecture for the fourth scenario.

4.5. Fifth Scenario (Improved FC layers with PCA)

A refined SNN architecture was developed as shown in Fig. 8, incorporating PCA for post-training feature space optimization. The architecture includes several advanced training mechanisms: (1) a contrastive learning framework that combines Euclidean distance metrics with

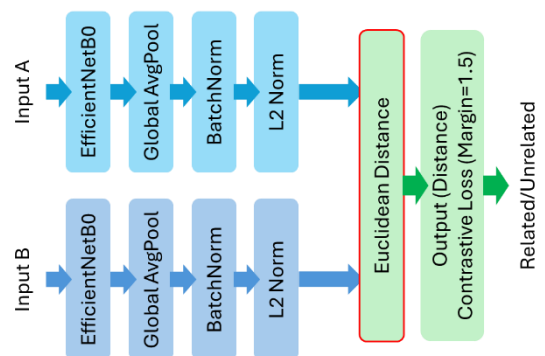


Fig. 8. The SNN architecture suggested for scenario 5.

margin-based contrastive loss to improve discriminative feature learning; (2) a regularization scheme using L2 weight decay ($\lambda=0.01$) on fully connected layers to prevent overfitting; and (3) training optimization with adaptive learning rate scheduling, plateau detection, and early stopping (patience=10 epochs) to maximize convergence efficiency.

4.6. Sixth Scenario (Fine tuning, PCA, with contrastive loss)

This scenario integrates six methodological advancements to optimize dermatological image analysis. First, it employs layer-specific fine-tuning, freezing early EfficientNetB0 layers to retain generic features while adapting the last 10 layers to domain-specific patterns. Second, L2 normalization projects the embeddings onto a unit hypersphere, thereby stabilizing distance-based similarity metrics. Third, an Adam optimizer with cosine decay scheduling ($\eta_{\max}=1e-4$, $\eta_{\min}=1e-6$ over 50 epochs) ensures smooth convergence. Fourth, ROC-derived threshold selection maximizes diagnostic accuracy by balancing sensitivity and specificity. Fifth, balanced batch sampling (with a 1:1 positive-negative ratio) mitigates class imbalance during training. Finally, a class-weighted contrastive loss function replaces binary cross-entropy, dynamically adjusting penalties for minority/majority classes. Fig. 9 depicts the proposed architecture for this scenario. These refinements collectively address key challenges in medical image analysis: preserving the utility of pretrained feature through selective fine-tuning, ensuring metric stability via normalized embeddings, and optimizing decision boundaries for clinical applicability. The architecture's design aligns with established principles of transfer learning [51] and contrastive representation learning [52], while introducing domain-specific adaptations for dermatological data.

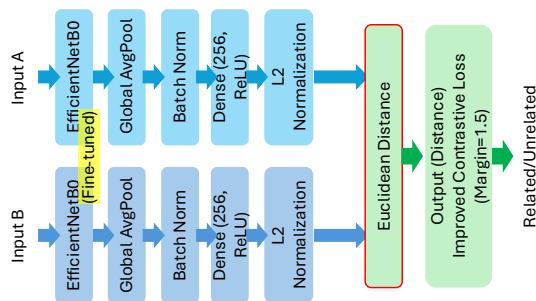


Fig. 9. The architecture of SNN employed in the sixth scenario.

5. RESULTS AND DISCUSSION

The proposed methods were assessed using standard training and validation metrics, alongside accuracy and loss graphs shown in Fig. 10. High accuracy signifies that the model correctly predicts a significant portion of instances compared to the total predictions; however, it alone is insufficient for comprehensive model evaluation. To enhance assessment, additional metrics are employed: high recall reflects fewer false negatives, while high precision indicates minimal false positives. The F1 score balances precision and recall, providing a more holistic view of performance. Conversely, high training accuracy paired with lower validation accuracy suggests the presence of mild overfitting.

The initial two scenarios yielded unsatisfactory results. In the first scenario, the training accuracy reached 74%, and validation accuracy was 78%, with high loss values of approximately 0.55 for training and 0.45 for validation after 75 epochs. The second scenario showed 64% accuracy for both training and validation, with a loss of about 0.64 for each.

Table 1 presents the comparison results for scenarios 3 through 6, highlighting the top metric in bold. It can be noticed that the sixth scenario excels across various metrics, achieving the highest contrastive accuracy in both training (99.51%) and validation (98.93%), alongside superior precision (99.43%) and F1-score (99.15%). It requires the fewest epochs (43) and the least training time (14,491 seconds). The average inference time, which was calculated over

Table 1. Metric comparison among scenarios 3-6.

Metric	Third Scenario	Fourth Scenario	Fifth Scenario	Sixth Scenario
Training Accuracy	97.48%	97.34%	99.04%	Contrastive 99.51%
Validation Accuracy	96.45%	97.59%	98.43%	Contrastive 98.93%
Precision	97.94%	96.62%	97.04%	99.43%
Recall	99.95%	99.03%	99.86%	98.86%
F1-Score	98.93%	97.81%	98.43%	99.15%
Optimal Threshold	--	--	--	0.6989
Training Epochs	75	75	100	43
Training Time (s)	25048	24955	34866	14491
Inference Time (ms)	87.83	78.35	77.02	78.52

100 runs, was moderately fast at 78.52 milliseconds. The fifth scenario closely follows with a training accuracy of 99.04% and a compact model size (~5 MB post-PCA), although it demands the most extended training duration (34,866 seconds). The third and fourth scenarios offer competitive yet less optimized outcomes, with the fourth scenario demonstrating better generalization capabilities, as evidenced by a

validation accuracy of 97.59% compared to the third's 96.45% while achieving a balanced inference time of 78.35 ms. Key insights include the advantage of the sixth scenario in contrastive learning, which efficiently balances speed and performance efficiently. The fifth scenario's PCA reduction offers a trade-off between reduced model size and increased computational cost achieving a minimal inference time of approximately 77 ms.

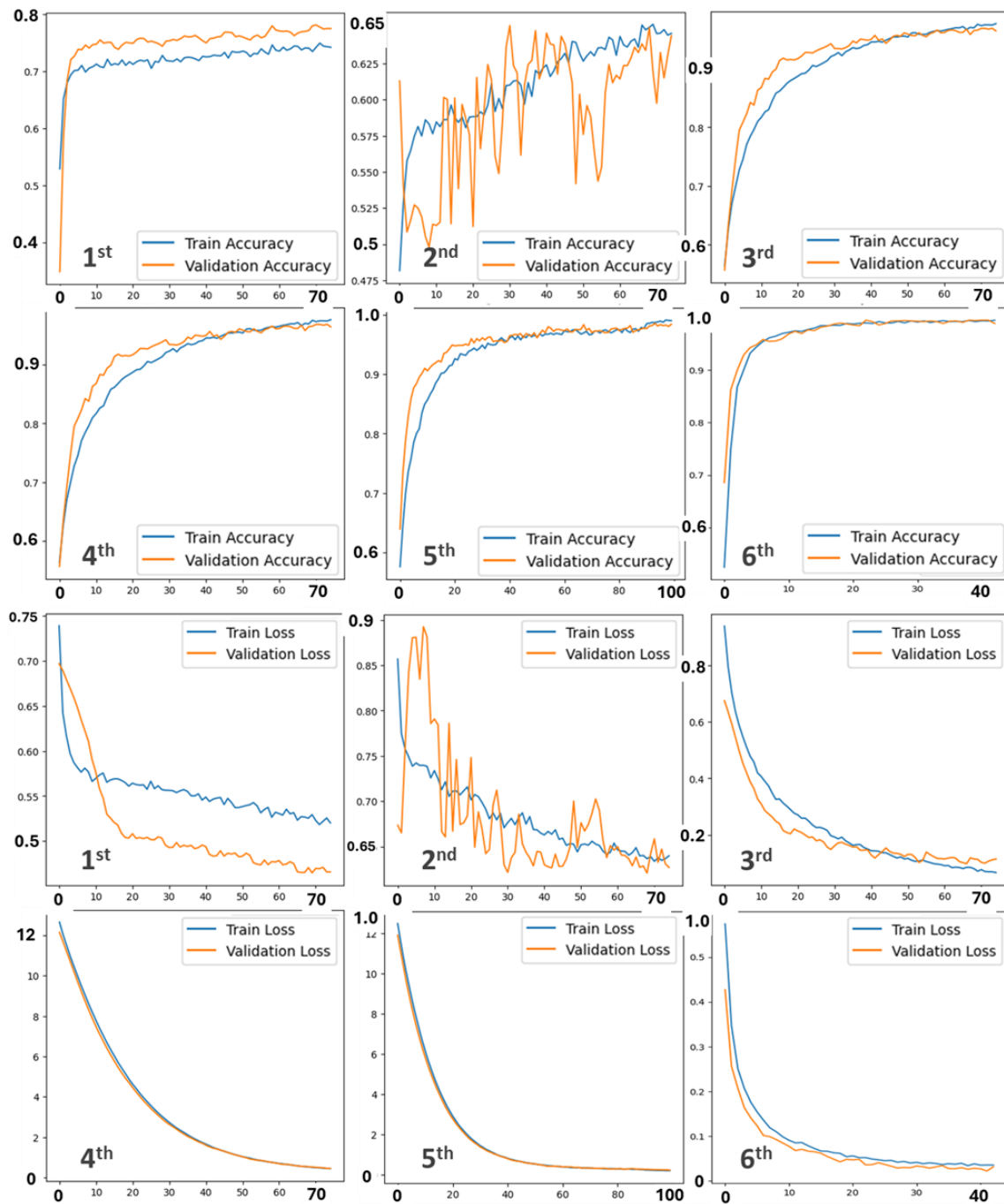


Fig. 10. Training and validation curves for accuracy and loss across the proposed six scenarios. Rows 1 and 2 display accuracy (Y-axis: accuracy), while Rows 3 and 4 show loss values (Y-axis: loss). The X-axis represents the number of epochs in all plots. "1st" corresponds to the first scenario, and so on.

While the third scenario focuses on maximizing recall (99.95%), the sixth optimizes precision (99.43%). The longest inference time, recorded at 87.83ms, occurred during the third scenario.

Fig. 10 illustrates the training/validation plots for all proposed scenarios, showcasing performance improvements. The proposed modification to architectural and training techniques minimizes erratic behavior and the gap between training and validation curves, reduces accuracy climb time, and loss descent time, and lowers the epoch count required to achieve optimal accuracy and loss.

The good performance of the sixth scenario can be attributed to the fine-tuning of the last 10 layers of the EfficientNetB0, which enhances the capture of fine details from the dermal images, balanced batch sampling, an enhanced contrastive loss, cosine learning rate scheduling, and optimal threshold selection using ROC-AUC. We may conclude that the technique of similarity checks via learning from the two embeddings using the FC layers outperforms the cosine and the Euclidean distance measures. The use of L2 regularization prevented overfitting, and the use of the post-training PCA technique reduced the trained model size and the inference time. The average inference time per image pair ranged from 77.02 ms (Scenario 5 with PCA) to 87.83 ms (Scenario 3 without optimization).

The limitations include a small dataset, a restricted variety of families, image data collected under controlled conditions, and manual ROI cropping, all of which affect generalizability. The proposed approach for KV using palm skin texture has not been previously implemented,

making it a novel contribution to the field. Comparisons were made with other visual KV methods, including facial recognition, hand geometry, and ear shape, as shown in Table 2. While exact dataset-based comparisons are rare due to different input modalities, this table highlights general performance trends across kinship verification strategies. Although direct comparisons are challenging due to varying datasets and methodologies, this analysis aimed to assess the acceptability of the proposed method within the realm of visual KV techniques. Notably, the proposed method significantly outperforms the others, suggesting several factors: the dermal image dataset was captured under controlled conditions, palm images are less susceptible to external noise, and palm dermal modality may inherently be simpler compared to facial or ear datasets. Additionally, the results might stem from the effective synergy between the dermal image data and the proposed architecture and training strategies. Dermal images leverage texture details that exhibit higher intra-class consistency. Furthermore, palm skin features fewer details than facial and ear images, facilitating easier similarity comparisons.

6. CONCLUSIONS

This study explored the potential of using palm dermal images as a new method for kinship verification through AI techniques. Results from various experiments using a skin image dataset that was produced from the MKH dataset indicated that this approach is both feasible and comparable to

Table 2. Comparison with other KV methodologies.

Note: Comparisons are provided for illustrative purposes only. Due to the absence of standardized dermal image datasets, results from various modalities (e.g., facial, ear, palm) are presented for perspective only

Reference	Method	DTL Model	Type of KV	Best Accuracy
[36]	ResNet50+DTL	ResNet50	Facial Images	77.25%
[37]	SNN	--		72%
[38]	FA-CNN	FaceNet & SphereFace		68.84%
[39]	AdvKin	--		89.9
[40]	SNN	ResNet & VGGNet		72.73%
[41]	SNN	SqueezeNet		67.66%
[43]	SNN	SENet50, VGG16 & ResNet50		73.8%
[44]	SNN + Age Transformation	ResNet50		76.38%
[45]	SNN-GLANet	Resnet50 & PVT		79.6 %
[47]	FF-NN Classifier	--	Hand-Images	93%
[48]	FF-NN Classifier	ResNet50	Hand-Images	92.8%
[49]	SNN	VGG16, ResNet-152, USTC-NELSLIP & AFF	Ear Images	64%
Ours	SNN	EffecientNetB0	Palm Dermal Images	99%

state-of-the-art facial kinship verification methods. However, to enhance generalization, experiments should be conducted with a larger dataset, which is currently unavailable. The EfficientNetB0 DTL model, when fine-tuned, showed promise for extracting dermal features. Similarity detection using a Siamese Neural Network (SNN) with fully connected layers trained with contrastive loss and ROC-AUC adaptive thresholding demonstrated superior performance. This preliminary step can be further evaluated and improved in future research by exploring other SNN architectures, such as triplet networks, additional DTL models, and more robust similarity assessment methods. While PCA was used to reduce the model size post-training, future research should also investigate model compression methods, such as weight pruning, quantization, and knowledge distillation to further optimize the model for edge deployment.

ACKNOWLEDGEMENTS

The author would like to thank the University of Mosul College of Engineering, and the Computer Engineering Department for their valuable supports to this work. The author also wants to declare that there is not any conflict of interest with this work.

REFERENCES

- [1] W. Wang, S. You, S. Karaoglu, and T. Gevers, "A survey on kinship verification," *Neurocomputing*, vol. 525, pp. 1–28, 2023. DOI: <https://doi.org/10.1016/j.neucom.2022.12.031>
- [2] E. O. Belabbaci, M. Khammari, A. Chouchane, M. Bessaoudi, A. Ouamane, et al., "Enhancing Kinship Verification through Multiscale Retinex and Combined Deep-Shallow features," 2023 6th International Conference on Signal Processing and Information Security (ICSPIS), 2023, pp. 178–183. DOI: <https://doi.org/10.1109/ICSPIS60075.2023.10344115>
- [3] M. Almuashi, S. Z. Mohd Hashim, D. Mohamad, M. H. Alkawaz, and A. Ali, "Automated kinship verification and identification through human facial images: a survey," *Multimed. Tools Appl.*, vol. 76, no. 1, pp. 265–307, 2017. DOI: <https://doi.org/10.1007/s11042-015-3007-5>
- [4] N. Kohli, "Automatic Kinship Verification in Unconstrained Faces using Deep Learning," LCSEE, UVU, Morgantown, WV, 2019. DOI: <https://doi.org/10.33915/etd.3938>
- [5] G. O'Brien and K. Murphy, "Fingerprint patterns through genetics," *J. Emerg. Investig.*, vol. 2, no. December, pp. 1–5, 2020. DOI: <https://doi.org/10.59720/20-012>
- [6] R. George, N. S. B. Nora Afandi, S. N. H. B. Zainal Abidin, N. I. Binti Ishak, H. H. K. Soe, et al., "Inheritance pattern of lip prints among Malay population: A pilot study," *J. Forensic Leg. Med.*, vol. 39, pp. 156–160, Apr. 2016. DOI: <https://doi.org/10.1016/j.jflm.2016.01.021>
- [7] S. Loganadan, M. Dardjan, N. Murniati, F. Oscandar, Y. Malinda, et al., "Preliminary Research: Description of Lip Print Patterns in Children and Their Parents among Deutero-Malay Population in Indonesia," *Int. J. Dent.*, vol. 2019, pp. 1–6, Mar. 2019. DOI: <https://doi.org/10.1155/2019/7629146>
- [8] E. O. AIGBOGUN, Jr, C. P. IBEACHU, and A. M. LEMUEL, "Fingerprint pattern similarity: a family-based study using novel classification," *Anatomy*, vol. 13, no. 2, pp. 107–115, 2019. DOI: <https://doi.org/10.2399/ana.19.065>
- [9] S. Mala, V. Rathod, S. Pundir, and S. Dixit, "Pattern self-repetition of fingerprints, lip prints, and palatal rugae among three generations of family: A forensic approach to identify family hierarchy," *J. Forensic Dent. Sci.*, vol. 9, no. 1, pp. 15–19, 2017.
- [10] S. Hormann, M. Knoche, and G. Rigoll, "A Multi-Task Comparator Framework for Kinship Verification," in *Proceedings - 2020 15th IEEE International Conference on Automatic Face and Gesture Recognition, FG 2020*, 2020, pp. 863–867. DOI: <https://doi.org/10.1109/FG47880.2020.00106>
- [11] M. Xu, X. Zhang, and X. Zhou, "Confidence-Calibrated Face and Kinship Verification," *IEEE Trans. Inf. Forensics Secur.*, vol. 19, no. 8, pp. 372–384, 2024. DOI: <https://doi.org/10.1109/TIFS.2023.3318957>
- [12] J. Yu, M. Li, X. Hao, and G. Xie, "Deep Fusion Siamese Network for Automatic Kinship Verification," in *Proceedings - 2020 15th IEEE International Conference on Automatic Face and Gesture Recognition, FG 2020*, 2020, pp. 892–899. DOI: <https://doi.org/10.1109/FG47880.2020.00127>
- [13] O. Laiadi, A. Ouamane, A. Benakcha, A. Taleb-Ahmed, and A. Hadid, "Multi-view Deep Features for Robust Facial Kinship Verification," in *Proceedings - 2020 15th IEEE International Conference on Automatic Face and Gesture Recognition, FG 2020*, 2020, pp. 877–881. DOI: <https://doi.org/10.1109/FG47880.2020.00118>
- [14] X. Wu, X. Feng, X. Cao, X. Xu, D. Hu, et al., "Facial Kinship Verification: A Comprehensive Review and Outlook," *International Journal of Computer Vision*, vol. 130, no. 6, pp. 1494–1525, 2022. DOI: <https://doi.org/10.1007/s11263-022-01605-9>
- [15] N. Nader, F. E.-Z. El-Gamal, S. El-Sappagh, K. S. Kwak, and M. Elmoogy, "Kinship verification and recognition based on handcrafted and deep learning feature-based techniques," *PeerJ Comput. Sci.*, vol. 7, p. e735, Dec. 2021. DOI: <https://doi.org/10.7717/peerj-cs.735>
- [16] M. C. Mzoughi, N. Ben Aoun, and S. Naouali, "A review on kinship verification from facial information," *Vis. Comput.*, vol. 41, no. 3, pp. 1789–1809, Feb. 2025. DOI: <https://doi.org/10.1007/s00371-024-03493-1>
- [17] W. Wang, S. You, S. Karaoglu, and T. Gevers, "A survey on kinship verification," *Neurocomputing*,

- vol. 525, pp. 1–28, 2023. DOI: <https://doi.org/10.1016/j.neucom.2022.12.031>
- [18] M. Tan and Q. V. Le, “EfficientNet: Rethinking model scaling for convolutional neural networks,” 36th Int. Conf. Mach. Learn. ICML 2019, vol. 2019-June, pp. 10691–10700, 2019. DOI: <https://doi.org/10.48550/arXiv.1905.11946>
- [19] A. Zhou, Y. Ma, W. Ji, M. Zong, P. Yang, et al., “Multi-head attention-based two-stream EfficientNet for action recognition,” *Multimed. Syst.*, vol. 29, no. 2, pp. 487–498, 2023. DOI: <https://doi.org/10.1007/s00530-022-00961-3>
- [20] J. A. Alhijaj and R. S. Khudayer, “Integration of EfficientNetB0 and Machine Learning for Fingerprint Classification,” *Inform.*, vol. 47, no. 5, pp. 49–56, 2023. DOI: <https://doi.org/10.31449/inf.v47i5.4724>
- [21] P. Eko Niti Taruno, G. Satya Nugraha, R. Dwiyanaputra, and F. Bimantoro, “Monkeypox Classification based on Skin Images using CNN: EfficientNet-B0,” *E3S Web Conf.*, vol. 465, p. 02031, Dec. 2023. DOI: <https://doi.org/10.1051/e3sconf/202346502031>
- [22] V. H. B. Hadi, A. B. Mutiara and R. Refianti, “Implementation of Convolutional Neural Network with EfficientNet-B0 Architecture for Brain Tumor Classification,” 2023 Eighth International Conference on Informatics and Computing (ICIC), Manado, Indonesia, 2023, pp. 1–6. DOI: <https://doi.org/10.1109/ICIC60109.2023.10381979>
- [23] N. N. A. Zahrani and R. Hedjar, “Comparison Study Of Deep-Learning Architectures For Classification of Thoracic Pathology,” in 2022 13th International Conference on Information and Communication Systems (ICICS), 2022, pp. 192–198. DOI: <https://doi.org/10.1109/ICICS55353.2022.9811150>
- [24] A. Nandy, S. Haldar, S. Banerjee, and S. Mitra, “A survey on applications of siamese neural networks in computer vision,” in 2020 International Conference for Emerging Technology, INCET 2020, 2020, no. 3, pp. 13–17. DOI: <https://doi.org/10.1109/INCET49848.2020.9153977>
- [25] N. Sharma, S. Gupta, H. G. Mohamed, D. Anand, J. L. V. Mazón, et al., “Siamese Convolutional Neural Network-Based Twin Structure Model for Independent Offline Signature Verification,” *Sustain.*, vol. 14, no. 18, pp. 1–14, 2022. DOI: <https://doi.org/10.3390/su141811484>
- [26] H. Jansen, M. P. Gallee, and F. H. Schröder, “Analysis of Sonographic Pattern in Prostatic Cancer: Comparison of Longitudinal and Transversal Transrectal Ultrasound with Subsequent Radical Prostatectomy Specimens,” *Eur. Urol.*, vol. 18, no. 3, pp. 174–178, 1990. DOI: <https://doi.org/10.1159/000463903>
- [27] Y. Li, C. L. P. Chen, and T. Zhang, “A Survey on Siamese Network: Methodologies, Applications, and Opportunities,” *IEEE Trans. Artif. Intell.*, vol. 3, no. 6, pp. 994–1014, 2022. DOI: <https://doi.org/10.1109/TAI.2022.3207112>
- [28] S. Roy, M. Harandi, R. Nock, and R. Hartley, “Siamese Networks: The Tale of Two Manifolds,” in 2019 IEEE/CVF International Conference on Computer Vision (ICCV), 2019, vol. 2019-Octob, pp. 3046–3055. DOI: <https://doi.org/10.1109/ICCV.2019.00314>
- [29] N. Serrano and A. Bellogín, “Siamese neural networks in recommendation,” *Neural Comput. Appl.*, vol. 35, no. 19, pp. 13941–13953, 2023. DOI: <https://doi.org/10.1007/s00521-023-08610-0>
- [30] A. Nandy, S. Haldar, S. Banerjee, and S. Mitra, “A Survey on Applications of Siamese Neural Networks in Computer Vision,” no. 3, pp. 13–17, 2020. DOI: <https://doi.org/10.1109/INCET49848.2020.9153977>
- [31] K. Martin, N. Windunga, S. Sani, S. Massie, and J. Clos, “A convolutional siamese network for developing similarity knowledge in the SelfBACK dataset,” *CEUR Workshop Proc.*, vol. 2028, pp. 85–94, 2017.
- [32] F. Wang, X. Xiang, J. Cheng, and A. L. Yuille, “NormFace,” in Proceedings of the 25th ACM international conference on Multimedia, 2017, pp. 1041–1049. DOI: <https://doi.org/10.1145/3123266.3123359>
- [33] R. Annisa and B. Soewito, “M2FRED Analysis Using MobileNet and Siamese Neural Network,” *J. Adv. Inf. Technol.*, vol. 14, no. 6, pp. 1312–1320, 2023. DOI: <https://doi.org/10.12720/jait.14.6.1312-1320>
- [34] G. Figueroa-Mata and E. Mata-Montero, “Using a convolutional siamese network for image-based plant species identification with small datasets,” *Biomimetics*, vol. 5, no. 1, 2020. DOI: <https://doi.org/10.3390/biomimetics5010008>
- [35] C. -Y. Wu, R. Manmatha, A. J. Smola and P. Krähenbühl, “Sampling Matters in Deep Embedding Learning,” *Proc. IEEE Int. Conf. Comput. Vis.*, vol. 2017-Octob, pp. 2859–2867, 2017. DOI: <https://doi.org/10.1109/ICCV.2017.309>
- [36] A. Othmani, D. Han, X. Gao, R. Ye, and A. Hadid, “Kinship recognition from faces using deep learning with imbalanced data,” *Multimed. Tools Appl.*, vol. 82, no. 10, pp. 15859–15874, 2023. DOI: <https://doi.org/10.1007/s11042-022-14058-6>
- [37] A. Harisha, B. K. Prasad, K. Rajeev, Maithri, and Nishchal, “A performance evaluation of convolution neural networks for kinship discernment: An application in digital forensics,” *Intell. Decis. Technol.*, vol. 16, no. 2, pp. 379–386, 2022. DOI: <https://doi.org/10.3233/IDT-210132>
- [38] R. Rachmadi, I. Purnama, S. Nugroho, and Y. Suprpto, “Family-Aware Convolutional Neural Network for Image-based Kinship Verification,” *Int. J. Intell. Eng. Syst.*, vol. 13, no. 6, pp. 20–30, 2020. DOI: <https://doi.org/10.22266/ijies2020.1231.03>

- [39] L. Zhang, Q. Duan, D. Zhang, W. Jia, and X. Wang, "AdvKin: Adversarial Convolutional Network for Kinship Verification," *IEEE Trans. Cybern.*, vol. 51, no. 12, pp. 5883–5896, 2021. DOI: <https://doi.org/10.1109/TCYB.2019.2959403>
- [40] T. Navghare, A. Muley, and V. Jadhav, "Siamese Neural Networks for Kinship Prediction: A Deep Convolutional Neural Network Approach," *Indian J. Sci. Technol.*, vol. 17, no. 4, pp. 352–358, 2024. DOI: <https://doi.org/10.17485/IJST/v17i4.3018>
- [41] A. Rehman, Z. Khalid, Fawad, M. A. Asghar, and M. J. Khan, "Kinship verification using Deep Neural Network Models," in *RAEE 2019 - International Symposium on Recent Advances in Electrical Engineering*, 2019, pp. 5–9. DOI: <https://doi.org/10.1109/RAEE.2019.8886969>
- [42] C. Bisogni and F. Narducci, "Kinship recognition: how far are we from viable solutions in smart environments?," *Procedia Comput. Sci.*, vol. 198, no. 2018, pp. 225–230, 2022. DOI: <https://doi.org/10.1016/j.procs.2021.12.232>
- [43] J. Yu, G. Xie, X. Hao, Z. Cui, L. Zhang, et al., "Deep Kinship Verification and Retrieval Based on Fusion Siamese Neural Network," in *2021 16th IEEE International Conference on Automatic Face and Gesture Recognition (FG 2021)*, 2021, no. February, pp. 1–8. DOI: <https://doi.org/10.1109/FG52635.2021.9666938>
- [44] A. Abbas and M. Shoaib, "Kinship identification using age transformation and Siamese network," *PeerJ Comput. Sci.*, vol. 8, p. e987, Jun. 2022. DOI: <https://doi.org/10.7717/peerj-cs.987>
- [45] D. Li and X. Jiang, "Kinship Verification Based on Global and Local Attention Mechanism," *Acad. J. Sci. Technol.*, vol. 5, no. 1, p. 2023, 2023.
- [46] J. Yu, G. Xie, M. Li, and X. Hao, "Retrieval of Family Members Using Siamese Neural Network," in *Proceedings - 2020 15th IEEE International Conference on Automatic Face and Gesture Recognition, FG 2020*, 2020, pp. 882–886. DOI: <https://doi.org/10.1109/FG47880.2020.00136>
- [47] S. Ibrahim Fathi and M. H. Aziz, "A Dataset for Kinship Estimation from Image of Hand Using Machine Learning," *Iraqi J. Electr. Electron. Eng.*, vol. 20, no. 2, pp. 127–136, 2024. DOI: <https://doi.org/10.37917/ijeee.20.2.11>
- [48] S. Fathi and M. Aziz, "Kinship Detection Based on Hand Geometry Using ResNet50 Model for Feature Extraction," *Al-Rafidain Engineering Journal (AREJ)*, vol. 29, no. 1, pp. 118–125, 2024. DOI: <https://doi.org/10.33899/arej.2024.145748.1321>
- [49] G. Dvorsak, A. Dwivedi, V. Struc, P. Peer, and Z. Emersic, "Kinship Verification from Ear Images: An Explorative Study with Deep Learning Models," in *2022 International Workshop on Biometrics and Forensics (IWBF)*, 2022, pp. 1–6. DOI: <https://doi.org/10.1109/IWBF55382.2022.9794555>
- [50] M. Thanoon and S. Dawwd, "DCNN For Cataract Disease Detection Based on Model Parallelism," *Al-Rafidain Engineering Journal (AREJ)*, vol. 29, no. 2, pp. 111–118, 2024. DOI: <https://doi.org/10.33899/arej.2024.147115.1338>
- [51] J. Yosinski, J. Clune, Y. Bengio, and H. Lipson, "How transferable are features in deep neural networks?," *Adv. Neural Inf. Process. Syst.*, vol. 4, no. January, pp. 3320–3328, 2014. DOI: <https://doi.org/10.48550/arXiv.1411.1792>
- [52] T. Chen, S. Kornblith, M. Norouzi, and G. Hinton, "A simple framework for contrastive learning of visual representations," *37th Int. Conf. Mach. Learn. ICML 2020*, vol. PartF16814, no. Figure 1, pp. 1575–1585, 2020. DOI: <https://doi.org/10.48550/arXiv.2002.05709>

الشبكات العصبية السيامية والتعلم بالانتقال للتحقق من القرابة من صور بشرة راحة اليد

مازن هاشم عزيز

قسم هندسة الحاسوب، كلية الهندسة، جامعة الموصل، الموصل، العراق

mazin.hazizi@uomosul.edu.iq

تاريخ القبول: 28 يوليو 2025

استلم بصيغته المنقحة: 2 يوليو 2025

تاريخ الاستلام: 26 ابريل 2025

الملخص

يُعد التحقق من القرابة مجالاً بحثياً بالغ الأهمية نظراً لتطبيقاته المتنوعة، بما في ذلك اختبارات الأبوة، ولم شمل الأسرة، والتحقيقات الجنائية. في حين كان تحليل الحمض النووي هو الأسلوب السائد، لا تزال تقنيات الذكاء الاصطناعي قيد الاستكشاف والاختبار. وقد حظي التحقق من القرابة عن طريق سمات الوجه، الذي يتضمن مقارنة السمات بين صورتين للوجه، باهتمام بحثي كبير. تقدم هذه الورقة نهجاً جديداً للتحقق من القرابة باستخدام صور راحة اليد. استُخدم نموذج EfficientNetB0 لاستخراج السمات العميقة من خلال التعلم الانتقالي. ووظفت بنية الشبكة العصبية السيامية لتقييم التشابه. أُجريت سيناريوهات تجريبية مختلفة تتعلق ببنية الشبكة، ومعايير التدريب، والضبط الدقيق. واستُخدمت مجموعة بيانات صور يد قرابة الموصل (MKH) لإنشاء مجموعة بيانات صور جلد راحة اليد، والتي تتكون من 7332 زوجاً مقسمة بالتساوي إلى فئات ذات صلة وغير ذات صلة. كانت النتائج واعدة، حيث حققت دقة تحقق تبلغ حوالي 99% وكانت النتائج واعدة، حيث حققت دقة تحقق تبلغ حوالي 99%، ومتوسط وقت استدلال 77.02 مللي ثانية لكل زوج من الصور باستخدام تقنية تحليل المكونات الرئيسية (PCA) بعد التدريب.

الكلمات الدالة:

التحقق من القرابة، الشبكة العصبية السيامية، التعلم بالانتقال العميق، EffecientNetB0، بشرة كف اليد.