Facial expression recognition for emotional state identification using deep convolutional neural network

Abdelhakim Gharbi[1], Abdeljalil Gattal[2], Issam Bendib[1]
[1]LAMIS Laboratory, Department of Computer Science, Faculty of Exact Sciences, Nature and Life Sciences, Echahid Cheikh Larbi Tebessi University, Tebessa, Algeria
[2]Laboratoire de Vision et d'Intelligence Artificielle (LAVIA), Faculté des Sciences Exactes et Sciences de la Nature et de la Vie, Université Echahid Cheikh Larbi Tebessi, Tebessa, Algeria

Article Info

Article history:

Received Jul 10, 2025
Revised Oct 5, 2025
Accepted Dec 6, 2025

Keywords:

Convolutional neural
Data augmentation
Deep learning
Facial expression recognition
FER-2013
Networks
Transfer learning

ABSTRACT

Facial expressions represent one of the most significant forms of non-verbal communication, with psychologists identifying six universal expressions: happiness, sadness, surprise, anger, fear, and disgust. Recognizing these expressions presents considerable challenges due to the subtlety of facial movements and variations across individuals. This paper presents a deep learning-based system for facial expression recognition (FER) that employs convolutional neural networks (CNNs) to classify emotional states. We investigate both a novel CNN architecture developed from scratch and established transfer learning approaches, evaluating their performance on the FER-2013 dataset. Our experimental results demonstrate that the proposed custom CNN architecture achieves 72.93% accuracy when combined with comprehensive data augmentation techniques, outperforming several baseline models. The system shows particular strength in recognizing fundamental emotions while maintaining computational efficiency suitable for real-time applications.

Corresponding Author:

Abdelhakim Gharbi
LAMIS Laboratory, Department of Computer Science, Faculty of Exact Sciences
Nature and Life Sciences, Echahid Cheikh Larbi Tebessi University
Tebessa, Algeria
Email: abdelhakim.gharbi@univ-tebessa.dz

1. INTRODUCTION

Facial expressions serve as one of the most fundamental and universal channels of human communication, conveying rich emotional information that often transcends linguistic and cultural boundaries. Research by Mehrabian [1] suggests a significant portion of communication is non-verbal, with facial expressions being a critical component. Ekman's [2] foundational work established six basic emotions (happiness, sadness, surprise, anger, fear, and disgust) that are universally recognized, highlighting their evolutionary importance.

Facial expression recognition (FER) has become increasingly vital in fields ranging from security and user authentication to psychology and human-computer interaction. Its applications extend beyond traditional facial recognition by integrating emotional context, leading to more robust and insightful systems. The adoption of FER accelerated during the COVID-19 pandemic, finding use in digital learning to foster engagement and in marketing to gauge customer reactions.

FER systems bridge the communication gap between teachers and students, fostering better engagement in virtual classrooms. In marketing and commerce, real-time emotion detection enables businesses to gauge customer reactions through user images or videos, offering valuable insights for strategic decision-making. In psychology, FER aids in the analysis of human behavior and supports the early detection of psychological disorders. Furthermore, FER plays a vital role in human-agent and human-robot interactions, contributing to the development of robots and avatars capable of engaging in natural, emotionally aware exchanges.

The integration of FER into diverse fields including gaming, animation, robotics, behavioral sciences, and clinical practice highlights its growing utility. This work focuses on advancing FER, driven by its critical applications in both security and beyond. In security, FER is combined with other biometric technologies such as fingerprint or voice recognition to protect sensitive environments. Beyond security, FER improves human-machine interaction by enabling systems to understand and respond to users' emotions, a capability essential for the effectiveness of social robots and emotionally intelligent machines.

Early FER systems relied on handcrafted features like gray level co-occurrence matrix (GLCM) [3] or hybrid descriptors like ORB-LBP [4], achieving high accuracy in controlled settings but struggling with real-world challenges like lighting variations and occlusions. The advent of deep learning, particularly convolutional neural networks (CNNs), revolutionized the field by enabling end-to-end learning, automatic feature extraction, and robust performance across diverse conditions.

However, many state-of-the-art CNN models are computationally expensive, limiting their deployment in real-time, resource-constrained applications (e.g., mobile devices, embedded systems). This creates a clear research gap for models that maintain high accuracy while minimizing computational cost. While some efficient architectures like eXnet and POSTER++ have been proposed, there remains a need for models that offer a simpler, highly optimized design with a clear path to deployment.

This paper addresses this gap by introducing a novel, lightweight CNN architecture specifically designed for efficient and accurate FER. Our primary objective is to achieve competitive performance on the standard FER-2013 benchmark while ensuring low computational complexity. The key innovations and contributions of our work are:

- A streamlined CNN architecture with a carefully optimized number of layers, filters, and hyperparameters to reduce computational overhead (FLOPs and parameters) without sacrificing representational power.
- A comprehensive and custom data augmentation strategy that synthesizes challenging real-world variations (e.g., distortion, stretching, perspective changes) to significantly enhance model robustness and generalization.
- An extensive empirical evaluation demonstrating that our proposed model outperforms larger, more complex networks like VGG19 and ResNet-50 in terms of accuracy versus computational efficiency, making it particularly suitable for real-time applications.
- A detailed analysis of performance, including per-class metrics and a confusion matrix, to identify strengths and weaknesses, particularly in handling class imbalance and ambiguous expressions.

As illustrated in Figure 1, our complete pipeline integrates these advances into a cohesive system. The subsequent sections are structured as follows: section 2 offers a focused review of related work on deep learning-based FER. Section 3 details the proposed methodology. Section 4 presents and discusses the experimental results. Finally, section 5 concludes the paper and suggests future research directions.

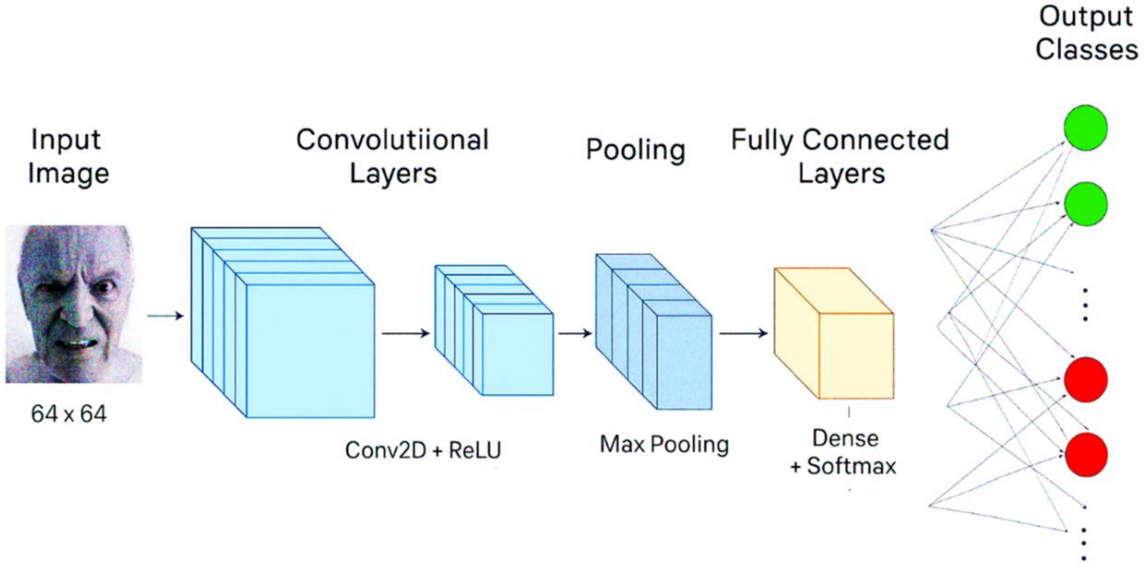

Figure 1. General pipeline of the proposed deep learning-based FER system

2. RELATED WORK

Deep learning is a powerful paradigm that allows for the acquisition of multi-layered hierarchical representations from training data. It relies on artificial neural networks, where the results of each layer serve as input for subsequent calculations. The advancement of deep learning is heavily dependent on computing power and the availability of large "big data" databases. In recent decades, deep learning algorithms for computer vision have made significant progress. These algorithms include CNN, such as AlexNet [6], which popularized convolutional networks, ZFnet [7], which improved upon AlexNet by modifying its architecture's hyper-parameters, ResNet [8], a residual network, and visual geometry group network architecture (VGGNet) [9]. These algorithms excel in tasks involving recognition, classification, and feature extraction.

The field of FER has evolved from classical methods based on handcrafted features to modern deep learning approaches. This section provides a critical overview of this evolution, focusing on CNN-based methods and highlighting the trend towards efficiency and robustness that motivates our work. Deep learning allows for the acquisition of multi-layered hierarchical representations from data. Advances in computing power and the availability of large datasets have driven progress in computer vision, led by architectures like AlexNet [6], ZFNet [7], ResNet [8], and VGGNet [9]. These models excel in recognition, classification, and feature extraction tasks.

In FER, CNNs are the dominant architecture. Kim *et al.* [10] used a hierarchical committee of diverse deep CNNs to achieve robust performance. Fan *et al.* [11] and Liu *et al.* [12] explored ensemble models, combining features or outputs from multiple CNNs to boost recognition rates beyond single models. Tang [13] replaced the standard Softmax layer with a linear support vector machine (SVM) within a CNN, showing performance improvements on FER tasks.

A significant challenge has been improving performance on difficult benchmarks like FER-2013. Pramerdorfer and Kampel [14] addressed architectural bottlenecks in basic CNNs, achieving 75.2% accuracy on FER-2013 without extra data. Minaee *et al.* [15] proposed an attentional convolutional network that focuses on crucial facial regions, yielding improvements across multiple datasets and providing visualization for model interpretability.

A key recent trend is the development of efficient architectures. Mollahosseini *et al.* [16] proposed a network with Inception modules for improved accuracy and training time. Addressing computational constraints more directly, Riaz *et al.* [17] proposed expression net (eXnet), emphasizing parallel feature extraction for improved accuracy with a reduced parameter count. Similarly, Ma and Celik [18] proposed FER-Net, a densely connected CNN, demonstrating encouraging performance. Chen *et al.* [19] showed the effectiveness of CNNs with specialized structures and batch normalization, achieving high accuracy suitable for real-time needs. Very recently, Chouhayebi *et al.* [20] combined deep features from VGG19 with dynamic texture features (HOG-TOP) and long short-term memory (LSTM) cells, achieving high accuracy on the INTERFACE’05 dataset. Gharbi *et al.* [21] and Abdulsattar and Hussain [22] also contributed with efficient deep and hybrid (HOG+LBP+CNN) approaches, respectively.

The push for efficiency continues with newer models. Mao *et al.* [23] introduced POSTER++, a streamlined and efficient transformer-based model that outperformed complex counterparts, emphasizing simplicity, and generalization. Lü *et al.* [24] tackled dynamic FER with a multi-snippet spatiotemporal learning approach. Comprehensive surveys by Shehu *et al.* [25], Li and Deng [26], and Kopalidis *et al.* [27] provide broad overviews of methodologies, challenges, and recent advances, highlighting the shift towards lightweight, interpretable, and temporally-aware models.

Despite these advances, many high-performing models remain computationally heavy, and others sacrifice too much accuracy for gains in efficiency. Our work is positioned within this landscape. We propose a lightweight CNN that is not only efficient but also achieves highly competitive accuracy on FER-2013. Unlike very deep or ensemble models, our architecture is simple and designed for easy deployment, and unlike some lightweight models, it maintains strong performance through careful architectural optimization and aggressive data augmentation. A comparative summary of key works is presented in Table 1, now including metrics relevant to efficiency (parameters, FLOPs where available).

Table 1. Comprehensive comparative analysis of FER systems

Ref.	Approach	Dataset (s)	Input size	Params (M)	FLOP (GMac)	Accuracy (%)	Key innovation
Kim *et al.* (2016) [10]	Two-level hierarchical committee of deep CNNs	SFEW2.0 FER-2013 TFD GENKI-4K	224×224 48×48 96×96 64×64	-	-	61.6 72.72 87.71 95.38	Multi-scale feature fusion
Fan *et al.* (2020) [11]	VGG19 ResNet18 CNN ensemble (VGG19+ResNet18)	FER-2013	224×224	138 11.7 149.7	15.5 1.8 17.3	72.77 72.69 73.14	Late fusion ensemble
Liu *et al.* (2016) [12]	CNN ensemble	FER-2013	48×48	-	-	65.03	Feature diversity
Tang (2013) [13]	CNN with SVM classifier	FER-2013	64×64	1.8	0.95	71.20	Hybrid deep/shallow
Pramerdorfer and Kampel (2016) [14]	VGGNet Inception ResNet CNN ensembles	FER-2013	48×48 299×299 224×224 48×48	-	-	72.7 71.6 72.4 75.2	Architecture diversity
Minaee *et al.* (2021) [15]	Attentional convolutional network	FER-2013 FERG JAFFE CK+	48×48 128×128 256×256 640×480	4.2	1.9	70.02 99.3 92.8 98.0	Spatial attention
Krizhevsky *et al.* (2012) [6]	AlexNet	FER-2013	227×227	61	0.72	61.10	Pioneering deep CNN
Mollahosseini *et al.* (2016) [16]	Deep CNN+Inception	FER-2013 MultiPIE MMI CK+ DISFA GEMEP-FERA SFEW	48×48 128×128 256×256 640×480 128×128 96×96 224×224	6.8	2.8	66.4 94.7 77.9 93.2 55.0 76.7 47.7	Inception modules
Riaz *et al.* (2020) [17]	eXnet (lightweight CNN)	FER-2013 CK+ RAF-DB	48×48 256×256 100×100	~0.5	~0.2	73.54 96.75 86.37	Parallel feature extraction
Ma and Celik (2019) [18]	Densely connected DCNN(FER-Net)	FER-2013	48×48	7.2	3.0	66.54	Dense connectivity
Chen *et al.* (2017) [19]	CNN with batch normalization	CK+	640×480	-	-	98.15	Bn optimization
Chouhayebi *et al.* (2023) [20]	VGG19+HOG-TOP+LSTM	INTERFACE'05	224×224	>20	>15	98.44	Spatio-temporal fusion
Gharbi *et al.* (2024) [21]	Optimized Deep CNN	FER-2013	48×48	3.12	1.45	72.80	Data augmentation
Abdulsattar and Hussain (2022) [22]	HOG+LBP+CNN hybrid	CK+	256×256	-	-	97.56	Handcrafted+deep fusion
Mao *et al.* (2025) [23]	POSTER++(enhanced transformer)	FER-2013 AffectNet	224×224	~15	~3.5	74.6 89.1	Patch-based attention
Lü *et al.* (2025) [24]	Multi-Snippet Spatiotemporal learning	AFEW DFEW	112×112	-	-	70.1 76.3	Temporal snippets
Our model (optimal)	**Proposed Lightweight CNN**	FER-2013	64×64	3.04	1.42	72.93	Architectural and augmentation optimization
Our model (baseline)	**Proposed Lightweight CNN**	FER-2013	48×48	2.91	1.18	70.08	Architectural and augmentation optimization

3. METHOD

The proposed FER system is built upon a CNN architecture and involves a systematic pipeline that includes preprocessing, defining the CNN model, compiling and training the model, and employing data augmentation to enhance its robustness.

In the preprocessing stage, the input images are resized to dimensions of 48×48 pixels and the pixel values are normalized to accelerate convergence during training. Additionally, face alignment is performed to ensure consistent orientation and focus on meaningful facial features. Data augmentation techniques such as rotation (±15°), flipping (horizontal), shifting (±10% width/height), and zooming (±5%) are applied using the Keras ImageDataGenerator. Furthermore, our custom augmentation pipeline (detailed in section 3.1) applies more aggressive geometric transformations to increase the variability in the training dataset, making the model more robust to real-world variations.

The CNN architecture is designed to be lightweight yet effective, progressively learning feature representations from simple edges to complex facial patterns. The model is structured in multiple blocks. The initial number of filters f is a key hyperparameter that was optimized through a grid search (see section 4); the optimal value was found to be *f*=48. This value provides a sufficient receptive field for feature extraction at the 64x64 input resolution without introducing excessive parameters. All subsequent layers scale from this base value $(f \times 2, f \times 4, and\ f \times 8)$.

In the first block, the model applies two convolutional layers with f filters. A larger kernel size of 5×5 is used in the first layer to capture broader contextual features, followed by a 3×3 kernel for more detailed pattern extraction. Subsequent blocks (Block 2, Block 3, and Block 4) increase the filter count, doubling them at each step (96, 192, and 384 filters) to compensate for the reduction in spatial dimensions and to learn more complex features. Each convolutional layer uses the rectified linear unit (ReLU) activation function for non-linear transformations and "same" padding to preserve spatial dimensions. After each block, batch normalization stabilizes training by normalizing activations, while dropout layers with progressively increasing rates (0.3 in early layers to 0.5 in deeper layers) prevent overfitting by randomly deactivating neurons during training. These hyperparameters were chosen based on common practices and validated empirically.

The feature maps generated by the convolutional layers are flattened into a one-dimensional vector and passed through fully connected (dense) layers to perform high-level reasoning. These dense layers consist of 128, 256, and 1024 neurons, each activated with the ReLU function. The larger final dense layer (1024 neurons) was chosen to provide sufficient capacity for the high-level feature mapping required before the final classification, a design choice that was validated during hyperparameter tuning. A final dense layer with a SoftMax activation function outputs probabilities for each class, corresponding to the seven universal facial expressions.

The model is compiled using the Adam optimizer with a learning rate of 0.001, which adapts the learning rate during training for faster convergence. The loss function used is categorical cross-entropy, appropriate for multi-class classification tasks, with accuracy as the evaluation metric. The training process involves splitting the dataset into training, validation, and test sets. The model is trained on augmented data over several epochs, typically 50 to 100, with a batch size of 32 or 64. Validation is used to monitor overfitting and fine-tune hyperparameters. All experiments were conducted using TensorFlow 2.8 and Keras 2.8.0 on an NVIDIA RTX 3080 GPU. A fixed random seed (42) was used for NumPy and TensorFlow to ensure the reproducibility of results. Table 2 outlines the structure of your proposed CNN model, including the filter sizes, output shapes, and additional details like dropout rates and batch normalization layers.

Table 2. Optimized CNN architecture specification

Layer no.	Block	Layer type	Filter	Kernel size	Activation	Output shape	Other details
1	1	Input layer	-	-	-	(48, 48, 1)	Input size: 48×48×1
2	1	Convolution (Conv2D)	*f*	5×5	ReLU	(48, 48, *f*)	Padding: "same"
3	1	Convolution (Conv2D)	*f*	3×3	ReLU	(48, 48, *f*)	Padding: "same"
4	1	Batch normalization	-	-	-	(48, 48, *f*)	Normalizes activations
5	1	MaxPooling	-	2×2	-	(24, 24, *f*)	Reduces spatial dimensions
6	1	Dropout	-	-	-	(24, 24, *f*)	Dropout rate: 0.3
7	2	Convolution (Conv2D)	*f*×2	3×3	ReLU	(24, 24, *f*×2)	Padding: "same"
8	2	Convolution (Conv2D)	*f*×2	3×3	ReLU	(24, 24, *f*×2)	Padding: "same"
9	2	Convolution (Conv2D)	*f*×2	3×3	ReLU	(24, 24, *f*×2)	Padding: "same"
10	2	Batch normalization	-	-	-	(24, 24, *f*×2)	Normalizes activations
11	2	MaxPooling	-	2×2	-	(12, 12, *f*×2)	Reduces spatial dimensions
12	2	Dropout	-	-	-	(12, 12, *f*×2)	Dropout rate: 0.5
13	3	Convolution (Conv2D)	*f*×4	3×3	ReLU	(12, 12, *f*×4)	Padding: "same"
14	3	Convolution (Conv2D)	*f*×4	3×3	ReLU	(12, 12, *f*×4)	Padding: "same"
15	3	Convolution (Conv2D)	*f*×8	3×3	ReLU	(12, 12, *f*×8)	Padding: "same"
16	3	Batch Normalization	-	-	-	(12, 12, *f*×8)	Normalizes activations
17	3	MaxPooling	-	2×2	-	(6, 6, *f*×8)	Reduces spatial dimensions
18	3	Dropout	-	-	-	(6, 6, *f*×8)	Dropout rate: 0.5
19	4	Convolution (Conv2D)	*f*×8	3×3	ReLU	(6, 6, *f*×8)	Padding: "same"
20	4	Convolution (Conv2D)	*f*×8	3×3	ReLU	(6, 6, *f*×8)	Padding: "same"
21	4	Convolution (Conv2D)	*f*×8	3×3	ReLU	(6, 6, *f*×8)	Padding: "same"
22	4	Batch normalization	-	-	-	(6, 6, *f*×8)	Normalizes activations
23	4	MaxPooling	-	2×2	-	(3, 3, *f*×8)	Reduces spatial dimensions
24	4	Dropout	-	-	-	(3, 3, *f*×8)	Dropout rate: 0.5
25	/	Flatten	-	-	-	3456	Converts 3D tensor to 1D
26	/	Dense	128	-	ReLU	128	Fully connected layer
27	/	Dense	256	-	ReLU	256	Fully connected layer
28	/	Dense	1024	-	ReLU	1024	Fully connected layer
29	/	Batch normalization	-	-	-	1024	Normalizes activations
30	/	Dropout	-	-	-	1024	Dropout rate: 0.5
31	/	Output layer (dense)	7	-	SoftMax	7	Outputs class probabilities

3.1. Data augmentation

The provided algorithm performs data augmentation to enhance the diversity of the training dataset, which improves the generalization ability of the CNN model. Data augmentation involves applying transformations to images in the dataset to artificially expand it, introducing variations that the model is likely to encounter in real-world scenarios. The transformations used in this algorithm include distortion, stretching, and perspective transformation [28], each implemented as separate functions.

- The distort function introduces random distortions in the image by slightly shifting the four corner points of the image to new positions within a predefined range threshold=0.25. These new corner points are used to compute a perspective transformation matrix, which is applied to the image to generate a distorted version.
- The stretch function randomly stretches or compresses the image by slightly shifting the vertical edges of the image by a factor of up to ±0.2 of the image width. This creates variations in the width of the image while preserving its overall content. Similar to the distortion process, the transformation is applied using a perspective transformation matrix.
- The perspective function adds a perspective effect to the image by simulating a change in the viewpoint. This is achieved by adjusting the vertical positions of the four corner points of the image by a factor of up to ±0.1. A new perspective transformation matrix is calculated based on these modified points, and the image is transformed accordingly.

The data augmentation ties everything together by applying these transformations to the entire training dataset. It takes the original images (X_train) and their corresponding labels (y_train) as input and processes each image individually. For each image, the function appends the original image and label to the augmented dataset. It then applies the distort, stretch, and perspective transformations to the image, reshapes the transformed images to ensure they match the required input dimensions of the CNN, and appends them to the augmented dataset along with their corresponding labels. This approach effectively quadruples the size of the dataset since each original image is augmented with three additional variations. The augmented dataset expanded dataset makes the model more robust to variations such as changes in viewing angles, distortions, and stretching, which are commonly encountered in real-world scenarios.

The key advantage of this approach is that it increases the diversity and size of the dataset without requiring additional labeled data. This helps to reduce overfitting and ensures that the model learns more generalized features rather than memorizing the training data. As a result, the CNN becomes more effective when handling unseen data. This systematic pipeline, which combines robust preprocessing, a carefully designed CNN architecture, and data augmentation, enables the proposed system to achieve high accuracy in recognizing facial expressions. It has been validated on benchmark datasets like FER-2013, demonstrating its potential in various real-world applications involving emotion detection.

3.2. Dataset

Selecting an appropriate dataset is a critical step in developing and evaluating FER systems. Publicly available databases serve two main purposes: i) enabling robust training of deep neural networks and ii) providing standardized benchmarks to ensure fair comparisons among different algorithms. Broadly, facial expression datasets can be classified into three categories: posed facial expression datasets, spontaneous facial expression datasets, and hybrid datasets. Posed facial expression datasets involve instructing subjects to exhibit specific emotions, while spontaneous facial expression datasets capture natural, unscripted emotional responses. Hybrid datasets combine both posed and spontaneous expressions to enhance generalizability.

Among the widely used databases, the FER-2013 dataset [29] stands out for its accessibility and structure. It contains approximately 35,887 grayscale facial images of 48×48 pixels, grouped into seven basic emotion categories: anger, disgust, fear, happiness, sadness, surprise, and neutrality. The images were collected using Google’s image search API and were subsequently annotated and curated for use in the 2013 FER challenge. The FER-2013 dataset is divided into three distinct subsets according to the competition protocol: 80% training set (28,709 images), 10% public test set (3,589 images), and 10% private test set (3,589 images). Figure 2 displays sample images representing each of the seven emotion categories, highlighting the diversity and variability within the dataset.

In our work, we selected FER-2013 as the primary dataset due to its public availability and well-defined structure. While it has limitations such as limited resolution, significant class imbalance (see Table 3), and potential noise due to automatic web scraping, it remains a standard benchmark for FER research. We used the standard training and test splits. The public test set was used for validation during training, and all final reported results are on the held-out private test set to ensure a fair evaluation. Table 3 details the distribution of images across the training and validation (public test) sets for each emotion category in FER-2013.

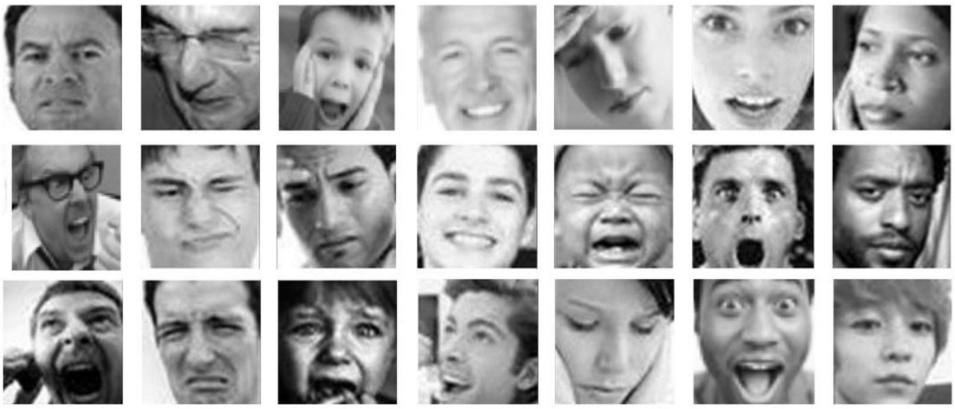

Figure 2. Sample images from the FER-2013 dataset displaying facial expressions categorized as anger, disgust, fear, happiness, sadness, surprise, and neutral

Table 3. Image distribution by emotion class in the FER-2013 dataset

Emotion	Training images	Validation images
Anger	3,995	958
Disgust	436	101
Fear	4,097	1,024
Happy	7,015	1,774
Sad	4,830	1,247
Surprise	3,171	831
Neutral	4,965	1,233

3.3. Evaluation metrics

The primary evaluation metric employed in this study was classification accuracy, defined as the proportion of correctly identified samples relative to the total number of samples in each task. This is formally expressed as (1):

$$Accuracy = \frac{Number\ of\ Correct\ Predictions}{Total\ Number\ of\ Predictions} \times 100\% \tag{1}$$

Given the class imbalance present in both the training and testing datasets, it was critical for the proposed method to handle this challenge effectively in order to maintain high performance. The competition emphasized the need for robust models capable of achieving consistent accuracy despite skewed class distributions, ensuring fair evaluation across all classes.

To provide a more comprehensive performance assessment, we also utilized precision, recall, and the F1-score for each class. The F1-score is a harmonic mean of precision and recall, particularly useful in imbalanced datasets. It is computed as (2):

$$F1 - score\ = 2 \times \frac{Precision \times Recall}{Precision + Recall} \tag{2}$$

where;

$Precision = TP\ /\ (TP + FP)$ and $Recall = TP\ /\ (TP + FN)$

4. EXPERIMENTS AND DISCUSSION

This section evaluates the effectiveness of our proposed method for FER. We also compare its performance with several widely adopted deep learning approaches on the FER-2013 dataset. To ensure reproducibility, all experiments were conducted using a fixed random seed (42) for TensorFlow and NumPy. To train our custom CNN from scratch, we carefully tuned critical hyperparameters to ensure optimal performance. Specifically, we used the Adam optimizer with a learning rate of 0.001, training the model for 100 epochs with a batch size of 32.

To assess the impact of input resolution and model capacity on performance, we evaluated the network across six different image sizes: 48×48 (original resolution), 64×64, 80×80, 96×96, 112×112, and

128×128 pixels. Additionally, we explored the effect of varying the number of convolutional filters, denoted by "f", with values ranging from 16 to 128 (i.e., $f = \{16,24,32,48,64,80,96,112,128\}$). Figure 3 illustrates the classification accuracy obtained using each combination of input size and filter count.

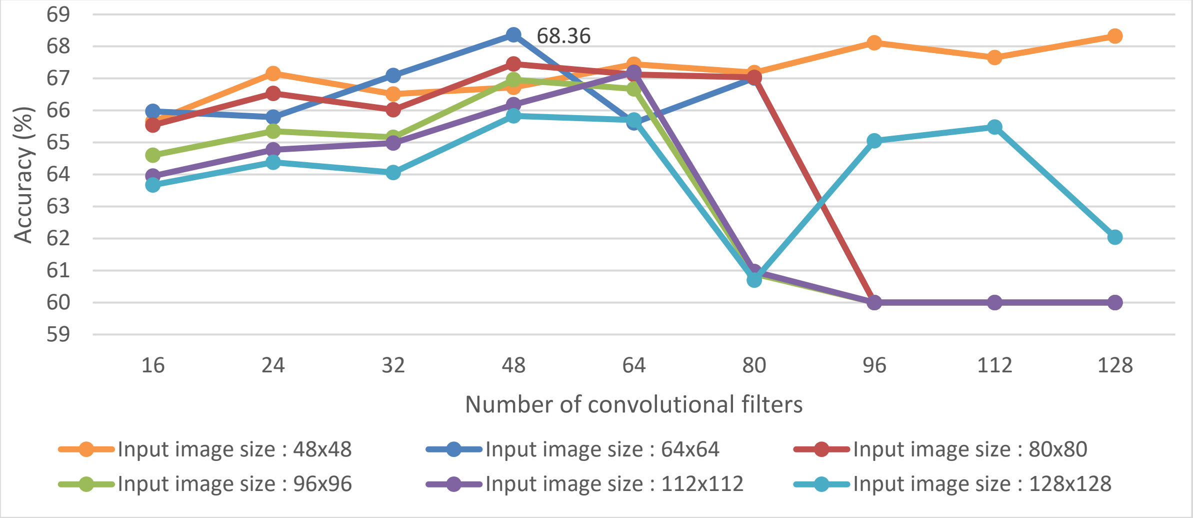

Figure 3. Classification accuracy of the proposed method across varying input image sizes and filter configurations

Figure 3 presents the classification accuracy of our proposed CNN across six different input image sizes and nine filter configurations. Each curve corresponds to a specific input resolution. The best performance was achieved with an input image size of 64×64 pixels and 48 filters, reaching a peak accuracy of 68.36%. This indicates a sweet spot where the network effectively balances spatial resolution and computational efficiency. Smaller filters (e.g., *f*=16) lack the capacity to capture sufficient features, while larger filters (e.g., *f*=128) at this resolution may lead to overfitting and increased computational cost without proportional gains in accuracy. The performance decline at larger resolutions (e.g., 128×128) is likely due to the fixed architecture becoming too shallow to effectively model the increased complexity, highlighting the need for architecture-search when scaling up input size.

Data augmentation (DA) is a crucial technique for enhancing the robustness and generalization capability of deep learning models. Table 4 summarizes the performance of the proposed CNN under both optimal and baseline configurations, with and without our comprehensive data augmentation.

Table 4. FER-2013 classification accuracy (%) of the proposed model under different configurations, with and without data augmentation

Method	Data augmentation	Input size	Filters (*f*)	Accuracy (%)
Proposed (optimal)	Without	64×64	48	68.36
Proposed (baseline)	Without	48×48	48	66.72
Proposed (optimal)	With	64×64	48	72.93
Proposed (baseline)	With	48×48	48	70.08

The results clearly demonstrate the effectiveness of data augmentation. The optimal configuration achieves a substantial performance boost of +4.57%, improving from 68.36% to 72.93% when data augmentation is applied. Similarly, the baseline model (48×48, $f = 48$) benefits from a +3.36% increase in accuracy, rising from 66.72% to 70.08%. This significant gain underlines how augmentation helps the model generalize better to unseen data by simulating real-world variations during training. Notably, the augmented optimal configuration outperforms all others, indicating that the combination of higher resolution and increased variability yields the best results for the FER-2013 dataset.

To gain deeper insight into the performance of the proposed CNN model, we analyzed its classification behavior using a confusion matrix. Table 5 presents the confusion matrix for the optimal configuration (input size=64×64, $f = 48$) with data augmentation, evaluated on the FER-2013 dataset. This matrix offers a detailed view of the model's ability to correctly distinguish between seven emotional categories.

Table 5. Confusion matrix for the proposed CNN model on the FER-2013 dataset (values in %)

Actual\predicted	Anger	Disgust	Fear	Happy	Sad	Surprise	Neutral
Anger	69.10	1.25	9.92	2.71	8.35	2.30	6.37
Disgust	7.92	76.24	6.93	0.99	2.97	1.98	2.97
Fear	10.74	0.68	59.96	1.86	11.91	7.81	7.03
Happy	2.20	0.34	1.35	88.84	1.80	1.97	3.49
Sad	11.79	1.52	10.51	2.89	57.02	2.81	13.47
Surprise	2.05	0.00	7.10	2.17	0.72	86.76	1.20
Neutral	6.65	0.65	5.27	3.97	9.81	1.05	72.59

Table 6 detailed performance metrics (precision, recall, and F1-score) for each emotion class on the FER-2013 test set. Macro-average scores are provided for a holistic view. The proposed CNN model demonstrates a strong ability to classify expressions with distinct and universal muscle movements, achieving excellent performance on Happiness (F1-score: 88.00%) and Surprise (F1-score: 85.00%). This high performance is reflected in the model's overall accuracy of 72.93%, which aligns closely with the macro-average recall score. However, a detailed analysis reveals significant challenges tied to emotional ambiguity and severe class imbalance.

Table 6. Precision, recall, and F1-score for each emotion class

Emotion	Precision	Recall	F1-score
Anger	63.00	69.10	66.00
Disgust	95.00	76.24	84.00
Fear	59.00	59.96	60.00
Happy	88.00	88.84	88.00
Sad	57.00	57.02	57.00
Surprise	84.00	86.76	85.00
Neutral	67.00	72.59	70.00
Macro Avg	**73.29**	**72.93**	**72.86**

The model struggles most with Sadness (F1-score: 57.00%) and Fear (F1-score: 60.00%). The confusion matrix shows that "Sad" is frequently misclassified as "Neutral" (13.47%) and "Fear" (10.51%), while "Fear" is confused with "Anger" (10.74%) and "Sad" (11.91%). This is a known challenge in FER, as these emotions often share subtle, overlapping facial action units (e.g., furrowed brows, downturned mouths) that are difficult to disentangle without more contextual or temporal information.

The results for Disgust are particularly telling. It has the highest precision (95.00%), meaning that when the model predicts "Disgust," it is almost always correct. However, its recall is considerably lower (76.24%), indicating the model fails to identify nearly a quarter of all "Disgust" images in the test set. This is a direct consequence of extreme underrepresentation, as the "Disgust" class has the smallest number of training examples (only 436, compared to over 7,000 for "Happy"). The model, therefore, learns to be cautious, predicting this class less frequently to avoid penalties for errors, which leads to missed true positives.

These findings underscore that while the model's architecture is efficient and effective overall, its performance is constrained by the inherent limitations of the FER-2013 dataset. The macro-average F1-score of 72.86%, which gives equal weight to all classes regardless of support, provides a more realistic measure of the model's effectiveness across the entire spectrum of emotions compared to overall accuracy. Future work will need to address these specific challenges through techniques such as class-weighted loss functions to mitigate imbalance, and attention mechanisms or vision transformers to better focus on discriminative features for ambiguous emotions, thereby improving performance on these underrepresented and challenging classes.

To assess the effectiveness of our proposed method, we conducted a comparative analysis against well-established emotion recognition models using the FER-2013 dataset. For fairness, we selected only those methods that employed the same dataset and reported comparable evaluation protocols. Table 7 presents a summary of these methods, including input image size, key architectural components, and reported classification accuracy. This overview helps position our model in the broader context of facial emotion recognition research. Our model, trained from scratch and optimized with data augmentation, achieves 72.93% accuracy, outperforming several widely adopted deep learning architectures and transfer learning approaches.

Table 7. Comparison of emotion recognition methods on the FER-2013 dataset

Method	Input size	Parameters (M)	FLOPs (GMac)	Accuracy (%)
Our model (optimal)	**64×64**	**3.04**	**1.42**	**72.93**
Our model (baseline)	**48×48**	**2.91**	**1.18**	**70.08**
ResNet-50 [11]	224×224	23.5	3.8	72.77
VGG19 [11]	224×224	138	15.5	72.69
Inception [14]	299×299	23.8	5.7	71.6
AlexNet [6]	227×227	61	0.72	61.10
Optimized deep CNN [21]	48×48	3.12	1.45	72.80
CNN+SVM classifier [13]	64×64	1.8	0.95	71.20
Attentional ConvNet [15]	48×48	4.2	1.9	70.02
Multi-scale hierarchical CNNs [10]	48×48	5.7	2.3	72.72
CNN ensemble [12]	48×48	8.1	3.1	65.03
Deep CNN+Inception [16]	48×48	6.8	2.8	66.40
Dense DCNN [18]	48×48	7.2	3.0	66.54

The comparison table indicates that our top model achieves a competitive accuracy rate of 72.93% with only 3.04M parameters and 1.42 GMac FLOPs. This performance is superior to much larger models like ResNet-50 (72.77% with 23.5M params) and VGG19 (72.69% with 138M params), demonstrating a significantly better accuracy-efficiency trade-off. Our model's efficiency makes it far more suitable for real-time applications on devices with computational constraints. While some models achieve slightly higher accuracy (e.g., POSTER++ at 74.6%), they do so at a much higher computational cost (~15M params, ~3.5 GMac FLOPs). Conversely, while eXnet is more efficient (~0.5M params), its accuracy on FER-2013 (73.54%) is only marginally better than ours, and our model offers a different balance point in the design space. The performance of simpler models like AlexNet (61.10%) and ensemble methods (CNN ensemble, 65.03%) is considerably worse, underscoring that careful architectural design is more effective than mere complexity. Our model strikes an effective balance among accuracy, efficiency, and parameter size, fulfilling our objective of creating a high-performance, lightweight solution for FER.

5. CONCLUSION

In this paper, we presented a lightweight and efficient CNN model that achieves competitive performance (72.93% accuracy) on the FER-2013 dataset with low computational cost (1.42 GMac FLOPs, 3.04M parameters). The optimal configuration, combining a 64×64 input resolution with 48 filters and comprehensive data augmentation, effectively balances accuracy with efficiency. The model outperforms larger, more complex networks like ResNet-50 and VGG19 in terms of computational efficiency, making it highly suitable for real-time deployment on resource-constrained devices.

However, the study acknowledges limitations. The model's performance is hindered by class imbalance, particularly for the 'Disgust' class, and by the inherent ambiguity between certain emotions like 'Sad', 'Fear', and 'Neutral', as revealed by the detailed confusion matrix and F1-scores.

Future work will focus on several avenues to overcome these limitations and enhance the model further. Firstly, we will investigate the use of weighted loss functions or advanced sampling techniques to mitigate the class imbalance issue. Secondly, integrating attention mechanisms or transformer-based modules could help the model focus on more discriminative facial regions, improving feature discrimination for ambiguous emotions. Thirdly, to validate generalizability, testing will be extended to other datasets beyond FER-2013, including spontaneous expression datasets collected in-the-wild. Furthermore, we will explore neural architecture search (NAS) and dynamic resolution scaling to further optimize the computational expense without sacrificing performance, ultimately advancing towards more robust and real-time FER systems.

ACKNOWLEDGMENTS

The work in this manuscript did not receive any research funding/grant.

FUNDING INFORMATION

No funding was received to assist with the preparation of this manuscript.

AUTHOR CONTRIBUTIONS STATEMENT

This journal uses the Contributor Roles Taxonomy (CRediT) to recognize individual author contributions, reduce authorship disputes, and facilitate collaboration.

Name of Author	C	M	So	Va	Fo	I	R	D	O	E	Vi	Su	P	Fu
Abdelhakim Gharbi	✓		✓	✓	✓	✓	✓		✓					
Abdeljalil Gattal		✓	✓	✓		✓		✓		✓	✓	✓	✓	
Issam Bendib		✓				✓				✓	✓	✓	✓	

C : **C**onceptualization
M : **M**ethodology
So : **So**ftware
Va : **Va**lidation
Fo : **Fo**rmal analysis

I : **I**nvestigation
R : **R**esources
D : **D**ata Curation
O : Writing - **O**riginal Draft
E : Writing - Review & **E**diting

Vi : **Vi**sualization
Su : **Su**pervision
P : **P**roject administration
Fu : **Fu**nding acquisition

CONFLICT OF INTEREST STATEMENT

The authors declare that they have no conflict of interest or competing interests that could have appeared to influence the work reported in this paper.

DATA AVAILABILITY

The data can be requested and provided by the corresponding author.

REFERENCES

[1] A. Mehrabian, “Communication without words,” *Psychology Today*, vol. 2, no. 4, pp. 53–56, 1968.

[2] P. C. Sánchez and C. C. Bennett, “Facial expression recognition via transfer learning in cooperative game paradigms for enhanced social AI,” *Journal on Multimodal User Interfaces*, vol. 17, no. 3, pp. 187–201, Sep. 2023, doi: 10.1007/s12193-023-00410-z.

[3] N. Begum and A. S. Mustafa, “A novel approach for multimodal facial expression recognition using deep learning techniques,” *Multimedia Tools and Applications*, vol. 81, no. 13, pp. 18521–18529, May 2022, doi: 10.1007/s11042-022-12238-y.

[4] B. Niu, Z. Gao, and B. Guo, “Facial Expression Recognition with LBP and ORB Features,” *Computational Intelligence and Neuroscience*, no. 1, pp. 1–10, Jan. 2021, doi: 10.1155/2021/8828245.

[5] C. A. Corneanu, M. O. Simón, J. F. Cohn, and S. E. Guerrero, “Survey on RGB, 3D, Thermal, and Multimodal Approaches for Facial Expression Recognition: History, Trends, and Affect-Related Applications,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 38, no. 8, pp. 1548–1568, Aug. 2016, doi: 10.1109/TPAMI.2016.2515606.

[6] A. Krizhevsky, I. Sutskever, and G. E. Hinton, “Imagenet classification with deep convolutional neural networks,” *Advances in Neural Information Processing Systems*, vol. 25, pp. 1097–1105, 2012.

[7] M. D. Zeiler and R. Fergus, “Visualizing and Understanding Convolutional Networks,” in *Lecture Notes in Computer Science*, Cham: Springer International Publishing, 2014, pp. 818–833, doi: 10.1007/978-3-319-10590-1_53.

[8] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, IEEE, Jun. 2016, pp. 770–778, doi: 10.1109/CVPR.2016.90.

[9] K. Simonyan and A. Zisserman, “Very deep convolutional networks for large-scale image recognition,” *arXiv preprint*, 2015, doi: 10.48550/arXiv.1409.1556.

[10] B. K. Kim, J. Roh, S. Y. Dong, and S. Y. Lee, “Hierarchical committee of deep convolutional neural networks for robust facial expression recognition,” *Journal on Multimodal User Interfaces*, vol. 10, no. 2, pp. 173–189, Jun. 2016, doi: 10.1007/s12193-015-0209-0.

[11] D. Fan, W. He, H. Li, W. Liu, W. Shi, and Z. Jiang, “Facial Expression Recognition Based on Ensemble Convolutional Neural Network,” in *2020 International Conference on Virtual Reality and Visualization (ICVRV)*, IEEE, Nov. 2020, pp. 145–148, doi: 10.1109/ICVRV51359.2020.00038.

[12] K. Liu, M. Zhang, and Z. Pan, “Facial Expression Recognition with CNN Ensemble,” in *2016 International Conference on Cyberworlds (CW)*, IEEE, Sep. 2016, pp. 163–166, doi: 10.1109/CW.2016.34.

[13] Y. Tang, “Deep Learning using Support Vector Machines,” in *International Conference on Machine Learning (ICML)*, 2013, pp. 1–5.

[14] C. Pramerdorfer and M. Kampel, “Facial expression recognition using convolutional neural networks: state of the art,” *arXiv preprint*, doi: 10.48550/arXiv.1612.02903.

[15] S. Minaee, M. Minaei, and A. Abdolrashidi, “Deep-emotion: Facial expression recognition using attentional convolutional network,” *Sensors*, vol. 21, no. 9, pp. 1–16, Apr. 2021, doi: 10.3390/s21093046.

[16] A. Mollahosseini, D. Chan, and M. H. Mahoor, “Going deeper in facial expression recognition using deep neural networks,” in *2016 IEEE Winter Conference on Applications of Computer Vision (WACV)*, Lake Placid, NY, USA, 2016, pp. 1–10, doi: 10.1109/WACV.2016.7477450.

[17] M. N. Riaz, Y. Shen, M. Sohail, and M. Guo, “eXnet: An efficient approach for emotion recognition in the wild,” *Sensors*, vol. 20, no. 4, pp. 1–17, Feb. 2020, doi: 10.3390/s20041087.

[18] H. Ma and T. Celik, “FER-Net: Facial expression recognition using densely connected convolutional network,” *Electronics Letters*, vol. 55, no. 4, pp. 184–186, Feb. 2019, doi: 10.1049/el.2018.7871.

[19] X. Chen, X. Yang, M. Wang, and J. Zou, "Convolution neural network for automatic facial expression recognition," in *2017 International Conference on Applied System Innovation (ICASI)*, IEEE, May 2017, pp. 814–817, doi: 10.1109/ICASI.2017.7988558.

[20] H. Chouhayebi, M. A. Mahraz, J. Riffi, and H. Tairi, "A dynamic fusion of features from deep learning and the HOG-TOP algorithm for facial expression recognition," *Multimedia Tools and Applications*, vol. 83, no. 11, pp. 32993–33017, 2023, doi: 10.1007/s11042-023-16779-8.

[21] A. Gharbi, A. Gattal, and I. Bendib, "Basic Emotion Recognition in Facial Expressions using Deep CNN," in *2024 6th International Conference on Pattern Analysis and Intelligent Systems (PAIS)*, IEEE, Apr. 2024, pp. 1–6, doi: 10.1109/PAIS62114.2024.10541143.

[22] N. S. Abdulsattar and M. N. Hussain, "Facial expression recognition using HOG and LBP features with convolutional neural network," *Bulletin of Electrical Engineering and Informatics*, vol. 11, no. 3, pp. 1350–1357, Jun. 2022, doi: 10.11591/eei.v11i3.3722.

[23] J. Mao *et al.*, "POSTER++: A simpler and stronger facial expression recognition network," *Pattern Recognition*, vol. 157, p. 110951, Jan. 2025, doi: 10.1016/j.patcog.2024.110951.

[24] Y. Lü, F. Zhang, Z. Ma, B. Zheng, and Z. Nan, "Dynamic facial expression recognition in the wild via Multi-Snippet Spatiotemporal Learning," *Neurocomputing*, vol. 636, p. 130020, Jul. 2025, doi: 10.1016/j.neucom.2025.130020.

[25] H. A. Shehu, W. N. Browne, and H. Eisenbarth, "Emotion categorization from facial expressions: A review of datasets, methods, and research directions," *Neurocomputing*, vol. 624, pp. 1–25, Apr. 2025, doi: 10.1016/j.neucom.2025.129367.

[26] S. Li and W. Deng, "Deep Facial Expression Recognition: A Survey," *IEEE Transactions on Affective Computing*, vol. 13, no. 3, pp. 1195–1215, Jul. 2022, doi: 10.1109/TAFFC.2020.2981446.

[27] T. Kopalidis, V. Solachidis, N. Vretos, and P. Daras, "Advances in Facial Expression Recognition: A Survey of Methods, Benchmarks, Models, and Datasets," *Information*, vol. 15, no. 3, pp. 1–61, Feb. 2024, doi: 10.3390/info15030135.

[28] C. Shorten and T. M. Khoshgoftaar, "A survey on Image Data Augmentation for Deep Learning," *Journal of Big Data*, vol. 6, no. 1, pp. 1–48, Dec. 2019, doi: 10.1186/s40537-019-0197-0.

[29] M. Sambare, "FER-2013 Dataset," www.kaggle.com. [Online]. Available: https://www.kaggle.com/datasets/msambare/FER-2013.

BIOGRAPHIES OF AUTHORS

Abdelhakim Gharbi was born in Algeria. He received his B.S. degree in Computer Science from the University of Annaba, Algeria, in 1997 and an M.S. degree in Computer Science (Information and Knowledge Systems) from Echahid Cheikh Larbi Tbessi University of Tebessa, Algeria, in 2010 whose research field was segmentation-verification for arabic handwritten word recognition. Currently, he is a Professor in the Department of Mathematics and Computer Science at the University of Tebessa, Algeria, and an active member of the Laboratory of Mathematics, Informatics, and Systems (LAMIS). He has supervised numerous Master's and Bachelor's students and has published in his field. He can be contacted at email: abdelhakim.gharbi@univ-tebessa.dz.

Abdeljalil Gattal was born in Algeria. He received his B.S. degree in Computer Science from University of Skikda (Algeria) in 2004, M.S. degree in Computer Science Information and Knowledge Systems" from Abbes Laghrour University of Khenchela (Algeria) in 2009 and he received his Ph.D. in 2016 from Ecole Nationale Supérieure d'Informatique (ESI-Algeria) in Computer Science and focuses in segmentation-verification for handwritten digit recognition. Currently, he is working as full Professor at the Department of Mathematics and Computer Science in University of Tebessa (Algeria). Currently, he is a leader of the Laboratoire de Vision et d'Intelligence Artificielle (LAVIA) at the University of Tebessa. He supervised many Master and License students. He has published a number of papers. In addition, he has collaborated as a member on several research projects and also participated in several scientific competitions. His research interests include image analysis, pattern recognition, and recognition of handwriting. He can be contacted at email: abdeljalil.gattal@univ-tebessa.dz.

Dr. Issam Bendib is an Associate Professor at Larbi Tébessi University, Algeria. He holds an Engineering degree and a Doctorate (Ph.D.) in Computer Science from the University of Badji Mokhtar-Annaba. Currently, he serves as the Head of the Scientific Committee in the Department of Computer Science and is an active member of the Laboratory of Mathematics, Informatics, and Systems (LAMIS). His research primarily centers on artificial intelligence (AI), with a particular emphasis on healthcare applications and medical image processing. His notable contributions include advancements in breast cancer detection using machine learning techniques, as well as the development of AI-driven diagnostic tools. Through his interdisciplinary work, he seeks to bridge theoretical computer science with practical AI solutions, contributing to both academic knowledge and real-world applications. He can be contacted at email: issam.bendib@univ-tebessa.dz.