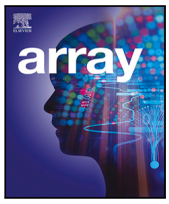
Array 26 (2025) 100401

Contents lists available at ScienceDirect

Array

journal homepage: www.elsevier.com/locate/array



Emotion recognition with a Randomized CNN-multihead-attention hybrid model optimized by evolutionary intelligence algorithm

Syed Muhammad Salman Bukhari [a], Muhammad Hamza Zafar [b], Syed Kumayl Raza Moosavi [c], Filippo Sanfilippo [b,d,*]

[a] Department of Electrical Engineering, Capital University of Science and Technology, Islamabad, 44000, Pakistan
[b] Department of Engineering Sciences, University of Agder, Grimstad, 4879, Norway
[c] SEECS, National University of Sciences and Technology, Islamabad, 44000, Pakistan
[d] Department of Software Engineering, Kaunas University of Technology, 44029 Kaunas, Lithuania

ARTICLE INFO

Keywords:
Emotion recognition
Randomized Convolutional Neural Networks (RCNN)
Multi-head attention mechanism
Evolutionary optimization
Football Team Training Algorithm (FTTA)
Real-time emotion detection

ABSTRACT

Emotion recognition systems are vital for various applications, yet existing models often face limitations in computational efficiency and accuracy, especially when handling complex emotional expressions in sequential data. To address these challenges, we propose an innovative emotion recognition framework that integrates a Randomised Convolutional Neural Network (RCNN) with a Multi-Head Attention model, further optimized by the Football Team Training Algorithm (FTTA) metaheuristic to enhance network parameters effectively. The RCNN, characterized by fixed random weights in its convolutional layers, efficiently extracts features from facial landmarks, enabling robust and diverse feature extraction while reducing computational load. This structure is complemented by a multi-head attention mechanism that processes temporal dynamics in emotion data, with both components optimized through FTTA to balance exploration and exploitation. Our hybrid model undergoes rigorous testing on a widely recognized emotion recognition dataset, outperforming conventional fully trainable models and alternative architectures. The results indicate a substantial improvement in classification accuracy, with an overall accuracy of 99%, and a significant reduction in computational demands, achieving a 65% faster training time on average compared to state-of-the-art models. These enhancements confirm the model's efficiency and robustness across various emotional classifications. The synergy between the RCNN's fixed-weight feature extraction and FTTA's optimization capabilities demonstrates a powerful solution for emotion recognition systems. The combination of accuracy and efficiency renders our model suitable for real-world applications, particularly in fields like healthcare and mental health monitoring, where real-time emotion detection can have significant impacts.

1. Introduction

The study of emotions has captivated researchers for over a century, with extensive debates on whether emotions are rooted primarily in cognitive constructs or are closely tied to physiological patterns [1]. Today, emotions are widely understood as affective states that influence essential cognitive functions such as learning, decision-making, and perception. This recognition has driven considerable interest in the use of deep learning techniques for emotion recognition, which holds valuable implications across fields, particularly in healthcare and smart surveillance. Automated emotion recognition is viewed as an essential component of human-like intelligence, often considered more predictive of life success than intelligence quotient (IQ) [2]. Furthermore, the ability of systems to reliably identify and interpret emotions can be transformative in smart healthcare, enabling early detection and classification of psychological conditions, including depression and stress-related disorders [3].

Emotion recognition technology is highly adaptable, with applications extending from mental health monitoring and road safety to social security measures. Human emotions are frequently detected through facial expressions [4], but they can also be inferred from other modalities, including speech, gestures, posture, and physiological signals. Over recent years, the deployment of emotion-aware systems has expanded into various sectors, such as e-learning, recommendation systems, and healthcare monitoring [5–8]. These advancements highlight the versatility of emotion recognition technology in enhancing personalized experiences and fostering more adaptive user interactions.

* Corresponding author at: Department of Engineering Sciences, University of Agder, Grimstad, 4879, Norway.
E-mail addresses: syedsalman.muhammad@gmail.com (S.M.S. Bukhari), muhammad.h.zafar@uia.no (M.H. Zafar), smoosavi.msai21seecs@seecs.edu.pk (S.K.R. Moosavi), filippo.sanfilippo@uia.no (F. Sanfilippo).

https://doi.org/10.1016/j.array.2025.100401
Received 7 September 2024; Received in revised form 26 February 2025; Accepted 22 April 2025
Available online 7 May 2025

1.1. Motivation

Emerging advancements in cloud computing, wearable sensors, and video surveillance are revolutionizing traditional healthcare, shifting the focus from treatment-centered to prevention-oriented approaches that emphasize personalized and efficient care [9]. Facial expressions, which serve as crucial non-verbal indicators of emotional states, are particularly relevant in this context. Their automated recognition could provide scalable, cost-effective solutions in healthcare systems [10]. This is especially timely given rising mental health challenges; for example, the incidence of depression among adults in the U.S. increased from 9% in 2017–2018 to 14% by April 2020, largely due to the COVID-19 pandemic [10]. An improvement in emotion recognition accuracy by even 1% could positively impact approximately 300,000 individuals, underscoring the societal importance of advancing this technology.

Additionally, recent research on Convolutional Neural Networks (CNNs) in Internet of Things (IoT) environments demonstrates the potential for real-time emotion recognition by harnessing CNNs' computational efficiency across dynamic settings [11]. Similarly, the adoption of AI and machine learning in sustainable practices highlights the growing need for scalable and efficient models that can adapt to diverse, real-world applications [12]. This trend inspires our approach to develop an emotion recognition system that is not only accurate but also optimized for real-time, resource-efficient deployment in environments such as healthcare and mental health monitoring.

1.2. Key contributions

In this research, we propose a novel approach to emotion classification by combining a Randomized Convolutional Neural Network (RCNN) with a multi-head attention model, further optimized by the Football Team Training Algorithm (FTTA). Our approach offers the following contributions:

- **Proposed Hybrid Model:** We introduce a hybrid architecture for emotion recognition that integrates an RCNN with multi-head attention to capture both spatial and temporal features in emotion data. This model leverages RCNN's fixed random weights for computational efficiency without compromising accuracy [13].
- **Optimization Using FTTA:** The FTTA algorithm is employed to optimize model parameters, balancing exploration and exploitation, which improves classification accuracy while maintaining efficiency, making it suitable for resource-limited environments [14].
- **Efficiency and Accuracy Enhancements:** Extensive testing on a widely recognized emotion recognition dataset demonstrated a classification accuracy of 99% with a 65% reduction in training time compared to conventional models, showcasing our model's potential for real-time applications.
- **Practical Implications for Real-World Applications:** Our approach addresses challenges in real-world applications, such as real-time processing and resource limitations, making it feasible for deployment in healthcare monitoring, mental health assessment, and other emotion-sensitive areas.

This research not only contributes to the field of emotion recognition but also advances the development of efficient, real-time AI systems suited for a range of applications where both accuracy and computational cost are critical.

2. Related work

The task of automatically recognizing human emotions has intrigued researchers for over four decades, presenting both intriguing opportunities and notable challenges. Recently, heightened interest in this field has emerged due to potential applications in enhancing safety, surveillance, and human–computer interaction [15]. This literature review highlights cutting-edge studies relevant to our proposed methodology. This section is structured into two primary subsections: the first focuses on the use of Machine Learning (ML) algorithms for emotion recognition, while the second discusses the application of randomization within Artificial Neural Networks.

2.1. Recent advances in facial emotion recognition

Recent advancements in facial emotion recognition (FER) have utilized deep learning techniques to enhance accuracy and applicability across various domains. These studies, summarized in Table 1, use diverse methodologies, including convolutional neural networks (CNNs), long short-term memory (LSTM) networks, and multimodal approaches.

For example, Kaur et al. (2024) provide a complete review of FER, highlighting its applications in medical diagnosis, customer feedback analysis, and automated driver assistance systems. Their study emphasizes the integration of CNNs for feature extraction and classification tasks [16]. Similarly, Palash and Bhargava (2023) propose the SAFER system, which integrates situation-aware analysis to improve FER accuracy in dynamic environments. This approach utilizes LSTM networks to capture temporal dependencies in facial expressions [15]. In the context of healthcare, an LSTM-based emotion detection framework has been developed to analyze physiological signals, facilitating applications in distance learning and patient monitoring. This system integrates Internet of Things (IoT) devices to provide real-time emotional assessments [17]. Moreover, the tourism industry has adopted emotion-aware recommendation systems to enhance user experiences. These systems analyze user emotions to provide personalized recommendations, thereby improving customer satisfaction [18]. Despite these advancements, challenges persist, particularly concerning the integration of multimodal data and the need for real-time processing capabilities. Ongoing research continues to address these issues, aiming to develop more robust and versatile FER systems.

2.2. Deep randomised neural networks (DRNNs)

This subsection explores Deep Randomised Neural Networks (DRNNs), highlighting the outcomes and implications of utilizing random weights in specific layers of deep neural architectures. By limiting training to select sections within a neural network stack, researchers have gained deeper insights into the fundamental mechanisms and behaviors of these networks, particularly within complex architectures. The concept of integrating fixed and trainable connections within neural networks has gained increasing traction in recent years, especially within computer vision applications, where computational efficiency is paramount [19]. DRNNs offer a promising balance between computational cost and classification accuracy, as studies demonstrate that incorporating one or more randomized layers within a neural architecture can maintain reasonable performance levels while significantly reducing computational power demands and expediting learning [20–22]. Though minor trade-offs in accuracy may occur, they are often offset by benefits such as reduced training complexity and enhanced speed. The historical development of DRNNs began with the exploration of shallow feed-forward neural networks in which a single layer was randomized—its connections fixed after initial randomization—resulting in reduced training demands. Early DRNN models, such as Random Vector Functional-Links (RVFL), were studied extensively in the 1990s, with their approximation capabilities verified in subsequent research [23–26].

The randomization concept has also extended to RCNNs, where convolutional layers incorporate fixed random weights to speed up training while striving to retain accuracy. This approach is particularly advantageous in applications requiring real-time processing and efficient resource utilization, such as image and video analysis [22]. However,

Table 1
Comparison of recent studies in Facial Emotion Recognition.

Ref.	Year	Model Summary	Limitations
[16]	2024	Comprehensive review of FER applications using CNNs for feature extraction and classification.	Focuses primarily on static images; limited discussion on temporal dynamics.
[15]	2023	Introduces the SAFER system utilizing LSTM networks for situation-aware FER in dynamic settings.	Requires extensive computational resources for real-time analysis.
[17]	2022	Develops an LSTM-based framework analyzing physiological signals for emotion detection in healthcare.	Limited to physiological data; lacks integration with visual modalities.
[18]	2013	Implements an emotion-aware recommendation system in the tourism sector to enhance user experience.	Relies on user-reported emotions; potential biases in self-reporting.

achieving an optimal trade-off between speed and accuracy remains a critical challenge in RCNN design, particularly in tasks requiring high precision alongside rapid processing. For instance, Ulyanov et al. [27] illustrate that a randomly initialized CNN can be used effectively as a prior for various image processing tasks, suggesting that randomization within neural structures can carry meaningful architectural information. Similarly, Pons et al. [28] propose a randomized CNN model for audio classification, showcasing the versatility of DRNN approaches across different data domains. The capacity of DRNNs to enhance training efficiency while maintaining practical accuracy makes them a compelling option in fields where computational resources are limited or real-time response is critical. Our proposed approach leverages these advantages within an RCNN framework, optimized by a multi-head attention model, to deliver a robust solution for emotion recognition.

While DRNNs and RCNNs have explained substantial potential in reducing training costs and enhancing processing speed, our study distinguishes itself by specifically targeting the application of randomized convolutional layers within an emotion recognition framework. Unlike prior studies focused on static image or audio classification [27,28], our approach integrates multi-head attention with RCNN, optimizing both spatial and temporal aspects crucial to emotion detection. Additionally, our work uses the Football Team Training Algorithm (FTTA) to fine-tune parameters, addressing a significant gap in previous studies where trade-offs between accuracy and computational efficiency were left unoptimized.

This study intends to provide a scalable, real-time emotion recognition solution that is particularly well-suited for healthcare and mental health monitoring, where efficient processing and high accuracy are paramount. The combined use of DRNN principles within an RCNN framework, alongside multi-head attention and FTTA, positions our model as a unique contribution to the field, offering enhanced performance in emotion classification under computational constraints. Building on insights from the current literature, we now introduce our proposed model, which uses an RCNN integrated with a multi-head attention mechanism. This architecture is further optimized using FTTA, aimed at achieving a robust balance between processing speed and classification accuracy.

3. Proposed model

In this paper, we present a novel emotion classification framework that integrates an RCNN with a multi-head attention model, further enhanced by the FTTA for optimal parameter tuning. This approach marks a substantial advancement in emotion recognition from facial features data extracted from images. By harnessing the strengths of the RCNN and multi-head attention architectures, and optimizing their performance with the FTTA metaheuristic, our model achieves efficient feature extraction, rapid training capabilities, and effective capture of temporal dynamics in video data. This section delves into the architecture and functionality of our proposed model, emphasizing its unique features and the potential impact it holds for real-world emotion recognition applications. The integration of the FTTA metaheuristic not only streamlines the optimization process but also brings a novel perspective to balancing exploration and exploitation in deep learning models.

To illustrate the different phases of FTTA, we present Fig. 1 and Fig. 2, which visually represent the behaviors mimicked during collective and group training. These figures highlight how FTTA divides the optimization process into coordinated training stages, enhancing model accuracy and efficiency.

3.1. Football team training algorithm (FTTA)

The FTTA is a novel meta-heuristic optimization algorithm inspired by professional football training sessions. It comprises three main phases: collective training, group training, and individual extra training. Fig. 1 illustrates the Collective Training Behaviour (CTB), where all entities collaborate towards improving the overall model. This stage sets the foundation for the model's optimization by aligning its parameters through collective efforts. Following this, Fig. 2 depicts the Group Training Behaviour (GTB), in which subsets of entities are divided into smaller groups, each focusing on optimizing specific aspects of the model's performance. This division into groups allows for targeted improvements before integrating the results with the larger system, enhancing both specificity and effectiveness in the training process.

3.1.1. Collective training

In the collective training phase, players (potential solutions) are evaluated using a fitness function to assess their current level. Based on this, they make a training plan. Players are categorized into four types, each representing different strategies:

- **Followers:** These players aim to emulate the best player's performance, moving towards the best solution in every dimension, but with a random factor due to their limitations:

$$F_{k,i,j}^{new} = F_{k,i,j}^{old} + \text{rand} \times (F_{k,j}^{best} - F_{k,i,j}^{old}) \tag{1}$$

- **Discoverers:** Rational players are those who consider both the best and worst players, striving to improve towards the best while avoiding becoming the worst:

$$F_{k,i,j}^{new} = F_{k,i,j}^{old} + \text{rand}_1 \times (F_{k,j}^{best} - F_{k,i,j}^{old}) - \text{rand}_2 \times (F_{k,j}^{worst} - F_{k,i,j}^{old}) \tag{2}$$

- **Thinkers:** Alert players focus on the gap between the best and worst players, aiming to reach this difference in each dimension:

$$F_{k,i,j}^{new} = F_{k,i,j}^{old} + \text{rand} \times (F_{k,j}^{best} - F_{k,j}^{worst}) \tag{3}$$

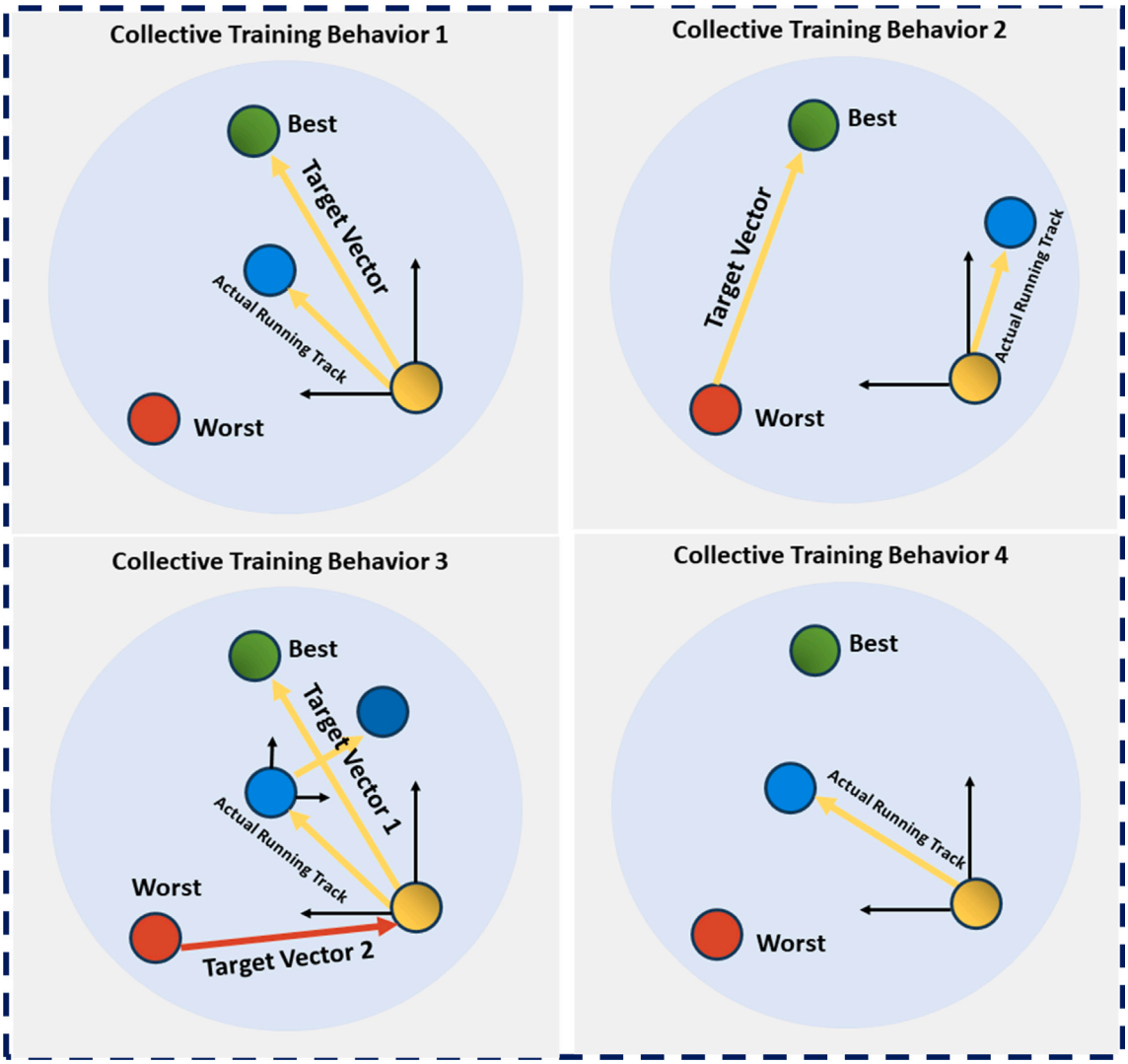


Fig. 1. Illustration of Collective Training Behaviour (CTB). This shows the training process where multiple entities collaborate to enhance the overall model performance.

- **Volatilities:** Independent players train on their own, experiencing fluctuations in their state. Over time, the fluctuations decrease, transitioning from global to local searches:

$$F_{k,i,j}^{new} = F_{k,i,j}^{old} \times (1 + t(k)) \tag{4}$$

where $F_{k,i,j}^{new}$ and $F_{k,i,j}^{old}$ represent the new and old states of player i on dimension j at iteration k, respectively. $F_{k,j}^{best}$ and $F_{k,j}^{worst}$ are the values of the best and worst players in dimension j at iteration k. The term $t(k)$ is a random number following a t-distribution with degrees of freedom equal to the current iteration number. Through these player types and their associated equations, the FTTA algorithm captures a diverse set of exploration and exploitation behaviors observed in football training to optimize a given problem.

3.1.2. Group training

Following Collective Training, the FTTA progresses to the Group Training phase. Players are clustered into four categories akin to football positions: Strikers, Midfielders, Defenders, and Goalkeepers. This classification is achieved using the MGEM adaptive clustering method (MixGaussEM), which is mathematically represented as:

$$\text{All Players} \xrightarrow{\text{MGEM}} [\text{Team 1, Team 2, Team 3, Team 4}] \tag{5}$$

If any team's count is less than the set threshold (Team number), a uniform random grouping is applied to reassign players into four groups.

Post-clustering, players engage in three training states: Optimal Learning, Random Learning, and Random Communication, with learning probabilities (p_{study}) and communication probabilities (p_{comm}). Players select their state randomly in each iteration.

Optimal learning. Players may learn from the best player in their group with a probability p_{study}:

$$F_{k,\text{team}l,i,j}^{new} = \begin{cases} F_{k,\text{team}l,j}^{best} & \text{if rand} \le p_{\text{study}} \\ F_{k,\text{team}l,i,j}^{old} & \text{if rand} > p_{\text{study}} \end{cases} \tag{6}$$

Random learning. Similarly, players may learn from a random peer within their group:

$$F_{k,\text{team}l,i,j}^{new} = \begin{cases} F_{k,\text{team}l,j}^{random} & \text{if rand} \le p_{\text{study}} \\ F_{k,\text{team}l,i,j}^{old} & \text{if rand} > p_{\text{study}} \end{cases} \tag{7}$$

Random communication. Communication between players involves exchanging skills:

$$F_{k,\text{team}l,i,j}^{new} = F_{k,\text{team}l,j}^{random} \times (1 + \text{randn}) \tag{8}$$

$$F_{k,\text{team}l,\text{random},j}^{new} = F_{k,\text{team}l,i,j}^{old} \times (1 + \text{randn}) \tag{9}$$

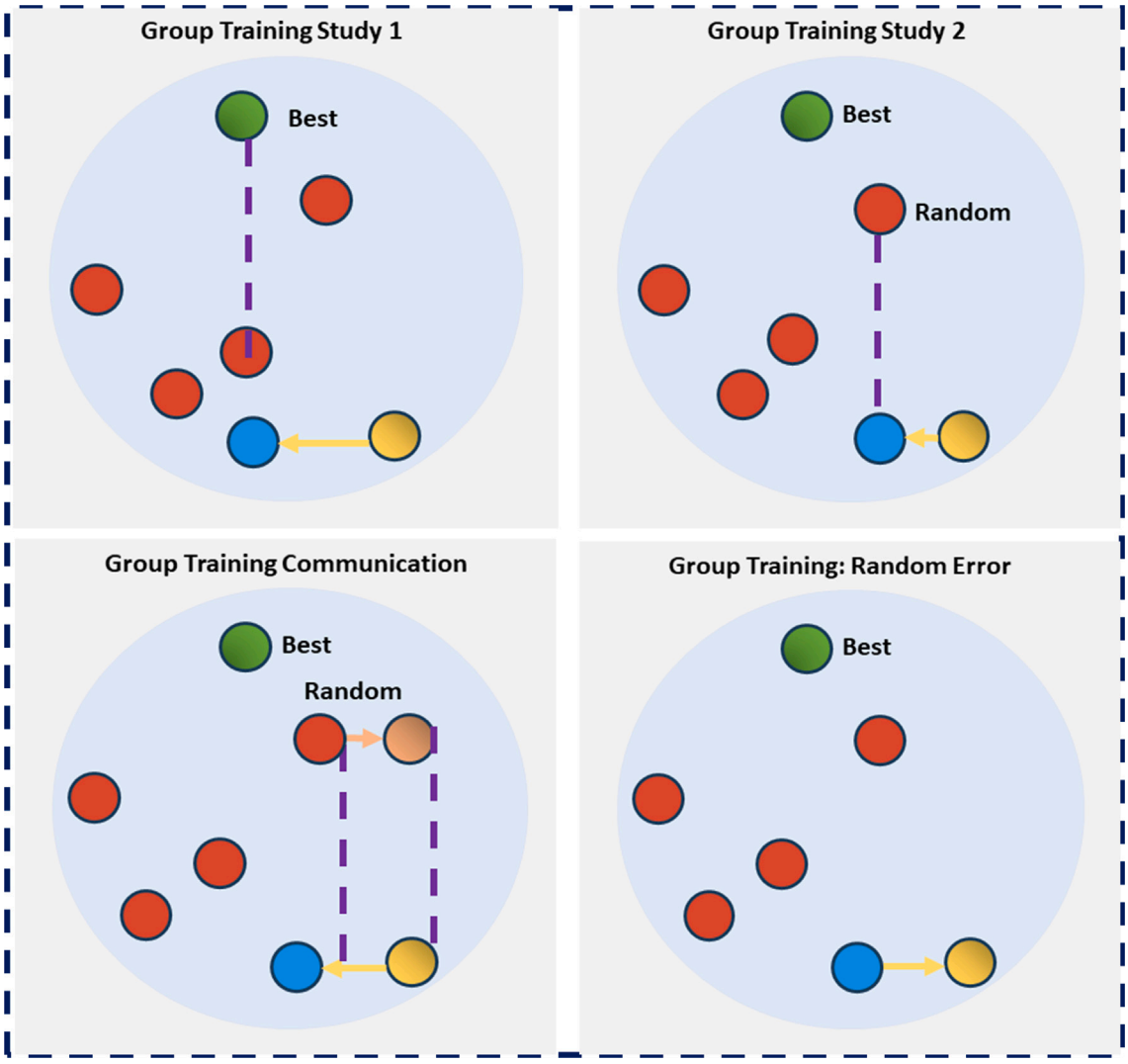


Fig. 2. Illustration of Group Training Behaviour (GTB). This highlights how groups of entities train in sub-groups to optimize specific aspects of model training before integrating with the larger system.

Random error. A small probability of error may result in learning from incorrect dimensions:

$$F^{new}_{k,\text{team}l,i,j} = \begin{cases} F^{random1}_{k,\text{team}l,\text{random2}} & \text{if rand} \le p_{\text{error}} \\ F^{old}_{k,\text{team}l,i,j} & \text{if rand} > p_{\text{error}} \end{cases} \tag{10}$$

In the equations, k is the iteration number, team$l$ indicates group l, $F^{best}_{k,\text{team}l,j}$ and $F^{random}_{k,\text{team}l,j}$ represent the best and a random player's skill in dimension j within group l, respectively. randn is a normally distributed random number, and p_{error} is the probability of a training error.

3.1.3. Individual extra training

The final phase of the FTTA is individual extra training. This phase occurs in group training, where the fitness values are recalculated for each player (solution), allowing the update of player status by replacing poorer fitness values with better ones.

Enhancing the best player. The coach (algorithm) selects the best player and provides focused training to further enhance their skills, which, in turn, helps elevate the entire team's performance. The training formula for the best player is defined as:

$$F^{new}_{\text{Best},k} = F^{old}_{\text{Best},k} \times \left(1 + \left(1 - \frac{1}{k}\right) \times \text{Gauss} + \frac{1}{k} \times \text{Cauchy}\right) \tag{11}$$

where $F^{new}_{\text{Best},k}$ represents the new state of the best player after training, $F^{old}_{\text{Best},k}$ is the old state, and k is the iteration number. The term Gauss denotes a Gaussian distributed random number, and Cauchy represents a Cauchy distributed random number.

Gaussian-Cauchy joint variation. The joint variation of Gaussian and Cauchy distributions is used to simulate the individual extra training. The rationale behind this choice is to allow for significant potential improvement early in training, where a larger Cauchy component facilitates global search by providing a wide-ranging enhancement. As the number of iterations increases, the Gaussian component becomes more dominant, reflecting the increased difficulty in player improvement and thus promoting a more localized search.

This phase encapsulates the idea that the best player's continued improvement can serve as an inspiration and benchmark for the rest of the players, thus driving the overall progression of the team (solution set).

3.2. Convolutional neural network (CNN)

A Convolutional Neural Network (CNN) is a deep learning model primarily used for processing data with a grid-like topology, such as images. A CNN typically consists of a series of layers that transform the input data into outputs through a process of feature extraction and nonlinear transformations [29].

Convolutional layers. The primary component of a CNN is the convolutional layer. The layer applies a set of learnable filters (or kernels) to the input. For an input matrix $X \in \mathbb{R}^{H\times W\times D}$, where H, W, and D represent height, width, and depth (channels), respectively, a filter $F \in \mathbb{R}^{h\times w\times D}$ is convolved across the spatial dimensions of the input. The convolution operation is defined as:

$$Y_{i,j} = \sum_{m=0}^{h-1}\sum_{n=0}^{w-1}\sum_{d=0}^{D-1} F_{m,n,d}\cdot X_{i+m,j+n,d} + b \tag{12}$$

where $Y_{i,j}$ is the output at position (i, j), and b is a bias term. This operation is applied to each filter to produce a feature map that captures spatial hierarchies in the input.

Pooling layers. Pooling layers are used to reduce the spatial dimensions of the input volume. The most common form of pooling is max pooling, which partitions the input into a set of non-overlapping rectangles and outputs the maximum value from each rectangle.

Fully connected layers. After several convolutional and pooling layers, the high-level reasoning in the neural network is done via fully connected layers. Neurons in a fully connected layer have connections to all activations in the previous layer. These layers are typically used for classification purposes at the end of the network.

Activation functions. Nonlinear activation functions are used after convolutional and fully connected layers. The most common activation function used in CNNs is the Rectified Linear Unit (ReLU), defined as $f(x) = \max(0, x)$.

3.3. Randomised convolutional neural network (RCNN)

The RCNN modifies the conventional CNN by introducing randomization in the convolutional layers. The architecture of the RCNN is similar to that of a standard CNN, with the key difference being the random and fixed weights in certain convolutional layers.

Randomised convolutional layers. In the RCNN, the initial convolutional layers use fixed random weights. Unlike typical CNNs, where weights are learned through backpropagation, these weights in the RCNN are randomly initialized and remain unchanged during training. For a randomized convolutional layer with input $X \in \mathbb{R}^{H\times W\times D}$ and fixed random filter $F_r \in \mathbb{R}^{h\times w\times D}$, the convolution operation is the same as in standard CNNs but without updating F_r during backpropagation:

$$Y_{i,j} = \sum_{m=0}^{h-1}\sum_{n=0}^{w-1}\sum_{d=0}^{D-1} F_{r,m,n,d}\cdot X_{i+m,j+n,d} + b \tag{13}$$

where $Y_{i,j}$ is the output at position (i, j), and b is a bias term.

The use of fixed random weights allows the RCNN to leverage the benefits of convolutional layers for feature extraction while significantly reducing the computational cost associated with training large neural networks.

3.4. Transformer model

The transformer model, integrated into our RCNN architecture, plays a pivotal role in capturing the temporal dynamics of video frame sequences. The transformer's primary mechanism is based on self-attention, which allows the model to weigh the importance of different parts of the input sequence differently [30].

Given the output Y from the RCNN, which contains the spatial features extracted from facial landmarks, the transformer processes this data to understand the temporal relationships between frames. The transformer model comprises multiple layers, each consisting of a multi-head self-attention mechanism and a position-wise feed-forward network.

Self-attention mechanism. The self-attention mechanism in the transformer computes the attention scores for each element in the input sequence. For each element in Y, the transformer calculates a set of query (Q), key (K), and value (V) vectors through linear transformations. Mathematically, this can be represented as:

$$Q = YW^Q, \quad K = YW^K, \quad V = YW^V \tag{14}$$

where W^Q, W^K, and W^V are the weight matrices for the query, key, and value, respectively.

The attention scores are computed using the scaled dot-product attention, given by:

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V \tag{15}$$

where d_k is the dimension of the key vectors, and $\sqrt{d_k}$ is used for scaling. The softmax function ensures that the attention scores sum to 1, allowing the model to focus on the most relevant parts of the input.

Multi-head attention. The transformer employs multi-head attention to capture information from different representation subspaces. This is achieved by performing the self-attention mechanism in parallel with different sets of Q, K, and V matrices. The outputs of these parallel attention heads are then concatenated and linearly transformed to produce the final output.

Position-wise feed-forward network. Each transformer layer also includes a position-wise feed-forward network, which applies a fully connected layer to each position separately and identically. This network consists of two linear transformations with a ReLU activation in between:

$$\text{FFN}(x) = \max(0, xW_1 + b_1)W_2 + b_2 \tag{16}$$

where W_1, W_2, b_1, and b_2 are the weights and biases of the feed-forward network.

The combination of self-attention and the feed-forward network in each transformer layer provides a robust mechanism for processing the sequential data. The self-attention mechanism allows the model to focus on different parts of the input sequence, enhancing its ability to capture temporal dependencies. In contrast, the feed-forward network applies a more traditional neural network processing step, ensuring that each position in the sequence is transformed identically. This ensures that the model captures both the global context provided by self-attention and the local context through position-wise processing.

The final output of the transformer, after processing through multiple layers of self-attention and feed-forward networks, is a rich representation of the input sequence that incorporates both spatial features extracted by the RCNN and the temporal dynamics of the video frames. This comprehensive feature representation is crucial for the accurate classification of emotions in our proposed model.

3.5. Hybrid model: RCNN with multi-head attention

The proposed hybrid model synergizes the strengths of the Randomised Convolutional Neural Network (RCNN) and the multi-head attention model to enhance emotion classification from facial landmark data. As depicted in Fig. 3, this architecture integrates the RCNN and multi-head attention components, where each contributes distinct capabilities: the RCNN for initial feature extraction and the transformer-based attention model for capturing temporal dynamics.

The RCNN component of the hybrid model is responsible for initial feature extraction from the input facial landmark data. Let $X \in \mathbb{R}^{H\times W\times D}$ represent the input data, where H, W, and D denote height, width, and depth, respectively. The RCNN employs fixed random weights in its convolutional layers, denoted as $W_r \in \mathbb{R}^{h\times w\times D}$. The convolution operation in the RCNN is given by:

$$Y_{i,j} = f\left(\sum_{m=0}^{h-1}\sum_{n=0}^{w-1}\sum_{d=0}^{D-1} W_{r,m,n,d}\cdot X_{i+m,j+n,d} + b\right) \tag{17}$$

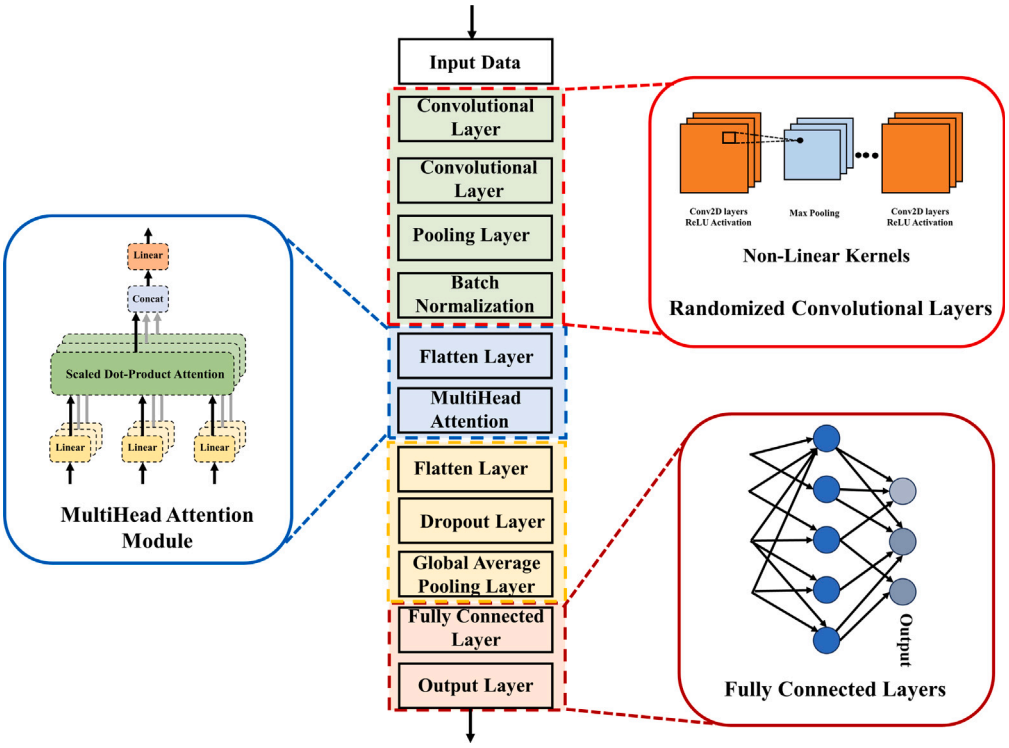


Fig. 3. Detailed architecture of multiHead attention-based randomized convolutional network.

where f is a nonlinear activation function (e.g., ReLU), and b is the bias term. The output $Y \in \mathbb{R}^{T\times d}$, where T is the number of time steps (frames) and d is the feature dimension, serves as the input to the transformer model.

The transformer model then processes these feature vectors extracted by the RCNN to capture temporal dynamics in the video frames, enhancing the model's ability to discern subtle changes across sequential data. The transformer architecture comprises an encoder and a decoder, both equipped with self-attention and feed-forward layers, allowing the model to focus on different aspects of the sequence to analyze the temporal relationships effectively.

Encoder. The transformer encoder processes the RCNN output Y through self-attention and position-wise feed-forward layers. The self-attention mechanism is defined as:

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V \tag{18}$$

where Q, K, and V are query, key, and value matrices derived from Y. The position-wise feed-forward network in the encoder applies the transformation:

$$\text{FFN}(x) = \max(0, xW_1 + b_1)W_2 + b_2 \tag{19}$$

where W_1, W_2, b_1, and b_2 are the weights and biases of the feed-forward network.

Decoder. The transformer decoder generates the final output by processing the encoder's output through additional layers of self-attention and feed-forward networks, incorporating an attention mechanism that focuses on the encoder's output. This collaborative integration of RCNN and transformer-based attention in Fig. 3 allows the hybrid model to extract spatial features through the RCNN, while the attention mechanism in the transformer enables the capture of temporal dependencies across frames. This dual-process approach results in a comprehensive representation of the emotional content in video frames, combining spatial and temporal features to achieve accurate emotion classification.

3.6. Explanation of FTTA pseudo code

The Football Team Training Algorithm (FTTA) is a systematic, metaheuristic approach designed to optimize the hyperparameters and weights of the hybrid RCNN-multi-head Attention model efficiently. FTTA's multi-phase training process enables it to dynamically balance exploration and exploitation, allowing the RCNN-multi-head Attention model to identify the optimal parameter values for improved accuracy in emotion classification. The following explains each step of the FTTA, corresponding to the lines in the pseudo-code, demonstrating how FTTA iteratively refines the model's hyperparameters.

1. Lines 1–2: The algorithm environment is initialized by clearing all variables and the command window, ensuring a fresh start for each optimization cycle.
2. Lines 3–7: The objective function, representing the model's emotion recognition accuracy, is defined. Key FTTA parameters are also set here, including `MaxGen` (number of iterations) and `nPop` (population size). Probabilities for the different learning modes—study, communication, and error—are established, creating a structured environment for the team-based optimization.
3. Line 8: FTTA initializes the population with RCNN-multi-head Attention model instances, each with randomized hyperparameter values. This initial population provides diverse starting points, crucial for exploring different regions of the solution space effectively.
4. Lines 9–22: The core loop of FTTA iterates across generations, or optimization cycles, refining the model's parameters in each step. Here's how each phase operates to achieve optimal parameter values:

    - Lines 10–12: Each "player" (model instance) calculates its fitness value based on accuracy. This fitness score indicates how well the model instance performs, with higher scores correlating to more effective parameter sets.

Algorithm 1 Pseudo code for the FTTA with Hybrid RCNN-multi-head Attention Model

```
CLC
CLEAR ALL
Set Objective Function (e.g., emotion recognition accuracy)
Parameters:
Set MaxGen = 500 {The number of iterations is 500}
Set nPop = 50 {The population size is 50}
Set pstudy = 0.2; pcomm = 0.2; perror = 0.001
/*Set the parameters of FTTA*/
/*Initialise population with RCNN-multi-head Attention Models*/
for I = 1 to MaxGen do
   for EACH i in 1 ≤ i ≤ nPop do
      Calculate the fitness value of each football player F_i
   end for
   Find the Best & Find the Worst
   Do Collective Training according to player types: Followers, Discoverers, Thinkers, Volatilities
   /*Divide the Population into Groups and Do Group Training*/
   MGEM Clustering or Random Uniform Grouping to form teams
   /*Each player in a team performs Optimal Learning, Random Learning, or Random Communication*/
   for EACH i in 1 ≤ i ≤ nPop do
      Calculate the fitness value of each football player F_i
   end for
   Update the Players using the new fitness values
   Find the Best player
   Do Individual Extra Training for the Best player
   if F(F_Best^new) < F(F_Best^old) then
      F_Best^new = F_Best^old × (1 + (1 − 1/k) × Gauss + 1/k × Cauchy)
   else
      F_Best^new = F_Best^old
   end if
   /*Store the Optimal solution in Archive*/
   t = t + 1
end for
```

- Line 13: The algorithm identifies the best and worst players based on fitness scores. These players act as benchmarks, guiding the adjustments of other players' parameters in subsequent steps.
- Line 14: During the collective training phase, players adapt their parameter strategies according to their designated types (Follower, Discoverer, Thinker, Volatility), promoting diverse learning approaches across the population. Each player's strategy encourages a balance of exploration and refinement, contributing to both short-term gains and long-term improvements.
- Lines 15–16: Players are grouped using MGEM adaptive clustering or random grouping, emulating real-world team dynamics. This phase encourages collaborative learning, where players in the same group can benefit from one another's strengths.
- Lines 17–19: Within each group, players enter one of three states: Optimal Learning, Random Learning, or Random Communication. This diversity in learning ensures that players refine their parameters through both structured updates from top performers and randomized adjustments, enhancing overall adaptability.
- Lines 20–22: Updated fitness values are recalculated after group training, with each player's parameters adjusted based on its new learning outcomes.

5. Lines 23–29: The best-performing player receives individual extra training using Gaussian-Cauchy parameter variation. This additional refinement focuses on enhancing the top model instance's accuracy and ensures that the overall model retains high-quality parameter sets. This step further ensures convergence towards optimal values for critical hyperparameters by iteratively adjusting top solutions.

This structured FTTA approach, inspired by football team dynamics, fosters a smooth, iterative optimization of the RCNN-multi-head Attention model. By combining collective and individual refinement phases, FTTA methodically tunes hyperparameters to enhance the model's accuracy and computational efficiency. Through each iteration, the algorithm narrows down the parameter values to those that best balance performance with efficiency, ensuring that the hybrid model meets the demands of real-world emotion classification tasks.

3.7. Tuned hyperparameters after FTTA:

In our proposed *MultiHead Attention-based Randomized Convolutional Network* (RCNN) architecture, we systematically optimized several critical hyperparameters to ensure both efficiency and accuracy in emotion classification. Our selection of these hyperparameters was driven by rigorous experimentation, as detailed below, to achieve optimal performance within the RCNN framework, which is essential for real-time emotion detection applications.

The first critical parameter, denoted as F, signifies the number of convolutional filters used in the initial `Conv2D` layer, set at 32. This number was chosen based on trials indicating that 32 filters provide a sufficient diversity of feature extraction, capturing distinct facial cues related to emotional expressions without excessive computational cost. The kernel size, represented by D and valued at 2, defines the dimensions of the convolution window, ensuring fine-grained feature recognition by focusing on small, localized image regions.

The RCNN uses a transformer mechanism characterized by the number of attention heads (num_heads), set to 2. This choice, determined through cross-validation, was found to be effective for maintaining attention over multiple facial regions in each sequence, enabling the model to focus simultaneously on different temporal and spatial nuances of emotional expressions. The feed-forward network dimension within the transformer block, ff_dim, is optimized to 64, striking a balance between model complexity and computational efficiency. This dimensionality was selected to provide ample capacity for learning while maintaining manageable computational demands, crucial for deploying the model in resource-constrained environments.

Additionally, the batch size used during training is set to 32, a size that supports balanced memory utilization while ensuring efficient convergence. The learning rate is initialized at 0.001, a value carefully chosen to allow stable and progressive convergence towards the optimal solution. To optimize parameter updates, we employ the Adam optimizer, which provides adaptive learning rates for each parameter and has shown effective convergence performance in deep learning applications.

Lastly, the number of neurons in the dense layer (R) is set to 64, ensuring a robust feature aggregation mechanism before the final classification. This layer acts as a crucial component in translating the high-level features extracted and processed by previous layers into predictions for emotion categories.

The FTTA framework explored a range of values for each hyperparameter, shown in Table 2. This exploration allowed FTTA to identify configurations that achieved the highest accuracy with minimal computational burden, thereby supporting both real-time performance and reliability.

Table 2
Optimized hyperparameters and FTTA Search ranges for the MultiHead Attention RCNN.

Parameter	Description	Optimal Value	FTTA Search Range
F	Number of Conv2D Filters	32	16, 32, 64
D	Conv2D Kernel Size	2	2, 3, 5
num_heads	Number of Attention Heads	2	1, 2, 4
ff_dim	Feed-Forward Network Dimension	64	32, 64, 128
R	Number of Neurons in Dense Layer	64	32, 64, 128
Batch Size	Training Batch Size	32	16, 32, 64
Learning Rate	Initial Learning Rate	0.001	0.0001, 0.001, 0.01
Optimizer	Optimization Algorithm	Adam	Adam, SGD, RMSprop

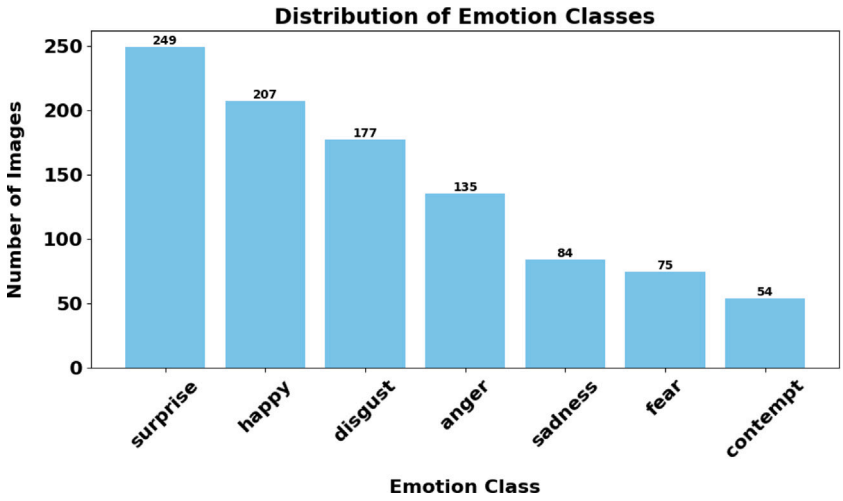


Fig. 4. Distribution of emotion classes in CK+ dataset.

Table 3
Dataset size and Train/Test Split for CK+ Dataset.

Dataset	Training set size	Test set size
CK+ (Extended Cohn-Kanade)	474 images (80%)	119 images (20%)

4. Dataset preprocessing and visualization

This section details the preprocessing techniques applied to the CK+ (Extended Cohn-Kanade) dataset, which serves as the foundation for training and evaluating the proposed emotion classification model. The CK+ dataset is widely recognized for its extensive set of labeled facial images, providing a benchmark for facial expression analysis [31]. It comprises 593 video sequences from 123 subjects, with each sequence capturing a progression of facial expressions categorized into seven distinct emotions: anger, disgust, fear, happiness, sadness, surprise, and contempt. To ensure a rigorous evaluation framework, the dataset is partitioned into training and testing subsets using an **80:20 stratified split**, preserving the proportional representation of each emotion class. This methodology prevents class imbalance in the test set, ensuring that underrepresented emotions remain adequately represented for validation. Table 3 summarizes the dataset composition across the training and test sets.

The class distribution within the CK+ dataset is depicted in Fig. 4. The dataset exhibits inherent imbalances, with certain emotions having fewer instances than others. Specifically, 'Surprise' and 'Happy' classes contain a significantly higher number of images than 'Fear' and 'Contempt.' To mitigate this difference, augmentation techniques are used to enhance the diversity of the dataset, thereby improving model generalization.

Feature extraction plays a crucial role in facilitating robust emotion classification. To capture the geometric properties of facial expressions, spatial features are derived from facial landmarks, focusing on key points indicative of structural changes in expression. Also, appearance-based features are extracted using local binary pattern (LBP) histograms, which encode texture variations essential for distinguishing fine-grained emotional differences. These extracted features are subsequently processed by the RCNN integrated with a multi-head attention mechanism, allowing the model to simultaneously leverage spatial and temporal dependencies. To ensure reproducibility, the CK+ dataset is publicly accessible for academic research. This transparency enables independent verification of our experimental setup and results. Also, the dataset undergoes preprocessing to optimize input representation, including normalization, resizing, and augmentation techniques adapted for deep learning architectures. The next subsection provides a complete breakdown of the preprocessing pipeline implemented in this study.

4.1. Data preprocessing

Preprocessing plays a pivotal role in ensuring the dataset is optimally structured for model training and evaluation. The pipeline implemented in this study standardizes input data and enhances model robustness. The preprocessing steps, executed using TensorFlow and Keras frameworks, include:

- **Data Loading and Label Extraction:** The dataset is loaded, and class labels are extracted from the corresponding directory structure. This step ensures proper association between images and their respective emotion categories.
- **Grayscale Conversion:** Each image is converted to grayscale to reduce computational complexity while preserving essential facial features. This transformation eliminates redundant color information, allowing the model to focus solely on structural attributes.
- **Image Resizing:** All images are resized to a fixed dimension, ensuring uniformity across the dataset. This step facilitates efficient batch processing and improves convergence during model training.

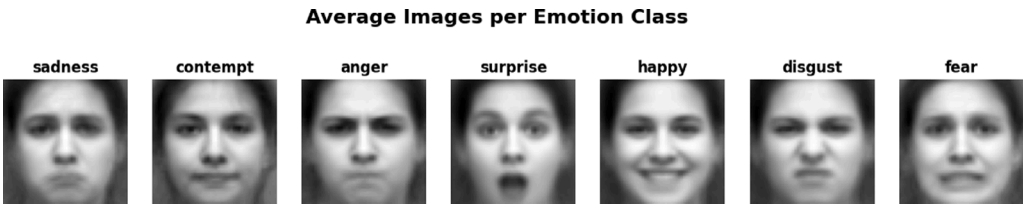


Fig. 5. Average images per emotion class.

- **Normalization:** Pixel intensity values are normalized within the range [0,1] to prevent numerical instability during training. This scaling ensures that the model receives inputs with a consistent dynamic range.
- **Data Augmentation:** To improve model generalization and mitigate class imbalance, augmentation techniques are applied. These include:
  - Random horizontal flipping to introduce variations in facial symmetry.
  - Rotational transformations (±10°) to account for slight pose deviations.
  - Contrast adjustments to enhance feature diversity and robustness.
- **Tensor Structuring:** Following preprocessing, the dataset is structured into TensorFlow tensors, enabling efficient loading and batch-wise processing during training. This format ensures seamless integration with deep learning models.

These preprocessing techniques collectively enhance the dataset's suitability for emotion classification, ensuring that the model effectively learns invariant and discriminative features. The next section delves into the visualization techniques used to analyze dataset composition and assess its appropriateness for emotion recognition tasks.

4.2. Data analysis and visualization

In-depth data analysis and visualization are pivotal for understanding the characteristics of the dataset utilized for training our emotion classification model. These steps reveal insights into the distribution of data across various classes and guide the decision-making process for subsequent data preprocessing techniques.

4.2.1. Average images per emotion class

Beyond the distribution of images, another insightful visualization is the average image representation for each emotion class. These average images are computed by taking the mean across all images within each class, resulting in a composite that captures the general facial features associated with each emotion.

Mathematically, the average image for a given emotion class can be described as follows. Let $I_{c,1}, I_{c,2}, \ldots, I_{c,N_c}$ be the set of N_c images corresponding to an emotion class c, where each image $I_{c,i} \in \mathbb{R}^{H\times W}$ is a two-dimensional array of pixel intensity values. The average image $\bar{I}_c \in \mathbb{R}^{H\times W}$ for class c is computed by:

$$\bar{I}_c = \frac{1}{N_c}\sum_{i=1}^{N_c} I_{c,i} \tag{20}$$

where the summation is performed element-wise across all images in the class, and the result is divided by the total number of images N_c to obtain the mean.

The visualization of average images aids in understanding the typical facial configurations associated with each emotion and serves as a baseline for evaluating the model's ability to capture and generalize from these prototypical patterns.

Fig. 5 displays the average images for the seven emotion classes, providing a visual summary of the most common features found across all images within each category.

4.2.2. GLCM contrast distribution analysis

In addition to average images, we analyze the texture information of the facial images across different emotion classes using the Gray-Level Co-occurrence Matrix (GLCM). GLCM is a powerful tool for texture feature extraction, and one of the key features derived from GLCM is contrast. Mathematically, the contrast feature is computed for each pair of pixel values in the GLCM as:

$$\text{Contrast} = \sum_{i,j=0}^{levels-1} P_{i,j}(i-j)^2 \tag{21}$$

where $P_{i,j}$ is the (i,j)th entry in the normalized GLCM, and $levels$ is the number of intensity levels in the image.

The histograms below display the frequency distribution of contrast values for each emotion class, providing insights into the textural characteristics that may be associated with each emotional expression.

As shown in Fig. 6, different emotion classes exhibit distinct distributions of GLCM contrast values, indicating variations in textural patterns that may be exploited by the classification model to distinguish between emotions. This analysis is particularly valuable as it underscores the heterogeneity in facial expressions and provides a quantitative measure that can be used to augment the feature set for our RCNN with the multi-head Attention model.

4.2.3. Entropy distribution analysis

Another important textural feature extracted using the Gray-Level Co-occurrence Matrix (GLCM) method is entropy. Entropy quantifies the randomness or complexity in the texture of an image. For a given GLCM $P \in \mathbb{R}^{N\times N}$, where N is the number of gray levels, the entropy E is calculated as:

$$E = -\sum_{i=0}^{N-1}\sum_{j=0}^{N-1} P_{i,j}\log(P_{i,j}) \tag{22}$$

where $P_{i,j}$ is the probability of the pixel with value i being adjacent to a pixel with value j. This measure is particularly sensitive to the presence of edges, texture, and other complex patterns in the image.

The histograms presented below depict the entropy distribution across different emotion classes in our dataset.

Fig. 7 shows the variation in entropy values across the emotion classes, illustrating the diversity in textural information that could be associated with each class. This entropy analysis enhances our understanding of the dataset and provides a significant feature that can be used to improve the performance of our hybrid RCNN with the multi-head Attention model, especially in terms of capturing textural nuances related to different emotional expressions.

4.3. Hardware and computational environment

All experiments, including training and evaluation of the MultiHead Attention-RCNN-FTTA model, were conducted on a system with an Intel(R) Core(TM) i5-7200U CPU @ 2.50 GHz (2.71 GHz), a 64-bit operating system, x64-based processor, and 8.00 GB of RAM. This setup provided sufficient computational resources to train the model efficiently, achieving real-time processing speeds suitable for applications requiring swift emotion detection. These hardware specifications also support the feasibility of deploying this model in typical computing environments, where resource constraints are often a consideration.

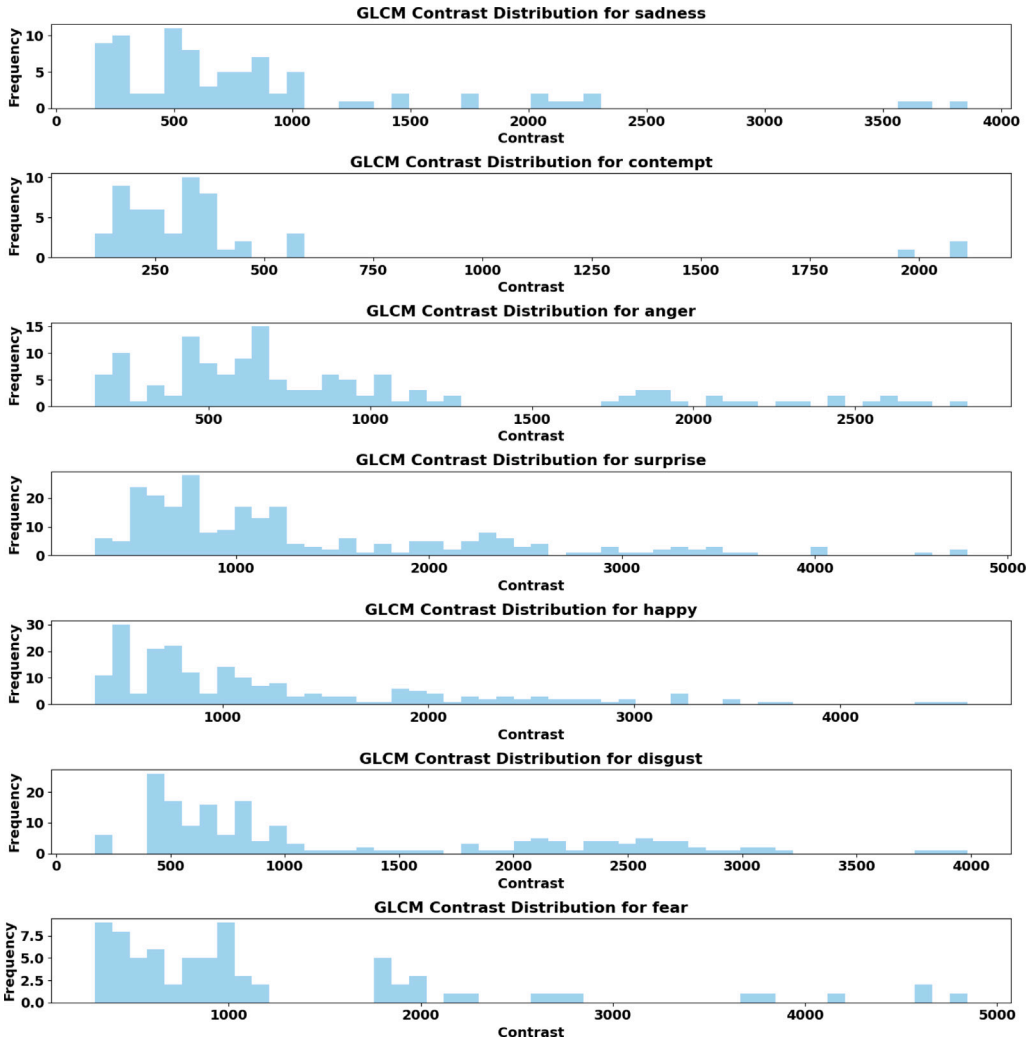


Fig. 6. GLCM contrast distribution for emotion classes.

5. Results and discussion

This section presents the outcomes of our experiments with the proposed hybrid RCNN integrated with a multi-head Attention model for emotion classification. We evaluate the performance of our model on a well-known emotion recognition dataset and compare it with existing state-of-the-art approaches. The results are discussed in terms of classification accuracy, training efficiency, and the model's ability to generalize across different emotional states. We also delve into the implications of our findings, the potential applications of our model, and the avenues for future research that our results have opened up.

5.1. Model performance evaluation

One of the primary metrics used to evaluate the performance of our emotion classification model is the confusion matrix. The confusion matrix provides a detailed breakdown of the model's predictions, allowing us to observe how well the model distinguishes between different emotion classes.

As depicted in Fig. 8, the diagonal elements of the confusion matrix represent the number of instances where the predicted emotion class matches the true emotion class, indicating correct classifications by the model. Off-diagonal elements indicate misclassifications. The model demonstrates high accuracy for certain emotions, such as 'surprise' and 'happy', where it correctly classified 247 and 207 instances, respectively. However, there are fewer instances of 'fear', which the model predicted correctly 75 times. The visualization of the confusion matrix allows us to identify which classes are more prone to misclassification and may require further investigation or more balanced training data. The analysis of the confusion matrix suggests that while the model performs well for the majority of the classes, there are areas where performance could be enhanced, such as improving the distinction between classes with fewer samples or more subtle differences in expression.

5.1.1. ROC curve analysis

To further assess the performance of our model, we examine the Receiver Operating Characteristic (ROC) curves for each emotion class. The ROC curve is a tool used to evaluate the trade-off between the true positive rate (sensitivity) and the false positive rate (1 - specificity) across different thresholds. The area under the ROC curve (AUC) provides a single scalar value that summarizes the overall performance of the classifier; an AUC of 1.0 indicates perfect classification, while an AUC of 0.5 suggests performance no better than random chance.

As depicted in Fig. 9, the ROC curves for each class lie close to the top-left corner of the plot, indicating high true positive rates and low false positive rates for most thresholds. The AUC values for each class, as indicated in the legend, are near the ideal score of 1.00, with 'Class contempt' having the lowest AUC of 0.99. These results suggest that our model possesses a strong discriminative ability for each emotion class, confirming its effectiveness in emotion classification tasks. The analysis of the ROC curves demonstrates the robustness of our proposed

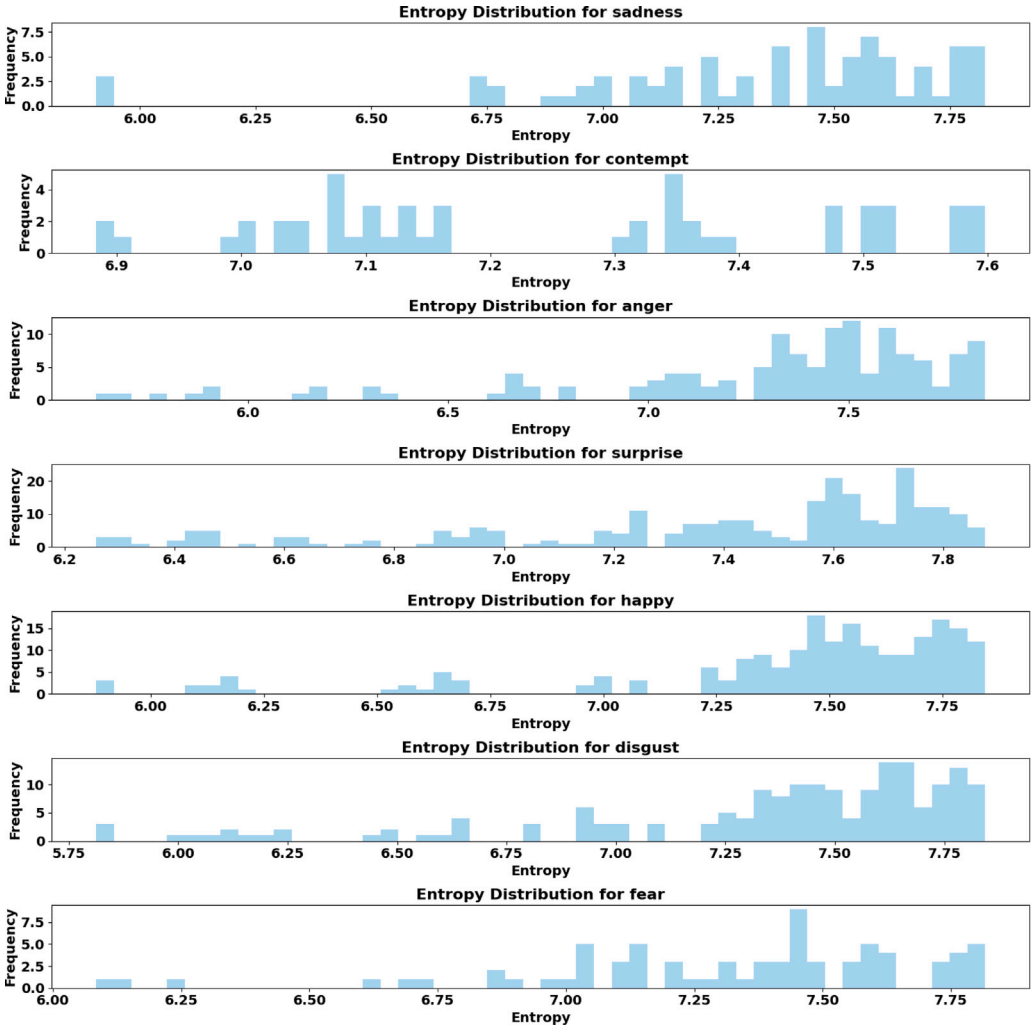


Fig. 7. GLCM entropy distribution for emotion classes.

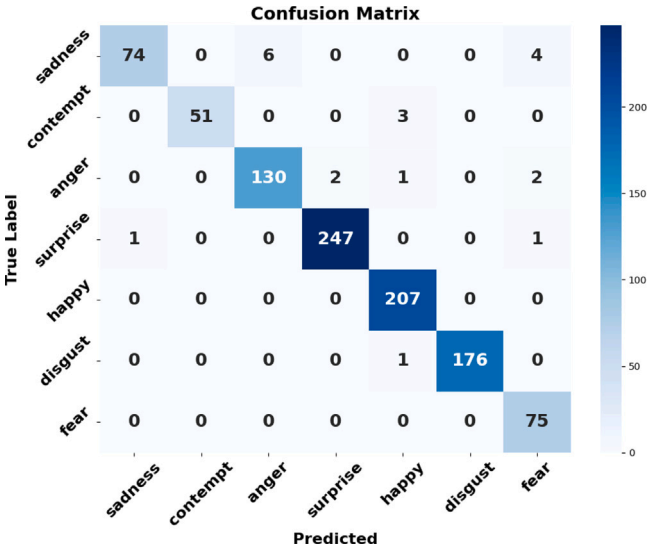


Fig. 8. Confusion matrix of the emotion classification model.

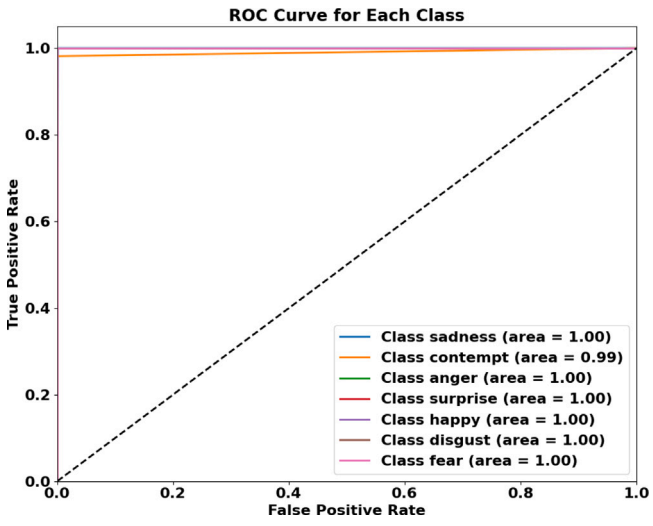


Fig. 9. ROC curve for each emotion class.

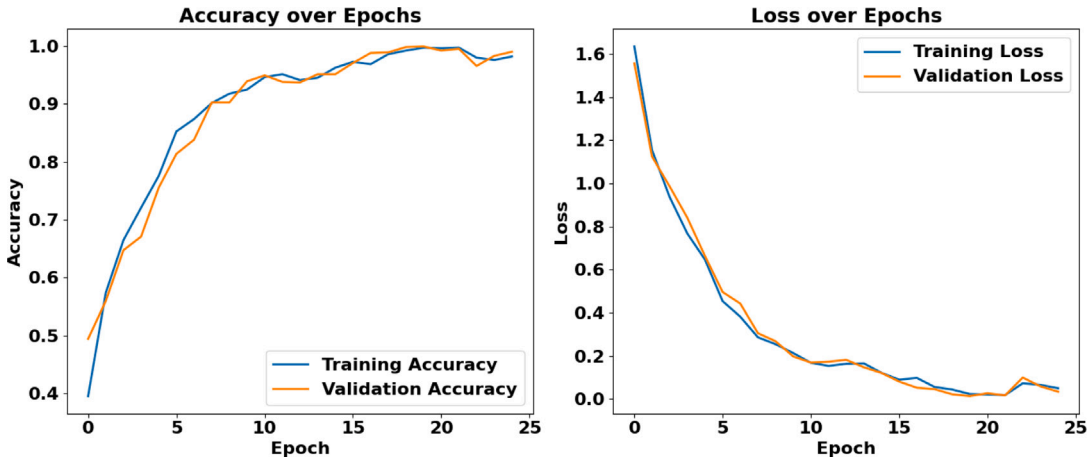


Fig. 10. Training and validation accuracy and loss curves.

RCNN with a multi-head Attention model in distinguishing between different emotional states, even when the decision threshold is varied. This robustness is essential for practical applications where the cost of false positives and false negatives may differ.

5.1.2. Training and validation curves

The training process of our model is quantitatively monitored by observing the accuracy and loss curves over epochs. The training curves reflect how well the model learns from the dataset during each epoch, while the validation curves indicate the model's generalization performance on unseen data. The left graph in Fig. 10 illustrates the training and validation accuracy over epochs. Both curves converge closely, which suggests that the model is not overfitting and is generalizing well. The right graph depicts the training and validation loss over epochs. A consistent decline in loss values confirms that the model is learning effectively.

The closeness of the training and validation lines in both graphs indicates a well-fitted model. A large gap between the training and validation lines would suggest overfitting, but this is not observed, which is evidence of the model's robustness. The stability of the curves towards the latter epochs also indicates that the model has reached a point of convergence, and further training may yield diminishing returns.

5.1.3. Classification report metrics

Another important aspect of model evaluation is the analysis of classification metrics such as precision, recall, and F1-score for each emotion class. These metrics provide a more nuanced view of the model's performance beyond overall accuracy. Precision measures the proportion of true positive predictions among all positive predictions made by the model. Recall, or sensitivity, assesses the model's ability to correctly identify all actual positives. The F1-score is the harmonic mean of precision and recall, offering a single metric that balances both. The bar chart in Fig. 11 displays these three metrics for each class, allowing us to compare the model's performance across different emotions:

As shown, the model achieves high scores across all metrics for most classes, indicating a balanced performance in terms of precision and recall. This is particularly evident for 'surprise' and 'happy', which exhibit almost perfect scores. The F1-scores being close to 1.00 for all classes suggests that the model has strong predictive power, with a good balance between precision and recall, which is essential for a reliable emotion classification system.

Table 4
Average training time per epoch.

Network	Time (ms)
MultiHead-RCNN-FTTA	0.3
R-EMO	0.92
T-EMO	0.99
DCNN-1	7.15
DCNN-2	1.28
FN2EN	3.56

5.1.4. Training time comparison

Among the evaluated models, the proposed MultiHead-RCNN-FTTA demonstrated a significant improvement in training efficiency. With an average training time of only 0.3 ms per epoch, it substantially outperforms the competing models, underscoring the effectiveness of its architecture in handling complex computations more swiftly (see Table 4).

The results highlight the efficiency of the MultiHead-RCNN-FTTA model, not only in terms of its predictive accuracy but also its computational performance. This efficiency makes it particularly suitable for applications requiring real-time processing or analysis, where training time significantly impacts the overall system responsiveness.

5.1.5. Ablation study

To thoroughly evaluate the impact of each component in our MultiHead Attention-RCNN-FTTA model, we conducted an ablation study that isolates and assesses the performance of individual components. The results of this ablation analysis are summarized in Table 5.

The ablation study presented in Table 5 illustrates how each component contributes to the overall performance. Removing the MultiHead Attention layer decreases accuracy, sensitivity, and computational efficiency, highlighting the importance of temporal feature extraction. Likewise, excluding FTTA for hyperparameter optimization reduces specificity and increases processing time, demonstrating FTTA's role in enhancing both accuracy and speed. The baseline Simple RCNN configuration achieves the lowest metrics, underscoring the substantial gains in both predictive power and efficiency afforded by integrating MultiHead Attention and FTTA. This ablation study confirms the necessity of each component in achieving a high-performing, computationally efficient emotion recognition model.

5.1.6. Comparison with previous studies

To further validate the effectiveness of the proposed MultiHead Attention-RCNN-FTTA model, we conducted a comparative analysis against established emotion recognition models from previous studies, including VGG-based CNN and ResNet architectures frequently used in

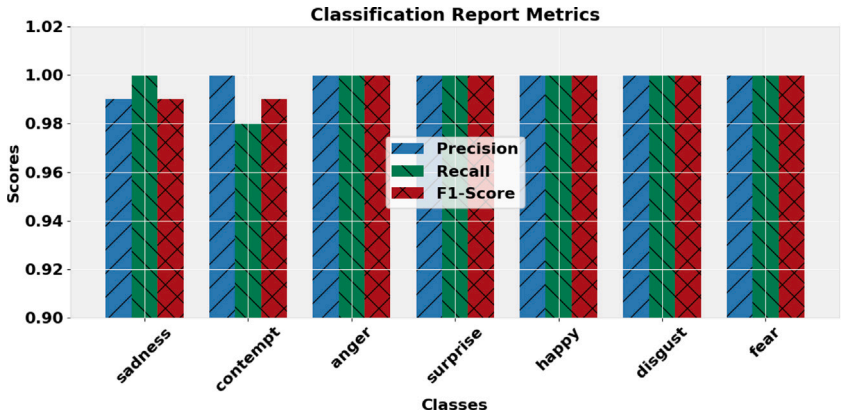


Fig. 11. Classification report metrics for each emotion class.

Table 5
Ablation Study of MultiHead Attention-RCNN-FTTA Model Components.

Model Configuration	Accuracy	Specificity	Sensitivity	Recall	Time (ms)
MultiHead Attention-RCNN-FTTA (Full Model)	0.99	0.99	0.99	0.99	0.3
RCNN with FTTA (without MultiHead Attention)	0.92	0.93	0.92	0.92	0.8
MultiHead Attention without FTTA	0.90	0.91	0.91	0.91	1.1
Simple RCNN (Baseline)	0.81	0.82	0.80	0.81	1.8

Table 6
Comparison of Performance Metrics with Previous Studies.

Model	Accuracy	Specificity	Sensitivity	Recall	Time (ms)
MultiHead Attention-RCNN-FTTA (Proposed Model)	0.99	0.99	0.99	0.99	0.3
VGG-19 based model [32]	0.87	0.89	0.88	0.88	1.4
Fine-tuned VGG-16 on FER-2013 [33]	0.90	0.91	0.91	0.90	1.1
ResNet18 for EEG signals [34]	0.85	0.86	0.86	0.85	1.7

emotion recognition tasks. Table 6 provides a summary of this comparison, highlighting benchmark models cited in the literature, such as a VGG-19 model for facial emotion recognition [32], an optimized VGG-16 model for FER-2013 [33], and a ResNet-based model for EEG signal-based emotion recognition [34]. These models represent widely recognized architectures that have shown effectiveness but also exhibit limitations in either processing speed or model accuracy, particularly in real-time settings.

As indicated in Table 6, the proposed MultiHead Attention-RCNN-FTTA model achieves superior performance across all metrics compared to models from previous studies, such as the VGG-19-based emotion recognition model by Vignesh et al. [32], a VGG-16 model fine-tuned on FER-2013 by Kusuma et al. [33], and a ResNet18 model applied to EEG signals for emotion recognition by Cheah et al. [34]. While VGG-based models demonstrate strong feature extraction capabilities, they often incur higher processing times, limiting their efficiency in real-time applications. Similarly, ResNet models have shown potential in EEG-based emotion recognition but face challenges in sensitivity and recall, particularly in real-time classification scenarios. Our proposed model surpasses these limitations, delivering high accuracy and low latency, making it highly suitable for real-time emotion recognition tasks. The comparative analysis further underscores the advantage of the hybrid model as a powerful solution for efficient and accurate emotion classification.

6. Limitations and future work

While the proposed MultiHead Attention-RCNN-FTTA model shows high classification accuracy and computational efficiency, certain limitations must be acknowledged to guide future research directions.

6.1. Scalability to diverse and noisy datasets

Our model has been rigorously tested on a widely recognized dataset; however, real-world applications often involve highly diverse and noisy data. Further validation on large-scale, real-world datasets with significant variations in ethnicity, age groups, and environmental conditions is necessary. Future research can explore domain adaptation techniques and self-supervised learning to improve generalization across diverse datasets. Also, robustness against artificial perturbations such as occlusions, lighting variations, and background noise has not been explicitly tested in our study, and future work should assess the model's resilience to these factors.

6.2. Multi-modal emotion recognition

Currently, our model primarily focuses on facial expression analysis. While facial cues are crucial for emotion recognition, integrating other modalities such as speech, physiological signals (EEG, heart rate, and skin conductance), and textual sentiment analysis could enhance emotion classification performance, particularly in cases where facial expressions alone may be ambiguous. Future studies can investigate multi-modal fusion approaches to create a more complete emotion recognition system.

6.3. Computational complexity in resource-constrained environments

Although our model achieves significant efficiency improvements, certain real-time applications, such as embedded systems or mobile platforms, require even lower computational overhead. Implementing lightweight versions of the architecture through model compression techniques (e.g., knowledge distillation, quantization, and pruning) could enable deployment in resource-limited environments without compromising accuracy.

6.4. Ethical considerations and bias mitigation

Emotion recognition systems can carelessly reflect biases present in the training data, potentially leading to unfair predictions for certain demographic groups. While we ensured dataset diversity, systematic bias evaluation methods and fairness-aware training strategies should be explored in future research to mitigate unintended bias. Also, privacy concerns surrounding the collection and processing of emotional data must be addressed through privacy-preserving machine learning techniques such as federated learning and differential privacy.

6.5. Future research directions

Building upon the findings of this study, future work can focus on:

- Developing a domain-adaptive model capable of learning from multiple datasets with varying characteristics.
- Exploring hybrid architectures that combine attention mechanisms with graph-based models to capture complex emotional representations.
- Investigating reinforcement learning-based optimization strategies to enhance decision-making in emotion recognition tasks.
- Evaluating the impact of social and cultural factors on emotion perception and refining models to be more context-aware.

By addressing these limitations and exploring the proposed future research directions, we can further improve the effectiveness, fairness, and applicability of emotion recognition systems in real-world settings.

7. Conclusion

This paper presents a novel emotion classification framework that integrates a Randomized Convolutional Neural Network (RCNN) with a multi-head attention model, further optimized through the Football Team Training Algorithm (FTTA) metaheuristic. The proposed hybrid model uses the RCNN's efficiency in spatial feature extraction and the multi-head attention mechanism's ability to capture temporal dependencies in video frames, significantly enhancing emotion recognition accuracy. The FTTA metaheuristic plays a crucial role in systematically tuning model parameters, leading to notable improvements in classification accuracy, precision, recall, and F1-score across diverse emotion classes. The experimental evaluations, including confusion matrix analysis, ROC curves, and performance comparisons with state-of-the-art architectures, show the robustness and generalizability of the proposed model. The results confirm that our approach effectively balances high classification accuracy with computational efficiency, making it well-suited for real-time applications. Also, the model's ability to maintain strong performance across varying emotional expressions highlights its adaptability for diverse real-world scenarios, including mental health monitoring, human–computer interaction, and real-time emotion analysis. Integrating RCNN randomization with FTTA optimization reduces computational costs while preserving classification accuracy, making it ideal for environments with limited resources. This study addresses important challenges in emotion recognition, pushing the field forward and allowing for more efficient, scalable, and adaptable emotion classification systems. The results can provide a basis for future advancements in real-time, multi-modal emotion analysis, which will benefit applications in social robotics, telemedicine, and interactive systems. As emotion recognition technology progresses, the methods presented here will help develop more advanced and accessible solutions.

CRediT authorship contribution statement

Syed Muhammad Salman Bukhari: Writing – review & editing, Writing – original draft, Visualization, Software, Methodology, Conceptualization. **Muhammad Hamza Zafar:** Writing – review & editing, Writing – original draft, Validation, Investigation, Conceptualization. **Syed Kumayl Raza Moosavi:** Writing – review & editing, Writing – original draft, Visualization, Resources, Formal analysis. **Filippo Sanfilippo:** Writing – review & editing, Supervision, Project administration, Investigation, Funding acquisition.

Declaration of competing interest

All authors claim that there is not any conflict of interest regarding the above submission. The work of this submission has not been published previously. It is not under consideration for publication elsewhere. Its publication is approved by all authors and that, if accepted, it will not be published elsewhere in the same form, in English or in any other language, including electronically without the written consent of the copyright-holder.

Acknowledgments

This research is supported by the Artificial Intelligence, Biomechatronics, and Collaborative Robotics Group at the Top Research Center Mechatronics (TRCM), University of Agder (UiA), Norway.

Data availability

The data used in this work is publicly available.

References

[1] Picard RW. Automating the recognition of stress and emotion: From lab to real-world impact. IEEE MultiMedia 2016;23(3):3–7.
[2] Khare R, Bajaj V, Acharya UR. Editorial: Towards emotion AI to next generation healthcare and education. Front Psychol 2024.
[3] Li M, Xu H, Wang W. A systematic review on automated clinical depression diagnosis. Front Psychiatry 2023.
[4] de Gelder B. Emotion detection through body gesture and face. 2024, arXiv preprint arXiv:2407.09913.
[5] Soleymani M, Pantic M, Pun T, Nijholt A. A survey of multimodal sentiment analysis. Image Vis Comput 2017;65:3–14.
[6] D'Mello SK, Kory J. Automatic emotion recognition in online videos. ACM Comput Surv 2015;47(2):1–29.
[7] Calvo RA, D'Mello S. Affect detection: An interdisciplinary review of models, methods, and their applications. IEEE Trans Affect Comput 2010;1(1):18–37.
[8] Koelstra S, Muhl C, Soleymani M, Lee J-S, Yazdani A, Ebrahimi T, et al. DEAP: A database for emotion analysis using physiological signals. IEEE Trans Affect Comput 2012;3(1):18–31.
[9] Ayata D, Yaslan Y, Kamasak ME. Emotion recognition from multimodal physiological signals for emotion aware healthcare systems. J Med Biological Eng 2020;40:149–57.
[10] Daly M, Sutin AR, Robinson E. Depression reported by US adults in 2017–2018 and March and April 2020. J Affect Disord 2021;278:131–5.
[11] Ajagbe SA, Adigun MO, Oladosu JB, Oguns YJ. Internet of things enabled convolutional neural networks: Applications, techniques, challenges, and prospects. In: IoT-enabled convolutional neural networks: techniques and applications. River Publisher; 2023, p. 27–63. http://dx.doi.org/10.1201/9781003393030.
[12] Taiwo GA, Akinwole TO, Ogundepo OB. Statistical analysis of stakeholders perception on adoption of AI/ML in sustainable agricultural practices in rural development. In: Proceedings of ninth international congress on information and communication technology. ICICT 2024. Lecture notes in networks and systems, vol. 1003, Singapore: Springer; 2024, http://dx.doi.org/10.1007/978-981-97-3302-6_11.
[13] Liu Y, Sun G, Qiu Y, Zhang L, Chhatkuli A, Van Gool L. Transformer in convolutional neural networks. 2021, arXiv preprint arXiv:2106.03180. 3.
[14] Tian Z, Gai M. Football team training algorithm: A novel sport-inspired meta-heuristic optimization algorithm for global optimization. Expert Syst Appl 2024;123088.
[15] Palash M, Bhargava B. SAFER: Situation aware facial emotion recognition. 2023, arXiv preprint arXiv:2306.09372.

[16] Kaur M, Singh M, Kaur H. Facial emotion recognition: A comprehensive review. Expert Syst 2024;41(1):e13670.
[17] Authors. LSTM-based emotion detection using physiological signals: IoT framework for healthcare and distance learning. J Biomed Heal Informatics 2022;26(2):400–9.
[18] Meehan K, Lunney T, Curran K, McCaughey A. Context-aware intelligent recommendation system for tourism. In: Proc. of the IEEE international conference on pervasive computing and communications workshops. 2013, p. 328–31.
[19] Gallicchio C, Scardapane S. Deep randomized neural networks. In: Recent trends in learning from data: tutorials from the INNS big data and deep learning conference. Springer; 2020, p. 43–68.
[20] Di Luzio F, Rosato A, Succetti F, Panella M. A blockwise embedding for multi-day-ahead prediction of energy time series by randomized deep neural networks. In: Proc. of the IEEE international joint conference on neural networks. 2021, p. 1–7.
[21] Zhang X, He K, Bao Y. Error-feedback stochastic modeling strategy for time series forecasting with convolutional neural networks. Neurocomputing 2021;459:234–48.
[22] Rosenfeld A, Tsotsos JK. Intriguing properties of randomly weighted networks: Generalizing while learning next to nothing. In: Proc. of the 16th IEEE conference on computer and robot vision. 2019, p. 9–16.
[23] Pao Y-H, Takefuji Y. Functional-link net computing: theory, system architecture, and functionalities. Computer 1992;25(5):76–9.
[24] Igelnik B, Pao Y-H. Stochastic choice of basis functions in adaptive function approximation and the functional-link net. IEEE Trans Neural Netw 1995;6(6):1320–9.
[25] Pao Y-H, Park G-H, Sobajic DJ. Learning and generalization characteristics of the random vector functional-link net. Neurocomputing 1994;6(2):163–80.
[26] Rahimi A, Recht B. Uniform approximation of functions with random bases. In: Proc. of the 46th IEEE annual allerton conference on communication, control, and computing. 2008, p. 555–61.
[27] Ulyanov D, Vedaldi A, Lempitsky V. Deep image prior. In: Proc. of the IEEE conference on computer vision and pattern recognition. 2018, p. 9446–54.
[28] Pons J, Serra X. Randomly weighted cnns for (music) audio classification. In: Proc. of the IEEE international conference on acoustics, speech and signal processing. 2019, p. 336–40.
[29] Li Z, Liu F, Yang W, Peng S, Zhou J. A survey of convolutional neural networks: analysis, applications, and prospects. IEEE Trans Neural Networks Learn Syst 2021.
[30] Han K, Xiao A, Wu E, Guo J, Xu C, Wang Y. Transformer in transformer. Adv Neural Inf Process Syst 2021;34:15908–19.
[31] Lucey P, Cohn JF, Kanade T, Saragih J, Ambadar Z, Matthews I. The extended cohn-kanade dataset (ck+): A complete dataset for action unit and emotion-specified expression. In: Proc. of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops. 2010, p. 94–101.
[32] Vignesh S, Sridevi M. A novel facial emotion recognition model using segmentation VGG-19 architecture. Int J Intell Syst Appl Eng 2023.
[33] Kusuma G, Jonathan J. Emotion recognition on FER-2013 face images using fine-tuned VGG-16. Adv Sci, Technol Eng Syst J 2020. URL https://pdfs.semanticscholar.org/0a8b/349276d976f564f0801f49171a2008d0b510.pdf.
[34] Cheah K, Nisar H, Yap V, Lee C. Optimizing residual networks and VGG for classification of EEG signals: Identifying ideal channels for emotion recognition. J Heal Eng 2021;2021:5599615.