Received: October 23, 2024. Revised: December 1, 2024. 832

International Journal of
Intelligent Engineering & Systems

http://www.inass.org/

SSCNetViT: A Hybrid Siamese Sequential Classification-ViT Model for Kinship Recognition in Indonesia Facial Microexpression

Ike Fibriani[1,2] Eko Mulyanto Yuniarno[1,3] Ronny Mardiyanto[1]
Mauridhi Hery Purnomo[1,3,4]*

[1]*Department of Electrical Engineering, Sepuluh Nopember Institute of Technology, Surabaya, Indonesia*
[2]*Department of Electrical Engineering, University of Jember, Jember, Indonesia*
[3]*Department of Computer Engineering, Sepuluh Nopember Institute of Technology, Surabaya, Indonesia*
[4]*University Center of Excellence on Artificial Intelligence for Healthcare and Society (UCE AIHeS), Surabaya, Indonesia*
* Corresponding author's Email: hery@ee.its.ac.id

Abstract: Kinship recognition based on facial appearance is a challenging task with its unique complexity. Analyzing facial dynamics allows to assess the degree of similarity between individuals in the context of kinship. Although significant research has been done on basic microexpressions, there is currently a surge of interest in recognizing combined facial expressions of emotions in the field of image processing. The main focus of this study is to recognize kinship through a microexpression approach by proposing a hybrid model of Siamese Sequential Classification Network (SSCN) and vision transformer as a feature extractor instead of traditional convolution. This model combines the advantages of SSCN and Vision Transformer (ViT) models, the authors call it SSCNetViT, not only can capture global context information and present stronger learning ability, but also introduce SSCN inductive bias to improve generalization performance. The model is tested on independently collected datasets from the local Indonesian dataset (LaIndo) and the Families in the Wild (FIW) dataset. The results show that the L32 backbone achieves the highest average accuracy of 90.07%, with the peak performance in the BB class (99.5%) and the lowest in the FD class (84.0%). In comparison, the B16 and B32 backbones yield lower average accuracies of 88.0% and 83.3%, respectively, highlighting the effectiveness of the approach for kinship verification. Thus, our proposed SSCNetViT model with B16 quadratic feature fusion and multiplicative fusion strategies achieves the best performance and achieves better accuracy that outperforms previous state-of-the-art (SOTA) studies.

Keywords: Vision transformers, Siamese sequential classification network, Feature fusion, Kinship recognition, Micro-expressions.

1. Introduction

Artificial intelligence is advancing rapidly, particularly with the internet's influence to find datasets. In the fields of computer vision and biometrics, there is significant focus on facial image analysis and modelling. One area of research that draws attention is kinship verification, which involves identifying family relationships through facial image analysis [1]. Currently, kinship recognition relies heavily on identifying similarities in specific facial features, such as nose shape, lip curvature, or eye structure [2, 3]. Kinship relationships can be determined by comparing these facial characteristics between two input images. Additionally, facial expressions captured in the images are also used for kinship detection [4]. Face expression is crucial in human communication and can be classified based on intensity, duration, and significance into micro-expressions and macro-expressions [5, 6]. In kinship verification, many advanced automatic methods have been developed, some of which demonstrate superior performance compared to human capabilities [7, 8]. Previous research has primarily focused on utilizing facial

Table 1. Previous Research Problem Correlated to Our Research

Research	Problems	Our Contribution
Robinson et al. [31]	Large-scale kinship datasets	• Developed a deep learning architecture named the Deep Siamese Fusion Network (SSCNetViT) specifically for kinship recognition to enhance classification accuracy in familial relationships. • Employed a Vision Transformer (ViT) as the core feature extractor in place of conventional convolutional neural networks (CNNs). ViT's capability to capture global dependencies in images through self-attention mechanisms makes it particularly effective in handling the complexity of facial micro-expressions. • Focused the research on analyzing micro-expressions, subtle and involuntary facial movements, from Indonesian faces. This focus aims to improve the model's accuracy in detecting kinship by recognizing these nuanced expressions, which are often overlooked in traditional facial recognition methods. • Collected an independent dataset of Indonesian micro-expression
Othmani et al. [15]	The incorporation of microexpressions in determining kinship classes has not been explored	
Yu et al. [16]		
Yan et al. [6]		
Yan et al. [6]	Microexpressions not used for kinship classification; imbalance in sample sizes across kinship classes	
Huang et al. [17]	Lower accuracy (55%-59%) compared to FIW dataset; sample imbalance across different kinship classes	
Li et al. [18]		
Huang et al. [8]		
Bisogni and Narducci [19]	Misclassification issues: mothers as fathers (31.25%), fathers as mothers (25%), daughters as sons (25%), sons as daughters (43.75%)	
Dosovitskiy et al. [21]	CNNs are not necessary for image classification; ViT outperforms CNNs while requiring fewer computational resources	
Phan et al. [22]	Face identification struggles with out-of-distribution data, slow inference times, and lacks interpretability in decision-making.	
Liao et al. [23]	Face anti-spoofing models struggle with generalization across domains, face challenges in identifying attack samples, and often require large, resource	

macro-expressions to determine kinship between individuals, involving the segmentation of input images into different parts that effectively represent human facial expressions [9]. Previous research has mainly focused on utilizing facial macro expressions to ascertain kinship between individuals. This involves dividing the input image into different parts that effectively represent human facial expressions.

Despite advancements in kinship verification, utilization of micro-expression in images to identify kinship relationships remains underexplored. Numerous automated methods for kinship recognition have been developed, some surpassing human accuracy [10,11]. Prior studies have mainly focused on using facial micro-expressions to determine kinship by dividing the input images into distinct areas representing human expressions [12]. However, the specific application of micro-expressions to identify familial relationships in images has not been adequately addressed. This study aims to fill this gap by collecting image data centered on micro-expression features to detect familial ties and enhance facial biometric analysis.

Originally developed for natural language processing, transformers have proven effective in visual tasks like image recognition [13,14], object detection [3], semantic segmentation [5], and image generation [5]. The self-attention mechanism effectively identifies important features within images, positioning transformers as a strong alternative to convolutional neural networks (CNNs).

Table 1 details the LaIndo dataset and related Hybrid CNN-ViT models. Derived from video recordings of individuals with defined kinship ties, the dataset emphasizes the untapped potential of micro-expressions in kinship recognition. This study introduces a hybrid model that combines CNNs' inductive bias with ViTs' global attention, advancing microexpression-based kinship recognition.

This study presents SSCNetViT, a deep learning model developed to recognize kinship by analyzing microexpressions specifically in Indonesian facial images. Given that the FIW dataset predominantly features individuals from Western backgrounds, Indonesian facial data was incorporated to improve diversity and enhance the model's performance evaluation. The model integrates a Vision Transformer (ViT) with a Deep Fusion Siamese Network to boost accuracy in kinship recognition [13]. Previous approaches typically relied on Convolutional Neural Networks (CNNs) to efficiently extract features from images [14]. In contrast, Vision Transformers (ViT) enhance classification by reducing data dimensionality. The extracted features undergo processing in a feature

fusion module, which merges them to identify kinship relationships. Various operations, including multiplication, feature distance, and squared subtraction, were performed prior to feature integration. To evaluate the classifier's effectiveness in combining ViT and feature fusion, both LaIndo datasets and the FIW dataset were employed. The main contributions of this study are:

1. A novel approach is proposed for developing an effective method for fusion of multiple siamese network features for face recognition. The selected dataset has imbalance issues that create bias during model training. Currently, there are no existing studies in this area, making this study unique and valuable in its examination of the relationship between micro-expressions and kinship.
2. We design a novel framework, Hybrid CNN and ViT then we named it CNNetViT which aims to address the challenges of computer-based kinship recognition and other biometric techniques which effectively overcomes the challenges in kinship recognition and produces excellent results.
3. A comprehensive model that modifies the classifier model by combining multiple features yields better accuracy and precision than other feature extractors of the proposed feature fusion is refined using transfer learning method.

The paper is structured as follows: Section 2 provides details about the materials and methods used; Section 3 describes the experiments conducted; Section 4 presents the results obtained and discusses their implications; and Section 5 summarizes the findings and outlines future research directions.

2. Materials and methods

We propose a deep relational network model that uses a micro-expression transformation step to preprocess two facial images before comparing them for kinship. The images are adjusted to the same

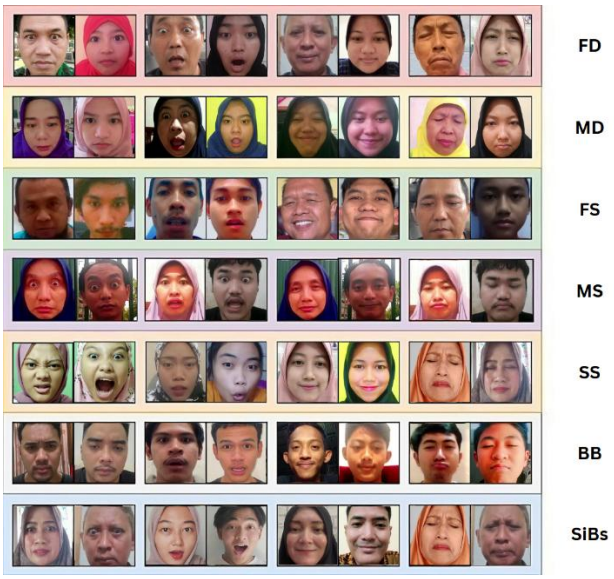

Figure. 1 LaIndo face images dataset depicting various kinship relationships that were independently

Table 1. Distribution Of Each Kinship Class

Kinship	Local Dataset	FIW Dataset
Mother-Daughter (MD)	155	736
Mother-Son (MS)	85	716
Father-Daughter (FD)	60	712
Father-Son (FS)	110	721
Sister-Sister (SS)	95	1029
Brother-Brother (BB)	70	991
Sister-Brother (SiBs)	60	1588

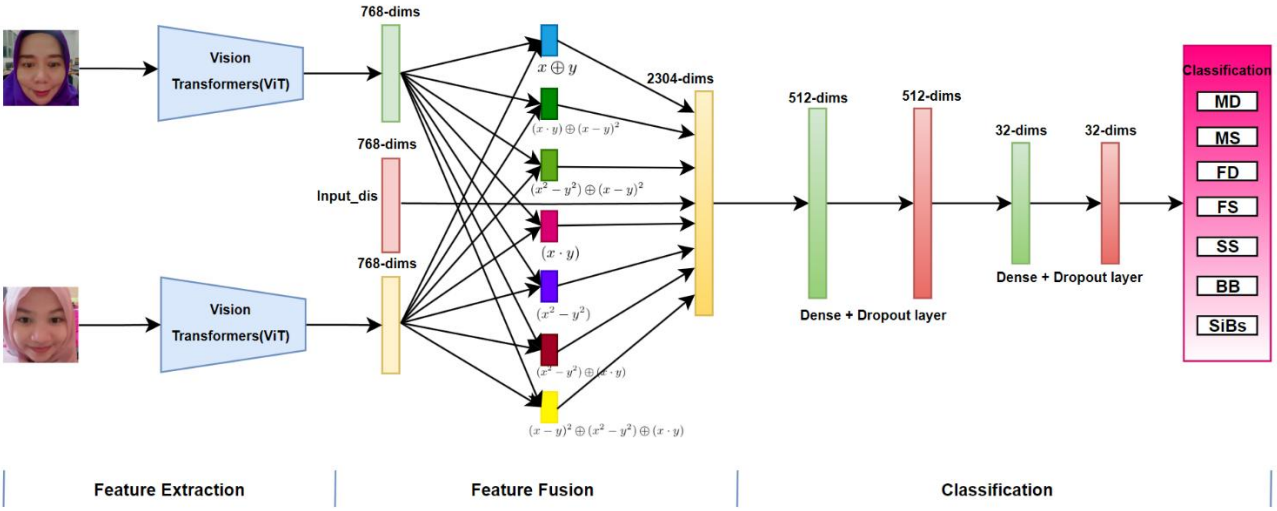

Figure. 2 The overview of the model architecture for kinship recognition

micro-expression state, and then a Siamese network with ViTs extracts their features. The Euclidean distance between these features is calculated, followed by a feature fusion process, which improves accuracy and speed in kinship classification [13, 24]. SSCNetViT specifically utilizes microexpressions as inputs to identify relationships between images, focusing on kinship. The model expands its focus to include various microexpressions, not just basic ones like smiling, helping to capture subtle kinship features. A time-distributed method processes 10 image frames simultaneously using a ViT pre-trained on ImageNet [25]. Transfer learning with ImageNet-pretrained weights was used to speed up feature extraction and improve kinship recognition between image pairs [26].

2.1. Dataset

This research utilized not only the publicly available FIW dataset [16] but also incorporated an inclusion of this LaIndo dataset aimed to improve its quality and relevance. The LaIndo dataset was collected through video recordings and capturing micro-expressions of individuals with defined kinship relationships, ensuring to minimize errors in facial expression recognition and kinship recognition.

Table 2 presents LaIndo dataset featuring family images representing various kinship relationships, including mother-daughter (MD), mother-son (MS), father-daughter (FD), father-son (FS), brother-brother (BB), sister-sister (SS), and sister-brother (SiBs). This diverse dataset allows for analysis of kinship recognition methods across different familial connections. It also addresses the kinship class imbalance, particularly in the SiBs and BB categories, seen in both the LaIndo and FIW datasets. Fig. 1 depicts the LaIndo dataset, which encompasses facial images of individuals of Indonesian descent. Furthermore, the LaIndo dataset includes microexpressions, which are a crucial aspect of this study. Images were normalized for size and format, then randomized to reduce bias. The dataset was split into training and validation sets, with the validation set for performance assessment and parameter tuning.

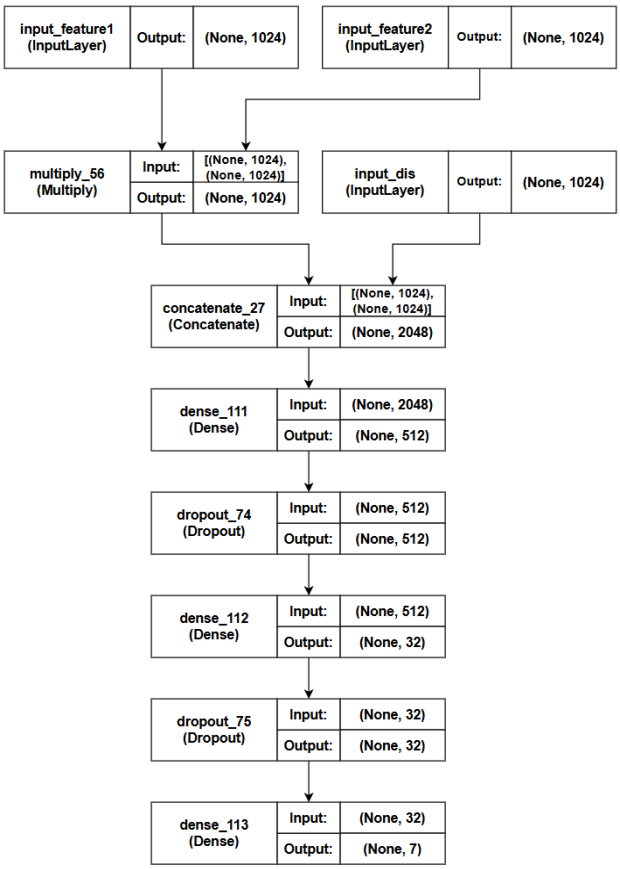

Figure. 3 The model architecture illustrates one of seven feature fusion examples, in which two input features and an additional input are combined through multiplication and concatenation

2.2. Proposed framework

2.2.1. Vision transformer with transfer learning

The feature extraction process utilised a transfer learning algorithm based on the pretrained Vision Transformer (ViT) model, trained on the ImageNet dataset. ViT, a deep learning model designed for image classification, employs the Transformer architecture. It partitions an image into fixed-size patches, embeds each patch, adds positional embeddings, and inputs the resulting sequence of vectors into a standard Transformer encoder. The encoder consists of multiple blocks with three key components: Layer Normalisation, a Multi-head Attention Network (MSP), and a Feedforward Network. ViT is extensively applied in image recognition tasks such as object segmentation, image classification, and action recognition. It is also utilised in generative modeling and multimodal applications, including visual grounding, visual question answering, and visual reasoning [27, 21]. Within ViT, images are transformed into embedded representations derived from input image patches.

$$z_0 = [x_{class}; x_p^1 E; x_p^2 E; \ldots; x_p^n E] + E_{pos}, \quad E \epsilon R^{(P^2 \cdot C) \times D}, E \epsilon R^{(N+1) \times D} \tag{1}$$

The formula uses Z_0 as the initial embedding vector, Xclass for data classification, and $X^1{}_p$ for positional information. The embedding matrix E maps data to a D-dimensional feature space, with

EER(P²CxD) for spatial features. Epos adds positional embeddings to preserve order or context.

Eq. (1) illustrates how a sequence is formed by concatenating the class token with the linearly embedded patches and adding positional embeddings. dataset. The essence was to classify kinship using a pair of input models through the SNN [30, 31].

$$z_l' = MSA(LM(z_l - 1)) \quad (2)$$

Where $z_l - 1$ (the output from the previous layer) is first processed by a linear transformation or layer normalization (LM) and then passed through a Multi-Head Self-Attention (MSA) mechanism. This approach allows the model to integrate information from all positions in the input sequence, capturing dependencies between elements and enhancing contextual understanding at each layer.

The Eq. (2) represents the application MSA mechanism in one of the transformer layers. MSA network allows the model to attend to different parts of image sequences in parallel and looking for relationships between both sequences.

Both equations are the main building block to create a vision transformer model. Images transformation in Eq. (1) and MSA mechanism in Eq. (2) extracts meaningful features from the image.

2.2.2. Siamese neural network

A Siamese Neural Network (SNN) is a specific kind of artificial neural network architecture that is designed to compare two inputs by processing them through two identical subnetworks that share weights. In contrast to a conventional classification task, an SNN is employed to ascertain the degree of similarity or relationship between two inputs. Each subnetwork within an SNN extracts pertinent features from a single input, utilising layers such as convolutional layers (in image-based tasks). The outputs of these subnetworks are subsequently evaluated with a distance measurement (e.g., Euclidean or cosine similarity). This enables the SNN to discern whether the inputs are analogous or disparate. The architecture of SNN can be broken down as follows:

a) Convolution Layer

In a siamese neural network (SNN), each input is passed through two identical subnetworks that share the same weights. These subnetworks utilize convolutional layers as feature extractors, particularly when processing image inputs, to automatically learn spatial hierarchies of features. The convolutional layer functions by applying convolutional filters, also referred to as kernels, to the input data, which may be, for example, images. These filters traverse the input, performing an element-wise multiplication between the filter and the portion of the input it covers, thereby producing a feature map.

The output of the convolutional layer for each subnetwork can be conceptualized as a transformed representation of the input image, with a particular emphasis on specific features such as edges, textures, or patterns. In an SNN, the convolutional layers in both subnetworks utilize the same set of weights to ensure that the feature extraction process is consistent across both inputs. The weight-sharing mechanism is of critical importance, as it enables the network to effectively compare the features of the two inputs and ensures that the same type of feature extraction occurs in both cases. The operation performed in a convolutional layer can be mathematically expressed as:

$$Output = \frac{n+2p-t}{d} + 1 \quad (3)$$

Where n is the size of the input feature map, p is the padding applied to the input, t is the size of the convolutional filter, d is the stride, which controls how the filter moves across the input.

b) Dropout Layer

The dropout layer performs a similar function to that observed in Convolutional Neural Network (CNN), whereby it prevents overfitting by randomly dropping (setting to zero) some of the neuron activations during training. In a Siamese Neural Network (SNN), dropout layers are applied to both subnetworks with the objective of improving generalization. The process of dropout is expressed as:

$$Output_{dropout} = Input \odot Mask \quad (4)$$

Where Output is the output of the dropout layer after applying dropout, Input is the input tensor to the dropout layer, represents element-wise multiplication, and Mask is a binary mask that randomly sets certain elements of the input tensor to 0 during training.

c) Concatenate Layer

The Concatenate Layer in an SNN is used to merge different input tensors or feature maps into a single tensor. This is particularly important when combining the outputs of different operations, such as element-wise differences (subtraction) and multiplications (similarity measures) between the two input feature vectors in SNN.

d) Dense Layer

A dense layer (also called a fully connected layer) plays a crucial role in learning non-linear

relationships between the extracted features after the convolutional or other feature extraction layers. A dense layer takes the output from the previous layers (whether it's a convolutional, subtraction or concatenation output) and fully connects every neuron in that layer to every neuron in the previous layer. This connection allows the network to capture complex relationships and dependencies between features, even if they are spatially distant or abstract. The mathematical operation for a dense layer is shown as:

$$Output_{FC} = Activation \left(\sum_{i=1}^{n} Input_i \times Weights_i + Biases\right) \quad (5)$$

Where Output is the output of the fully connected layer, Input is the input feature to the layer, Weights are the learnable parameters that connect each input to the neurons in the dense layer, Biases are the learnable offset terms for each neuron, and activation is the activation function applied element-wise to the output.

2.2.3. SSCNetViT

Fig. 4 shows the flowchart of the SSCNetViT framework is as follows:

1. Read FIW-LaIndo dataset.
2. Perform data preparation: Data cleaning and data normalization.
3. Split data into training and validation.
4. Train data using SSCNetViT.
5. Classification performance results are evaluated using Confusion Matrix.
6. Result of accuracy value.

The proposed SSCNetViT model classifies two images to determine their familial relationship using a ViT pre-trained model as a feature extractor for discriminative features. The output is a feature vector representing microexpression images. Features from both images are extracted by an SNN model and analyzed in the classification model. Two facial images (Input A and B) are fed into identical ViT backbones, generating 768-dimensional feature vectors. The Euclidean distance is then used to compute the feature distance between the kinship.

$$d = \sqrt{\sum_{i=1}^{n} (A_i - B_i)^2} \quad (6)$$

pair:

Which *d* is the Euclidean distance, thus A and B are two different input location in the latent space of features.

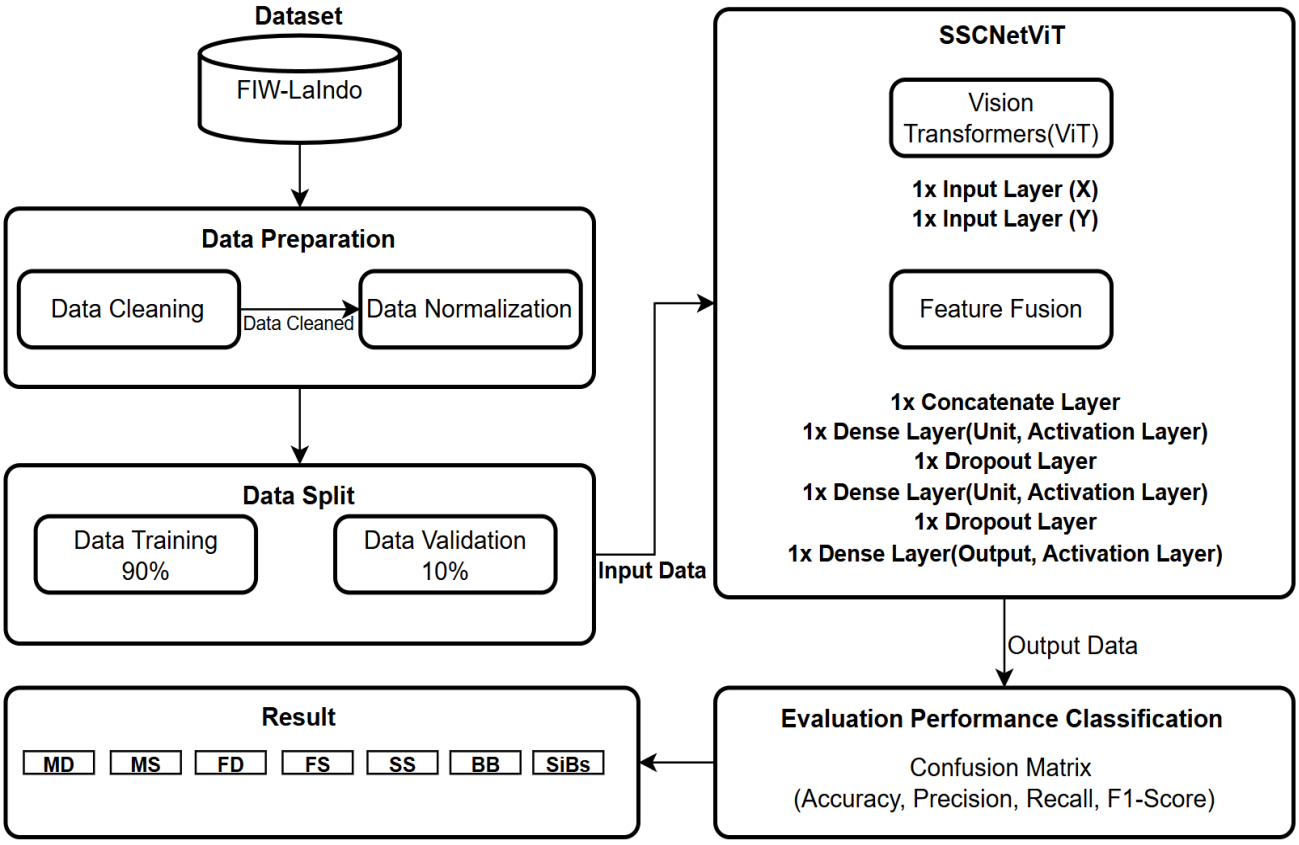

Figure. 4 SSCNetViT Framework Flowchart

The vectors undergo element-wise addition, multiplication, and squared differences, with seven combinations of these operations (Fig. 2) fused into a single 2304-dimensional vector. This is processed through fully connected (FC) layers with dropout, reducing the dimensionality to 512, 32, and ultimately a 32-dimensional vector for classifying kinship relationships like Mother-Daughter and Father-Son. The model compares microexpressions in the input images to determine kinship. The variations ViT B16, B-32, L-16, and L-32 represent different ViT configurations for image processing using a transformer-based architecture.

1) ViT B-16: This base model divides images into 16 patches and processes them through transformer layers to extract features and understand visual data.
2) ViT B-32: Similar to B-16, but with 32 patches, allowing for a more detailed representation of the input image.
3) ViT L-16: A larger architecture ("Large") using 16 patches, capable of capturing more complex patterns for tasks requiring higher-level image understanding.
4) ViT L-32: Like L-16, but with 32 patches, this model captures more intricate details, making it suitable for tasks needing comprehensive visual understanding.

Each model balances computational efficiency with expressive power, allowing selection based on task requirements.

2.2.4. Classification performance

To assess our model's performance, we employed various classification metrics to evaluate its effectiveness on unseen data. Among these, the confusion matrix was particularly important, offering a detailed breakdown of correct and incorrect predictions for each class.

The confusion matrix is a key tool for evaluating classification models, especially in multi-class scenarios. It provides a summary of the classifier's performance by comparing predicted labels against actual labels, helping to identify areas of strength and improvement. The Confusion Matrix provides the following information:

a) True Positive (TP)

Correct predictions where the model correctly classifies a positive sample.

b) True Negative (TN)

Correct predictions where the model correctly classifies a negative sample

c) False Positive (FP)

Incorrect predictions where the model incorrectly classifies a negative sample as positive

d) False Negative (FN)

Incorrect predictions where the model fails to classify a positive sample

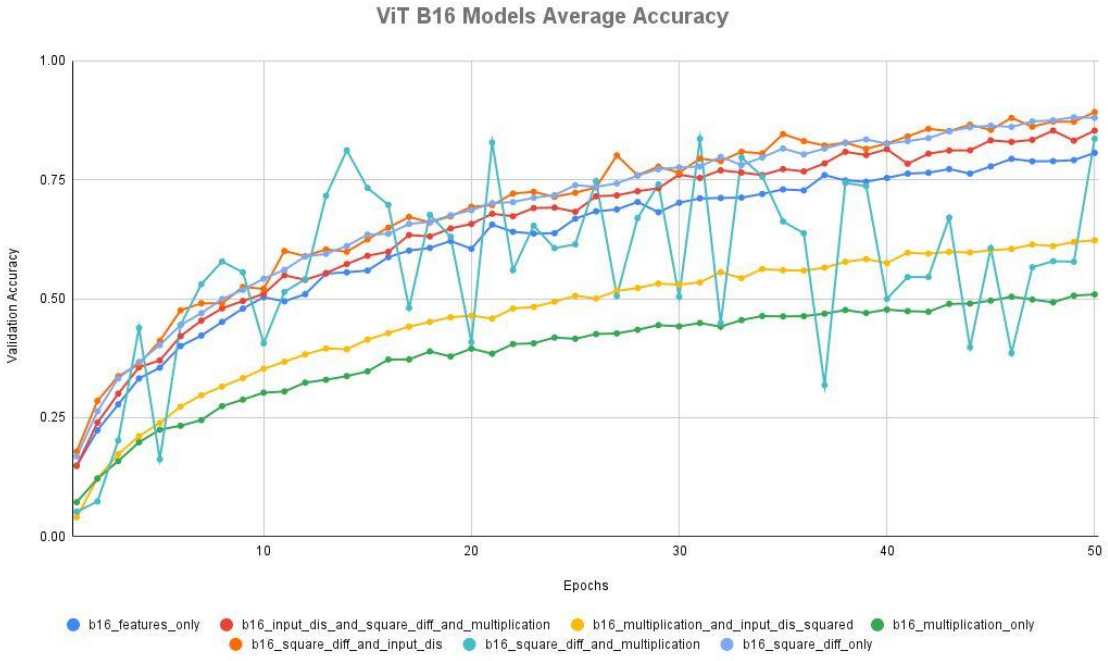

Figure. 5 The validation accuracy trends of different ViT B16 models over 50 epochs, using various feature fusion strategies

Table 2. The comparison of the performance of different backbones (B16, B32, L16, L32) with various feature fusion methods across several metrics

Back bone	Feature Fusion	BB	FD	FS	MD	MS	SiBs	SS	Average
B16	x $\oplus$ y	0.9448	0.9171	0.8718	0.8608	0.7434	0.7221	0.5845	0.8064
B16	$(x-y)^2$ $\oplus (x^2-y^2)$ $\oplus (x \cdot y)$	0.9339	0.9279	0.9194	0.9138	0.8165	0.7552	0.7062	0.8533
B16	$(x \cdot y) \oplus (x-y)^2$	0.717	0.6629	0.6623	0.6009	0.5838	0.568	0.5625	0.6225
B16	$(x \cdot y)$	0.6682	0.5826	0.5769	0.5106	0.4544	0.429	0.3415	0.509
B16	(x^2-y^2) $\oplus (x-y)^2$	0.9736	0.9666	0.9661	0.9502	0.8806	0.789	0.7187	0.8921
B16	(x^2-y^2) $\oplus (x \cdot y)$	0.9223	0.9125	0.9021	0.862	0.8201	0.7333	0.7009	0.8362
B16	(x^2-y^2)	0.9595	0.9553	0.9403	0.9333	0.8771	0.7699	0.7266	0.8803
B32	$x \oplus y$	0.8726	0.8556	0.8227	0.8172	0.6584	0.6247	0.5515	0.7432
B32	$(x-y)^2$ $\oplus (x^2-y^2)$ $\oplus (x \cdot y)$	1	0	0	0	0	0	0	0.1429
B32	$(x \cdot y) \oplus (x-y)^2$	1	0	0	0	0	0	0	0.1429
B32	$(x \cdot y)$	1	0	0	0	0	0	0	0.1429
B32	(x^2-y^2) $\oplus (x-y)^2$	0.9568	0.9432	0.9163	0.8948	0.832	0.7319	0.73	0.8579
B32	(x^2-y^2) $\oplus (x \cdot y)$	1	0	0	0	0	0	0	0.1429
B32	(x^2 − y^2)	0.9206	0.9199	0.8743	0.8577	0.8572	0.7618	0.6377	0.8327
L16	$x \oplus y$	0.8177	0.7644	0.6673	0.6598	0.5907	0.5867	0.5505	0.6624
L16	$(x-y)^2$ $\oplus (x^2-y^2)$ $\oplus (x \cdot y)$	0.8734	0.8516	0.8149	0.723	0.6967	0.6581	0.5313	0.7356
L16	$(x \cdot y) \oplus (x-y)^2$	0.6716	0.6251	0.5592	0.5348	0.4446	0.4393	0.3699	0.5206
L16	$(x \cdot y)$	0.6654	0.5212	0.5179	0.4566	0.4351	0.2357	0.2227	0.4364
L16	(x^2-y^2) $\oplus (x-y)^2$	0.9304	0.9269	0.8775	0.8694	0.8466	0.7989	0.5984	0.8354
L16	(x^2-y^2) $\oplus (x \cdot y)$	0.8393	0.8311	0.7556	0.726	0.6839	0.6594	0.472	0.7096
L16	(x^2-y^2)	0.8717	0.8679	0.8252	0.8004	0.7712	0.706	0.6405	0.7833
L32	$x \oplus y$	0.9962	0.9953	0.9725	0.9618	0.8431	0.7735	0.766	0.9012
L32	$(x-y)^2$ $\oplus (x^2-y^2)$ $\oplus (x \cdot y)$	0.9916	0.9884	0.9879	0.9774	0.9205	0.7965	0.6547	0.9024
L32	$(x \cdot y) \oplus (x-y)^2$	0.9619	0.9532	0.9479	0.9333	0.8777	0.7789	0.7214	0.882
L32	$(x \cdot y)$	0.8869	0.8469	0.8083	0.8047	0.752	0.7383	0.4981	0.7622
L32	(x^2-y^2) $\oplus (x-y)^2$	0.978	0.9747	0.9735	0.9734	0.8958	0.7659	0.7405	0.9003
L32	$\boldsymbol{(x^2-y^2)}$ $\boldsymbol{\oplus (x \cdot y)}$	**0.9941**	**0.9937**	**0.9894**	**0.9856**	**0.8579**	**0.8319**	**0.6967**	**0.907**

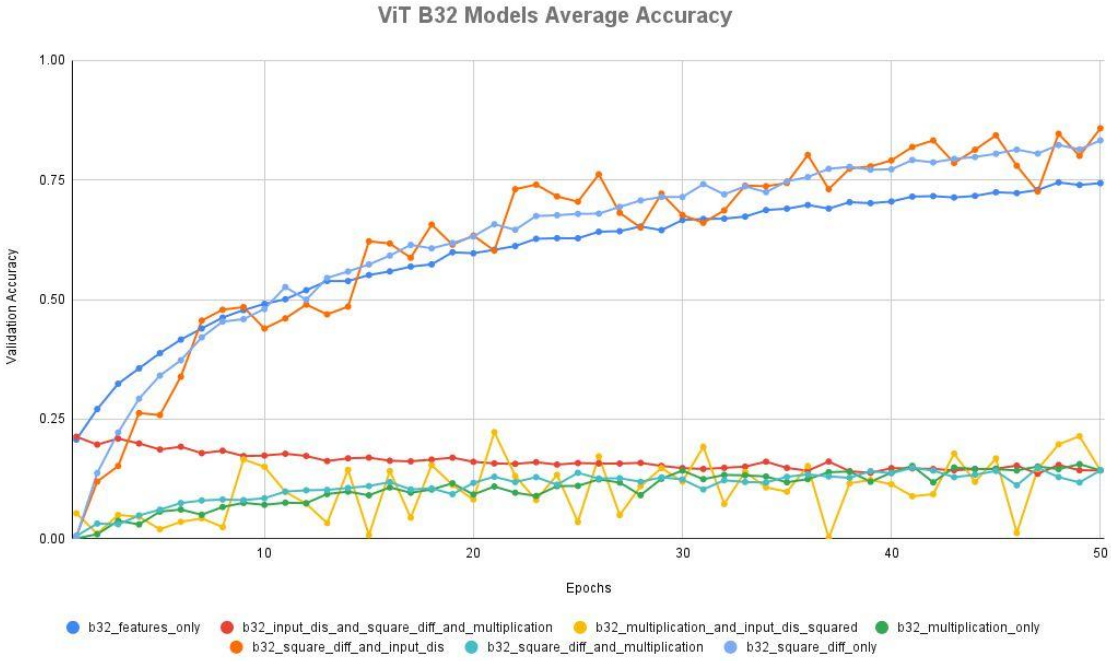

Figure. 6 The validation accuracy trends of different ViT B32 models over 50 epochs, using various feature fusion strategies

Table 3. Comparison results in previous research.

	Model	**Dataset**	**Accuracy (%)**
Dosovitskiy, et al. [21]	STLB-IP	CASME2	59.51
Robinson, et al. [31]	STCLQP	CASME2	56.10
J. Coe and M. Atay [20]	3D-FCNN	SMIC CASME	55.49
J. Lunter [28]	Deep and handcrafted feature	FIW	71.00
Rahmadi, et al. [30]	FA-CNN	RFIW'17	72.39
A. Shadrikov [32]	Vuvko	FIW	78.00
Luo, et al. [33]	DeepBlueAI	FIW	76.00
Yu, et al. [16]	Ustc-nelslip	FIW	76.00
Hörmann, et al. [34]	Stefhoer	FIW	74.00
L32 Backbone (Best Proposed Method)		**FIW-LaIndo**	**90.07**

Using the confusion matrix, there are several evaluation metrics that can be calculated:

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \quad (7)$$

- Precision represents a proportion of correctly identified positive samples out of all samples predicted to be positive. Precision is calculated using the following formula:

$$Precision = \frac{TP}{TP+FP} \quad (8)$$

- Recall refers to the proportion of correctly classified positive samples out of the total number of actual positives, with equation:

$$Recall = \frac{TP}{TP+FN} \quad (9)$$

- The F1-score is the harmonic mean of precision and recall, offering a balanced metric that considers both false positives and false negatives. It is especially valuable for imbalanced datasets, as it emphasises the lower value between precision and recall. The F1-score ranges from 0 to 1, with higher values indicating a more accurate and dependable model. A higher F1-score indicates that the model performs well in correctly identifying positive samples while minimizing false positives and false negatives:

$$F1-score = \frac{2\times(Recall\times Precision)}{Recall+Precision} \quad (10)$$

3. Experimental analysis

This section delineates the methodology employed to conduct the experiment and provides a comprehensive analysis of the results, thereby demonstrating the efficacy of the proposed model.

The model incorporates different parameter configurations based on its feature fusion variations, ranging from 41.3 million to 716.2 million parameters. Training and evaluation were performed on an NVIDIA DGX HPC equipped with an H100 GPU, offering 60 teraflops of floating-point processing power. This section presents a discussion of the datasets used, the experimental scenarios, and the results achieved.

3.1. Performance evaluation of feature methods

SSCNetViT is designed for similarity learning tasks, like kinship verification. Unlike traditional classifiers, SSCNetViT process paired inputs simultaneously, learning a distance metric to differentiate related from unrelated pairs. This is ideal for kinship verification, where two facial images are compared to assess familial relationships. Given two input images, x_1 and x_2, each is processed through identical, weight-sharing subnetworks denoted as $f(x_1)$ and $f(x_2)$. The resulting feature embeddings are compared using a distance metric, such as Euclidean distance or cosine similarity, to measure their similarity. In this context, the Euclidean distance between the feature vectors is mathematically represented in function $d(x_1,x_2)$ such as:

$$d(x_1, x_2) = ||\Delta f||_2 \tag{11}$$

where $\Delta f = (x_1 - x_2) = f(x_1) - f(x_2)$ denotes the L2 norm. The objective of the SSCNetViT network is to minimize this distance for image pairs that share a kinship relationship (positive pairs) and maximize the distance for non-kinship pairs (negative pairs). The training process leverages a contrastive loss function to achieve this goal. The contrastive loss encourages the network to learn discriminative features by penalizing dissimilar positive pairs and rewarding well-separated negative pairs. The loss function is defined as follows:

$$L(W, Y, x_1, x_2) = (1 - Y)\frac{1}{2}d(x_1, x_2)^2 + Y\frac{1}{2}(max(0, m - d(x_1, x_2)))^2 \tag{12}$$

where W represents the network weights that are being optimized during the training process, Y acts as a binary label indicating whether the input (x_1,x_2) is a kinship pair or not, and everything composes into loss function L. In this formulation, the binary label $Y \epsilon \{0,1\}$ is used to represent the relationship between input pairs. Specifically, $Y = 0$ indicates that the pair of images are similar or exhibit kinship, while $Y = 1$ denotes that the images are dissimilar or non-kinship pairs.

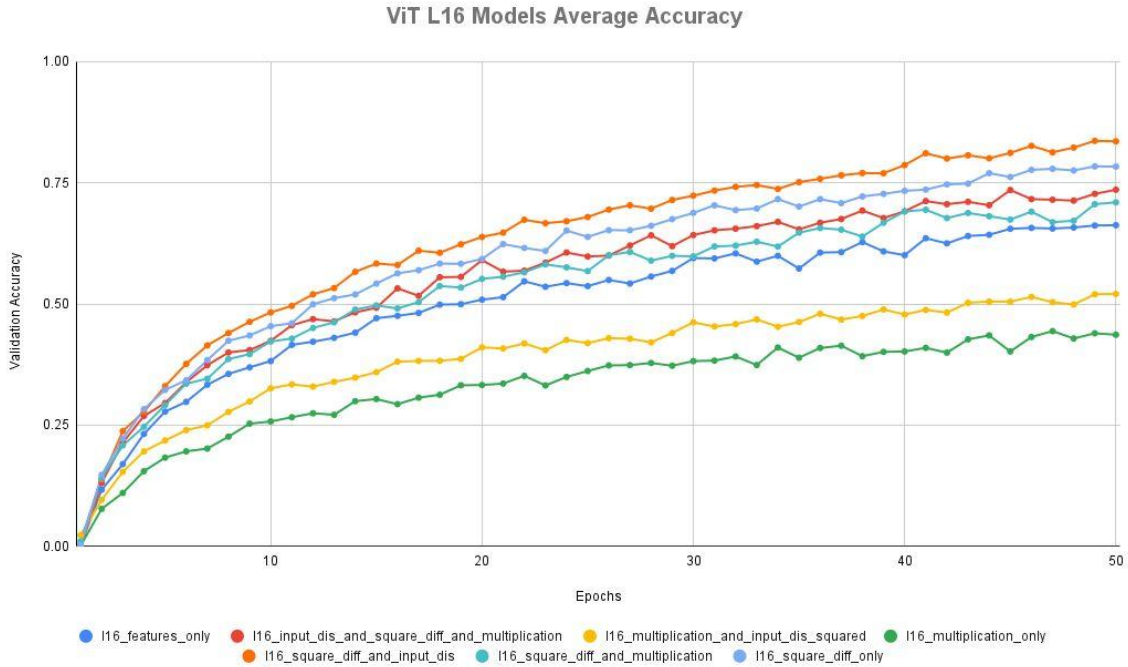

Figure. 7 The validation accuracy trends of different ViT L16 models over 50 epochs, using various feature fusion strategies

SSCNetViT model employs a margin parameter, which defines the minimum distance between non-kinship pairs to create clearer separation between dissimilar pairs during the training process.

The shared network weights (W) are crucial to process both inputs in the two branches of the Siamese network. By sharing these weights, the model ensures that the same feature extraction process is applied to both images, maintaining consistency in the extracted features. These weights are optimized to reduce the distance between similar pairs while increasing the distance between dissimilar pairs, guided by the contrastive loss function. In this experiment, a Vision Transformer (ViT) served as the feature extractor in the Siamese architecture. ViT captures global relationships in image data, useful for facial microexpression analysis. Images x_1 and x_2, are processed independently through the ViT, producing high-dimensional embeddings. These are compared using a distance function, and the combined representation is classified via fully connected layers. This setup, along with contrastive loss, proved effective for kinship verification by learning expressive and discriminative facial microexpression features. This enables the model to effectively discern subtle yet meaningful facial cues that correlate with familial relationships.

3.2. Facial microexpression image dataset

The experiment used a personal dataset alongside the FIW dataset with the proposed SSCNetViT model to classify kinship relationships. The model identified connections such as mother-daughter, mother-son, father-son, father-daughter, and sibling relationships. The dataset size was compared with SSCNetViT's performance. For training, 90% of the data was used, with the remaining 10% reserved for validation.

3.3. Comparison of proposed framework with other reference methods

To evaluate the proposed framework, its classification accuracy was compared to several existing methods using various datasets. The results, shown in Table 4, demonstrate the superior performance of the FIW-LaIndo method. SSCNetViT's improved performance can be attributed to several theoretical advancements. Firstly, its use of Vision Transformers allows it to capture global relationships within facial images, which is essential for recognizing subtle facial cues in microexpression-based kinship recognition. This is a significant advantage over traditional CNN-based methods.

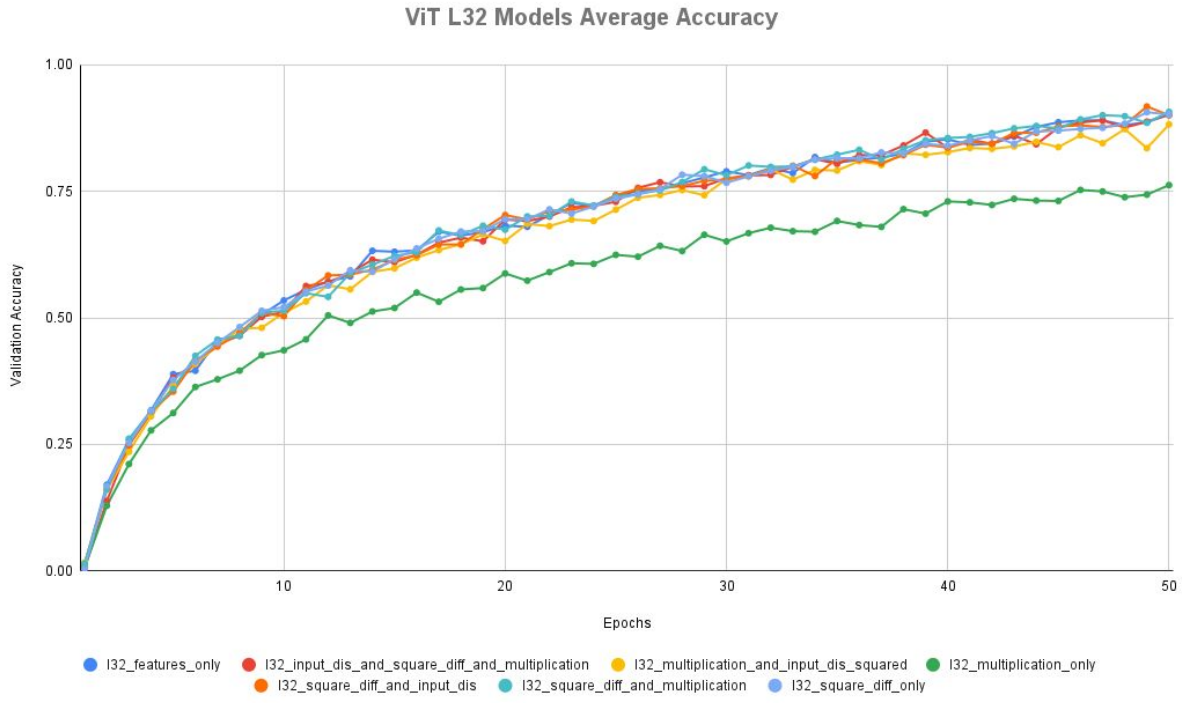

Figure. 9 The validation accuracy trends of different ViT L32 models over 50 epochs, using various feature fusion strategies

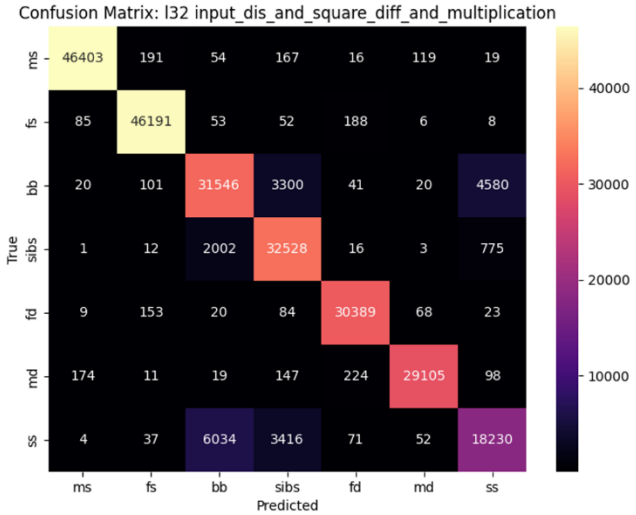

Figure. 8 Confusion matrix of the best L32 feature fusion method

Table 4. Calculation of Precision, Recall, and F1-score of the best model.

Kinship	Precision	Recall	F1-score
MD	0.9909	0.9774	0.984
MS	**0.9937**	**0.9880**	**0.991**
FD	0.9892	0.9884	0.985
FS	0.7940	0.9916	0.990
SS	0.7681	0.6547	0.707
BB	0.7940	0.7965	0.795
SiBs	0.8195	0.9205	0.867

Secondly, the model's enhanced feature fusion with the Siamese Sequential Classification Network (SSCN) architecture introduces a bias that improves generalization across kinship classes. By using techniques like Euclidean distance and element-wise operations, SSCNetViT more effectively differentiates familial relationships compared to traditional methods. Lastly, the incorporation of a contrastive loss function designed for discriminative learning further enhances the model's ability to distinguish between kin and non-kin pairs. learning further enhances the model's ability to distinguish between kin and non-kin pairs.

As illustrated in Table 4, models such as STLB-IP and STCLQP, evaluated on the CASME2 dataset, demonstrated accuracy rates of 59.51% and 56.10%, respectively. Similarly, the 3D-FCNN model, evaluated on the SMIC CASME dataset, demonstrated an accuracy rate of 55.49%. These results are noteworthy, but they are considerably lower than those obtained with our proposed method.

Further comparisons with models tested on the FIW dataset, including those based on deep and handcrafted features, Vuvko, DeepBlueAI, and others, demonstrate accuracy rates between 71.00% and 78.00%. While these methods demonstrate satisfactory performance, they still fall short of the 90.07% accuracy achieved by the proposed FIW-LaIndo method. It is noteworthy that FA-CNN, when tested on the RFIW'17 dataset, achieved an accuracy of 72.39%, which is also considerably lower than that of our model. The notable enhancement in accuracy exhibited by the SSCNetViT model can be attributed to its optimized management of the FIW dataset and its capacity to discern more intricate patterns within the data. The proposed model's superior capacity for generalization across intricate facial verification tasks is clearly demonstrated by this comparative analysis.

In conclusion, the proposed best method offers a notable enhancement in classification accuracy compared to existing state-of-the-art methods. Its elevated performance underscores the effectiveness of our approach in addressing the complexities associated with kinship verification tasks in facial images.

4. Results and discussion

racting facial microexpression features from the dataset using ViT, resulting in feature vectors for each frame. These frames are then paired based on kinship relations, such as pairing a daughter's frames with her father or mother. The paired features are fused and passed into the sequence-based classifier for kinship classification. The LaIndo dataset shows significant sample imbalances across kinship classes, similar to the FIW dataset. For instance, the MD class has 155 samples locally compared to 736 in FIW. This imbalance persists across other classes as well. All models were trained for 50 epochs using both LaIndo and FIW datasets, with results shown in Table 2. Each model achieved over 80% accuracy, with the L32 backbone performing best, averaging 90.07%. It scored 99.5% in the BB class and 84.0% in FD. In comparison, B16 and B32 backbones had lower average accuracies of 88.0% and 83.3%, respectively. Table 5 shows the precision, recall and F1-score of the best model.

In this experiment, the feature fusion method (x−y) L(x−y) outperformed other combinations across most kinship classes. The three models showed strong performance in kinship prediction, with the L32 backbone providing the most consistent and accurate results. Image 3 presents an overview of the facial recognition system using ViT. Features

from two images are extracted into a 768-dimensional space, then undergo various mathematical operations for feature fusion, resulting in a combined 2304-dimensional vector. This vector is processed through dense layers with dropout for regularization, leading to final classifications into categories such as MD, MS, FD, FS, SS, BB, and SiBs.

The proposed SSCNetViT leverages ViT, pre-trained on ImageNet to extract features from an Indonesian face dataset incorporating microexpressions. It utilizes a Siamese neural network (SNN) to process two inputs, measure feature distance, and perform feature fusion before classifying kinship. The study evaluated four ViT backbones—B16, B32, L16, and L32—for kinship recognition, with the highest accuracies of 0.8921 (B16), 0.8579 (B32), 0.8354 (L16), and 0.907 (L32). The model performs best on the MS class with an F1-score of 0.991, while the lowest performance is on the SS class with an F1-score of 0.707. Overall, the model has outstanding performance with an average F1-score of 0.903.

The L32 backbone showed the best performance due to its deeper architecture, superior feature extraction capability, and ability to handle imbalanced data robustly. Its smaller patch size captures finer details, while extensive self-attention mechanisms improve feature representation. With all backbones exceeding 85% accuracy, this ViT-based model outperforms previous studies, highlighting its effectiveness in kinship recognition using microexpression facial images.

5. Conclusion

In conclusion, this research presents a transfer learning of ImageNet Vision Transformers (ViT) for kinship recognition through facial microexpressions. The models achieved over 85% accuracy, with the L32 backbone performing best at 90.07%. This highlights SSCNetViT's effectiveness in feature extraction with imbalanced data. The L32 model showed consistent accuracy improvement and low loss. The combination of Siamese Neural Networks (SNN) and ViT-based feature extraction significantly improved kinship classification accuracy and efficiency. Future research could explore other factors like ethnicity and additional familial traits.

Conflicts of Interest

The authors declare no conflict of interest.

Author Contribution

Conceptualization: Ike Fibriani and Ike Fibriani, Methodology: Ike Fibriani, Eko Mulyanto Yuniarno, Software: Ike Fibriani, Eko Mulyanto Yuniarno, Validation: Eko Mulyanto Yuniarno, Ronny Mardiyanto, Mauridhi Hery Purnomo, Formal Analysis: Ike Fibriani, Writing—original draft preparation; Ike Fibriani, Writing—review and editing; Ike Fibriani, Visualization; Eko Mulyanto Yuniarno, Mauridhi Hery Purnomo, Supervision; Mauridhi Hery Purnomo, Funding acquisition: Ike Fibriani

Acknowledgments

The authors are grateful to the Balai Pembiayaan Pendidikan Tinggi (BPPT), The Ministry of Education, Culture, Research and Technology, and Lembaga Pengelola Dana Pendidikan (LPDP) Indonesia for supporting and funding this research.

References

[1] M. H. Shirali, A. P. Yazdanpanah, and S. Momtaz, "A Novel Approach for Kinship Verification using Facial Features and Machine Learning Techniques", *International Journal of Computer Applications (IJCA)*, Vol. 179, No. 24, 2018.

[2] P. Alirezazadeh, A. Fathi, and F. Abdali-Mohammadi, "Effect of Purposeful Feature Extraction in High-dimensional Kinship Verification Problem", *J. Comput. Secur.*, Vol. 3, No.3, pp. 183–191, 2016.

[3] Z. Xia, X. Hong, X. Gao, X. Feng, and G. Zhao, "Spatiotemporal Recurrent Convolutional Networks for Recognizing Spontaneous Micro-Expressions", *IEEE Trans. Multimed.,* Vol. 22, No. 3, pp. 626–640, 2020, doi: 10.1109/TMM.2019.2931351.

[4] M. Sajjad et al., "A comprehensive survey on deep facial expression recognition: challenges, applications, and future guidelines", *Alexandria Eng. J.*, Vol. 68, pp. 817–840, 2023, doi: 10.1016/J.AEJ.2023.01.017.

[5] P. Adegun and H. B. Vadapalli, "Facial microexpression recognition: A machine learning approach", *Sci. African*, Vol. 8, p. e00465, 2020, doi: 10.1016/J.SCIAF.2020.E00465.

[6] W. J. Yan, Q. Wu, J. Liang, Y. H. Chen, and X. Fu, "How Fast are the Leaked Facial Expressions: The Duration of Micro-Expressions", *J. Nonverbal Behav.*, Vol. 37, No.

4, pp. 217–230, 2013, doi: 10.1007/S10919-013-0159-8.

[7] M. Takalkar, M. Xu, Q. Wu, and Z. Chaczko, "A survey: facial micro-expression recognition", *Multimed. Tools Appl.*, Vol. 77, No. 15, pp. 19301–19325, 2018, doi: 10.1007/S11042-017-5317-2.

[8] X. Huang, S. J. Wang, G. Zhao, and M. Piteikainen, "Facial Micro-Expression Recognition Using Spatiotemporal Local Binary Pattern with Integral Projection", In *Proc. of the IEEE International Conference on Computer Vision*, pp. 1–9, 2015, doi: 10.1109/ICCVW.2015.10.

[9] H. Guerdelli, C. Ferrari, W. Barhoumi, H. Ghazouani, and S. Berretti, "Macro- and Micro-Expressions Facial Datasets: A Survey", *Sensors*, Vol. 22, No. 4, p. 1524, 2022, doi: 10.3390/S22041524.

[10] MB. Lopez, A. Hadid, E. Boutellaa, J. Goncalves, V. Kostakos, and S. Hosio, "Kinship verification from facial images and videos: human versus machine", *Mach. Vis. Appl.*, Vol. 29, No. 5, pp. 873–890, 2018, doi: 10.1007/S00138-018-0943-X/TABLES/10.

[11] R. Goel et al., "A Study of Deep Learning-Based Face Recognition Models for Sibling Identification", *Sensors*, Vol. 21, No. 15, p. 5068, 2021, doi: 10.3390/S21155068.\

[12] N. Samadiani et al., "A Review on Automatic Facial Expression Recognition Systems Assisted by Multimodal Sensor; Data", *Sensors*, Vol. 19, No. 8, p. 1863, 2019, doi: 10.3390/S19081863.

[13] H. H. Goh et al., "A multimodal approach to chaotic renewable energy prediction using meteorological and historical information", *Appl. Soft Comput.*, Vol. 118, p. 108487, 2022, doi: 10.1016/J.ASOC.2022.108487.

[14] R. L. Abduljabbar, H. Dia, and P. W. Tsai, "Development and evaluation of bidirectional LSTM freeway traffic forecasting models using simulation data", *Sci. Reports*, Vol. 11, No. 1, pp. 1–16, 2021, doi: 10.1038/s41598-021-03282-z.

[15] Othmani, D. Han, X. Gao, R. Ye, and A. Hadid, "kinship verification from faces using deep learning with imbalanced data", *Multimed. Tools Appl.*, Vol. 82, No. 10, pp. 15859–15874, 2023, doi: 10.1007/S11042-022-14058-6.

[16] J. Yu, M. Li, X. Hao, and G. Xie, "Deep Fusion Siamese Network for Automatic Kinship Verification", In *Proc. of* in *15th IEEE International Conference on Automatic Face and Gesture Recognition (FG 2020)*, pp. 892–899, 2020, doi: 10.1109/FG47880.2020.00127.

[17] X. Huang, G. Zhao, X. Hong, W. Zheng, and M. Pietik¨ainen, "Spontaneous facial micro-expression analysis using Spatiotemporal Completed Local Quantized Patterns", *Neurocomputing*, Vol. 175, No. PartA, pp. 564–578, 2016, doi:10.1016/J.NEUCOM.2015.10.096.

[18] J. Li, Y. Wang, J. See, and W. Liu, "Micro-expression recognition based on 3D flow convolutional neural network", *Pattern Anal. Appl.*, Vol. 22, No. 4, pp. 1331–1339, 2019, doi: 10.1007/S10044-018-0757-5/TABLES/2.

[19] Bisogni and F. Narducci, "kinship verification: how far are we from viable solutions in smart environments?", *Procedia Comput. Sci.*, Vol. 198, pp. 225–230, 2022, doi: 10.1016/J.PROCS.2021.12.232.

[20] J. Coe and M. Atay, "Evaluating Impact of Race in Facial Recognition across Machine Learning and Deep Learning Algorithms", *Comput. 2021*, Vol. 10, No. 9, p. 113, 2021, doi: 10.3390/COMPUTERS10090113.

[21] Dosovitskiy et al., "An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale", *arXiv:2010.11929 [cs]*, 2021, Available: https://arxiv.org/abs/2010.11929v2.

[22] Phan, H., Le, C.X., Le, V., He, Y. Nguyen, A. "Fast and Interpretable Face Identification for Out-Of-Distribution Data Using Vision Transformers", *arXiv:2311.02803* , 2023, Available: https://arxiv.org/abs/2311.02803.

[23] C. Liao, et al., "Domain Invariant Vision Transformer Learning for Face Anti-spoofing", In: *Proc. of 2023 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, Waikoloa, HI, USA, pp. 6087-6096, 2023, doi: 10.1109/WACV56688.2023.00604

[24] R. L. Abduljabbar, H. Dia, and P. W. Tsai, "Development and evaluation of bidirectional LSTM freeway traffic forecasting models using simulation data", *Sci. Reports*, Vol. 11, No. 1, pp. 1–16, 2021, doi: 10.1038/s41598-021-03282-z.

[25] L. Zhang, Y. Bian, P. Jiang, and F. Zhang, "A Transfer Residual Neural Network Based on ResNet-50 for Detection of Steel Surface Defects", *Appl. Sci*., Vol. 1, 2023.

[26] Kensert, P. J. Harrison, and O. Spjuth, "Transfer Learning with Deep Convolutional Neural Networks for Classifying Cellular Morphological Changes", *SLAS Discov.*, Vol. 24, No. 4, pp. 466–475, 2019, doi: 10.1177/2472555218818756.

[27] K. Han et al., "A Survey on Vision Transformer", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 45, No. 1, pp. 87–110, 2023.

[28] J. Lunter, "Beating the bias in facial recognition technology", *Biometric Technol. Today*, Vol. 2020, No. 9, pp. 5–7, 2020, doi: 10.1016/S0969-4765(20)30122-3.

[29] J. Jin, "Convolutional Neural Networks for Biometrics Applications", *SHS Web Conf.*, Vol. 144, p. 03013, 2022, doi: 10.1051/SHSCONF/202214403013.

[30] R. F. Rahmadi, I. K. E. Purnama, S. M. S. Nugroho, and Y. K. Suprapto, "Family-Aware Convolutional Neural Network for Image-based Kinship Verification", *Int. J. Intel. Eng. & Sys.*, pp. 20-30, 2020.

[31] J. P. Robinson, M. Shao, Y. Wu, and Y. Fu, "Families in the Wild (FIW): Large-scale kinship image database and benchmarks", In: *Proc. of* in *MM 2016: Proceedings of the 24th ACM International Conference on Multimedia*, pp. 242–246, 2016, doi: 10.1145/2964284.2967219.

[32] A. Shadrikov, "Achieving better kinship verification through better baseline", *arXiv preprint,* arXiv:2006.11739, 2020. Available: https://arxiv.org/abs/2006.11739.

[33] Z. Luo, Z. Zhang, Z. Xu, and L. Che, "Challenge report: Recognizing families in the wild data challenge", *arXiv preprint* arXiv:2006.00154, Available: https://arxiv.org/abs/2111.00598.

[34] S. Hörmann, M. Knoche and G. Rigoll, "A Multi-Task Comparator Framework for Kinship Verification", In: *Proc. of 2020 15th IEEE International Conference on Automatic Face and Gesture Recognition (FG 2020)*, Buenos Aires, Argentina, pp. 863-867, 2020, doi: 10.1109/FG47880.2020.00106.