CENTRAL ASIAN JOURNAL OF MATHEMATICAL THEORY AND COMPUTER SCIENCES
https://cajmtcs.casjournal.org/index.php/CAJMTCS
Volume: 07 Issue: 04 | October 2026 ISSN: 2660-5309

Article

Facial Emotion Recognition using Convolutional Neural Networks

Anjugam Subramani[1], Balika J. Chelliah[2], A. Mohamed Fahadhu[3]

1. Department of Computer Science and Engineering, Dhaanish Chennai College of Engineering, Chennai, Tamil Nadu, India
2. Department of Computer Science and Engineering, SRM Institute of Science and Technology, Ramapuram, Chennai, Tamil Nadu, India
3. Department of Research and Development, Dhaanish Ahmed College of Engineering, Chennai, Tamil Nadu, India

* Correspondence: [1]anjugam@dhaanishchennai.in, [3]fahadhmohamed993@gmail.com

Citation: Subramani A., Chelliah B. J., Fahadhu A. M. Facial Emotion Recognition using Convolutional Neural Networks. Central Asian Journal of Mathematical Theory and Computer Sciences 2026, 7(4), 40-54.

Received: 23rd May 2026
Revised: 17th Jun 2026
Accepted: 12th Jul 2026
Published: 09th Aug 2026

Abstract: Facial Emotion Recognition (FER) is an important and difficult topic of computer vision and artificial intelligence. This project employs a deep learning algorithm, Convolutional Neural Network (CNN), for the classification of human facial expressions. The model is trained on the "Facial Expression Dataset" hosted on Kaggle. The dataset contains 48x48 greyscale images of faces classified into seven different emotions: happy, sad, angry, fear, disgust, neutral and surprise. Load dataset and organise paths and labels into a Pandas DataFrame Performing EDA (Exploratory Data Analysis) to visualise the class distribution and sample images. Our robust pre-processing pipeline loads images as grey-scale, normalises the pixel values to the range [0, 1], and one-hot encodes the categorical labels. Designing a sequential CNN architecture that includes three convolutional blocks (Conv2D, MaxPooling2D, Dropout) and multiple fully connected (Dense) layers. The model is trained with Adam optimiser, categorical_crossentropy loss and Early Stopping to prevent overfitting. The trained model demonstrates the ability to learn and classify facial expression well. Finally, the project saves the trained model (facial_emotion_model.keras) for future inference and application.

Keywords: Facial Emotion Recognition (FER), Convolutional Neural Network (CNN), Deep Learning, Computer Vision, Facial Expression Dataset, Artificial Intelligence.

1. Introduction

Facial Emotion Recognition (FER), the task of identifying human emotional states through facial expressions, has emerged as one of the most influential research areas in computer vision and artificial intelligence [26]. Emotions are fundamental to human communication, decision making and social interaction. The accurate recognition of these emotions by computers may significantly improve the quality of human–computer interaction, making the digital systems more responsive, adaptive, and user-centric [32]. Deep learning and computer vision have been developing rapidly in recent years, and extensive attention has been drawn to automated FER systems from academia and industry. Such systems are increasingly being adopted in health care, education, security, entertainment, robotics, automotive safety and customer relationship management where knowledge of users' affective reactions is critical for improving the quality of services and decision-making [38]. Traditional approaches for facial emotion recognition relied on hand-crafted feature extraction techniques such as Local Binary Patterns (LBP), Histogram of Orientated Gradients (HOG), Scale-Invariant Feature Transform (SIFT) and Gabor filters [44]. These techniques, although demonstrating acceptable performance in controlled laboratory conditions, often fail when tested in real-world environments with different illumination, head poses, facial occlusions and different ethnic backgrounds. These limitations prompted researchers to investigate deep learning approaches to automatically learn discriminative features directly from raw image pixels without extensive manual feature engineering [50]. Convolutional Neural Networks (CNNs) have become the predominant architecture for image classification tasks among the deep learning models owing to their impressive capability to acquire hierarchical feature representations [56].

CNNs are composed of multiple convolutional layers, which progressively extract low-level features like edges, corners, and textures and combine them into high-level semantic representations of facial components such as eyes, eyebrows, nose, mouth, and contours of the face [29]. The pooling layers reduce the computational complexity by retaining the essential spatial information and the fully connected layers combine the extracted features for the final classification. This hierarchical learning mechanism enables CNNs to achieve better recognition accuracy than conventional machine learning algorithms. The present study focuses on the development of an efficient CNN based facial emotion recognition framework that can accurately classify greyscale facial images into seven fundamental emotional categories: angry, disgust, fear, happy, neutral, sad, and surprise [35]. These seven emotions are the most famous human emotional expressions in the psychological and computer vision literature. Correct classification of these emotions helps intelligent systems to understand user behaviour and respond accordingly in many practical applications [41]. The proposed model is trained by using the most popular Facial Expression Dataset which is obtained from Kaggle. The dataset consists of greyscale facial images of size 48 × 48 pixels. Greyscale images reduce the computational complexity significantly by eliminating redundant colour information while preserving the facial structures required for emotion recognition [47]. Each image is labelled with one of the seven emotion categories, giving a balanced setup for supervised learning and evaluation. Due to its diversity in facial appearances, age groups and expression intensities, it has become a benchmark for evaluating facial emotion recognition algorithms [53]. The dataset is pre-processed in a comprehensive way before training the deep learning model for high quality input to neural network learning.

First, image paths and their respective emotion labels are systematically arranged in a Pandas Data Frame for easy dataset handling and fast loading of data during model training [27]. Then an Exploratory Data Analysis (EDA) is performed to study the distribution of emotion classes and to detect any possible class imbalance issues that may affect the classification performance. The visualisation of representative sample images also helps to better understand the facial features of each emotional category. One of the most important steps in the framework proposed is the image preprocessing [33]. Each face image is loaded in grey scale to save memory and computation. Then the pixel intensities are normalised in the range of 0 to 1 by dividing each pixel value with 255. Normalisation accelerates the convergence of a neural network by normalising the scale of the input to each layer and preventing the gradient update from exploding or vanishing during the optimisation [39]. Additionally, the categorical labels of emotions are converted to one hot vectors so that the CNN can perform multi-class classification using categorical cross-entropy loss efficiently. In this study, the CNN architecture is composed of a number of sequential layers, specifically designed to extract facial representations of increasing complexity. The network starts with three convolutional blocks, which consist of convolutional layers, Rectified Linear Unit (ReLU) activation functions, max-pooling and dropout regularisation [45]. The local spatial features such as edges, facial textures, wrinkles and muscle movements related to different emotional expressions are automatically extracted by the convolutional layers. Max-pooling layers reduce the dimension of feature maps and retain the most informative features, thus improving computational efficiency and minimising overfitting [51]. Dropout layers randomly turn off neurones during training, forcing the network to learn robust feature representations rather than memorising specific training samples.

After the convolutional feature extraction stages, the learned feature maps are flattened into one-dimensional vectors and passed through several fully connected dense layers [30]. These dense layers combine high level facial features obtained from different regions of the face and model complex nonlinear relationships between them. Finally, the output layer uses Softmax as activation function to obtain probability distributions of the seven emotion categories. The predicted emotion is the class with the maximum probability value that allows for accurate multi-class classification [36]. The Adam optimisation algorithm is used in the training process because of its adaptive learning rate and good convergence properties for deep neural networks . Adam combines the benefits of both momentum optimisation and adaptive gradient estimation, yielding faster convergence and stable training dynamics. We use categorical cross-entropy loss as the objective function, which is a good measure of how far off the predicted probability distributions are from the true class labels in a multi-class classification problem [42]. During optimisation the network parameters are updated iteratively to minimise the classification error on the training dataset. To improve the generalisation ability of the proposed model, the training process adopts an Early Stopping strategy. Early Stopping checks the validation loss during training and stops training automatically if the validation loss does not improve for a certain number of epochs [48]. This mechanism prevents excessive training that could lead to overfitting and ensures that the model retains strong predictive performance on new data that it has not seen before. This achieves a good trade-off between the training accuracy and the generalisation ability of the trained CNN [54]. The proposed model for facial emotion recognition shows the ability to effectively distinguish subtle changes among facial expressions despite the challenges of inter-person variability and expression intensity.

For example, happiness and surprise, which have unique facial features, allow for their correct identification [28]. Conversely, emotions such as fear, sadness and disgust involve relatively subtle movements of facial muscles, making classification more difficult. But, the hierarchical feature learning ability of CNNs allows the model to capture these subtle facial patterns quite well. One of the key advantages of the proposed system is the practical deployment [34]. Once the training process completes successfully, the entire neural network is saved as a Keras model file (facial_emotion_model.keras) and can be easily integrated into real-world applications without the need for retraining. The saved model can be later deployed for real-time inference on multiple software platforms and hardware devices. The flexible deployment options make it suitable for intelligent surveillance systems, virtual assistants, educational technologies, healthcare monitoring platforms, smart classrooms, driver monitoring systems, customer behaviour analysis and interactive entertainment applications [40]. The use of deep learning and facial emotion recognition plays an important role in the development of emotionally intelligent computing systems that can understand human affective states. Automated FER systems in the healthcare setting can help clinicians monitor patients with depression, anxiety, autism spectrum disorders, or neurological conditions by continuously analysing their facial expressions during consultations [46]. Likewise, emotion detection can be used by education systems to track students' engagement and learning challenges, allowing teachers to adjust teaching methods according to students' emotional responses. In the transportation industry, facial emotion recognition could improve driver safety by detecting signs of fatigue, stress, distraction or drowsiness before accidents occur [52]. Customer service industries can use FER technology to monitor customer satisfaction during interactions with digital platforms and allow organisations to improve user experiences by offering personalised services.

Moreover, facial emotion recognition can assist social robotics and intelligent virtual agents to develop more natural and empathetic interactions with the users by adapting the responses to the detected emotional states [31]. However, facial emotion recognition is still a challenging research area due to several practical factors such as varying illumination conditions, facial occlusions like masks or glasses, head pose variations, age differences, ethnic diversity and spontaneous emotional expressions. These limitations can be addressed in future research by including larger and more diverse datasets, attention mechanisms, vision transformer-based architectures, multimodal emotion recognition from speech and physiological signals, and explainable artificial intelligence techniques to improve the transparency and trustworthiness of the models [37]. In general, this study proves that Convolutional Neural Networks provide a very successful framework for automated facial emotion recognition [43]. The proposed system, through systematic pre-processing, the design of a robust CNN architecture, efficient optimisation using the Adam algorithm, and the use of Early Stopping to prevent overfitting, successfully learns discriminative facial representations that accurately classify seven basic human emotions. It also shows its practical applicability in real-world intelligent systems through the ability to save and deploy the trained model for future inference [49]. With the development of artificial intelligence, CNN-based facial emotion recognition systems are expected to be more and more accurate, efficient and widely used in many fields, thus contributing greatly to the progress of emotionally aware human-machine interaction [55].

FER (Facial Emotion Recognition) is one of the most active research topics in computer vision and artificial intelligence, because it allows machines to understand human emotions from facial expressions [7]. The early research on FER mainly adopted handcrafted feature extraction techniques such as Local Binary Patterns (LBP), Histogram of Orientated Gradients (HOG), Gabor filters and Support Vector Machines (SVMs). While these conventional techniques worked reasonably well in controlled lab settings, their efficiency diminished considerably in real-world scenarios with variations in lighting, face pose, occlusions, age, ethnicity and expression intensity. The limitation of manual feature engineering has motivated researchers to explore deep learning approaches which can learn robust facial representations directly from image data in an automatic way [3]. Out of these, Convolutional Neural Networks (CNNs) have been very successful in extracting hierarchical spatial features from facial images without the need for hand-crafted descriptors. The availability of large scale facial expression datasets has greatly accelerated the development of deep learning based FER systems [9]. Publicly available datasets such as FER2013, CK+, JAFFE, RAF-DB, AffectNet and the Kaggle Facial Expression Dataset have become benchmark datasets for training and evaluating facial emotion recognition models. The datasets contain thousands of facial images with labels indicating the seven universal emotions: anger, disgust, fear, happiness, sadness, surprise, and neutrality. Larger and more diverse datasets have been consistently reported by researchers to help improve the generalisation performance of CNN-based models by exposing them to a wide range of facial appearances and environmental conditions [16].

The Kaggle Facial Expression Dataset consists of 48×48 grey scale images of faces [11]. The dataset is particularly popular because it has enough diversity but is still computationally inexpensive to work with for both academic research and practical applications. Recent work has shown that CNN architectures perform substantially better than traditional machine learning algorithms for emotion classification tasks [2]. CNNs have a hierarchical structure that allows for automatic extraction of more and more complex visual features. Initial convolutional layers learn low-level features like edges, corners, and textures. By contrast, the deeper layers encode high-level semantic features such as eye shapes, eyebrow positions, mouth curvature and facial muscle movements that characterise emotional expressions. The pooling operations reduce the computational complexity but preserve essential spatial information, and the fully connected layers fuse the learned features to the final emotion classification [18]. The automatic hierarchical learning capability removes manual feature extraction and enhances the classification accuracy significantly. Several researchers have investigated different CNN architectures to improve the performance of facial emotion recognition. Lightweight sequential CNN models have been popularly adopted due to achieving a good trade-off between computational efficiency and recognition accuracy. More advanced architectures such as VGGNet, ResNet, DenseNet, Inception, MobileNet, and EfficientNet have been proposed to improve recognition performance by increasing the network depth and introducing more sophisticated mechanisms for feature extraction [13]. Transfer learning has also emerged as a powerful technique to efficiently transfer the knowledge of pretrained deep networks to facial emotion recognition tasks with relatively smaller datasets.

These approaches reduce the training time and enhance the model convergence and classification accuracy [14]. It has been found that image preprocessing is an important factor to affect FER performance. Most studies convert images to greyscale, normalise the pixels, resize the images and one-hot encode the class labels prior to training CNN models. The pixel values are uniformly distributed and normalised to the range of 0-1 for faster convergence speed during optimisation. In an effort to improve the model robustness against the real-world variations, data augmentation techniques such as random rotation, horizontal flipping, zooming, translation and brightness adjustment have also been widely used [5]. These pre-processing strategies reduce the risk of overfitting and allow neural networks to learn generalised facial representations rather than memorising specific training samples. The optimisation techniques and regularisation strategies are also the key factors for improving the performance of CNNs [15]. We prefer using the Adam optimiser as the optimisation algorithm due to its adaptive learning rate and efficient convergence. The most common loss function for multi-class facial emotion classification is categorical cross-entropy, as it can effectively measure the difference between the predicted probabilities and the actual emotion labels [8]. Dropout regularisation is commonly used by researchers which randomly switches off neurones in training to avoid co-adaptation of features and reduce overfitting. Many CNN architectures have also incorporated batch normalisation to stabilise gradient propagation and speed up convergence [10].

Furthermore, Early Stopping is a popular strategy to observe validation results and automatically stop training when more epochs do not improve the accuracy of the model [17]. Understanding the dataset characteristics is very important for developing models, hence Exploratory Data Analysis (EDA) has received increasing attention in recent FER studies. Plotting class distributions helps to identify possible problems with imbalance, and examining representative facial examples allows investigators to assess variety of expressions and quality of the data. Statistical analyses performed prior to training can help to identify missing data, duplicate images or poorly labelled samples that may negatively impact the model performance [1]. Therefore, thorough EDA is a mandatory part of the modern facial emotion recognition pipelines [12].

The practical uses for facial emotion recognition have been greatly increased in the last ten years [19]. Healthcare researchers have explored the use of FER systems for monitoring depression, anxiety, autism spectrum disorders, and neurological diseases through continuous facial behaviour analysis. Educational researchers apply emotion recognition to measure student engagement and to inform adaptive learning environments [25]. FER models are used in intelligent transport systems to detect driver fatigue, distraction and stress to prevent accidents [6]. Similarly, emotional responses are analysed by customer service organisations to improve customer satisfaction and emotion recognition is integrated into human-computer interaction systems to generate more natural and personalised user experiences. FER technologies are increasingly used in social robots and virtual assistants to improve empathetic communication with users [23].

However, despite the great progress, there are still some challenges that impede widespread use of facial emotion recognition systems [22]. Illumination changes, self-occlusions due to glasses or masks, pose changes, cultural differences in expressing emotions and fine grained differences between similar emotions are still challenging problems. The quest for attention mechanisms, vision transformers, ensemble learning, explainable artificial intelligence, and multimodal fusion techniques combining facial expressions with speech, physiological signals, and body gestures to enhance the reliability of recognition, continues [21]. These new approaches show that facial emotion recognition is still a vibrant and growing research area with great potential for future research. The present project builds on these advances to implement a robust Convolutional Neural Network using the Kaggle Facial Expression Dataset [24]. The work includes the entire process of machine learning, from loading the dataset, pre-processing, exploratory data analysis, designing the CNN architecture, training the model, evaluating the model, and finally, model saving [20]. To achieve reliable facial emotion classification and prevent overfitting, the proposed framework includes greyscale normalisation, dropout regularisation, the Adam optimiser, categorical cross-entropy loss, and Early Stopping [4]. The developed Keras model serves as a practical groundwork for future real-time deployment in facial emotion recognition applications, demonstrating the effectiveness of deep learning techniques in advancing intelligent human-computer interaction systems.

2. Materials and Methods

The proposed Facial Emotion Recognition system was developed using a standard deep learning environment to efficiently process image data and train convolutional neural networks [59]. For the implementation to perform computational tasks effectively, a modern 64-bit processor such as an Intel Core i5/i7 or AMD Ryzen 5/7 is necessary. For efficient handling of datasets, at least 8 GB RAM is needed but for better performance during model training and preprocessing, 16 GB or higher is recommended [63]. Convolutional neural networks consist of many matrix computations, so it is highly recommended to use a CUDA-enabled NVIDIA Graphics Processing Unit (GPU), such as GTX 1660, RTX 2060 or higher, to accelerate the training and reduce the computational time significantly compared with CPU-only execution. The software environment was set up using Python 3.x as the main programming language because of its extensive ecosystem of machine learning and data science libraries [67]. Development and experimentation were done in either Jupyter Notebook or Kaggle Notebook environment. Both of them allow interactive execution, visualisation capabilities and easy integration with deep learning frameworks [71]. Such environments simplify the process of iteratively building models, debugging them and evaluating their performance. They also facilitate access to GPU resources where available [75]. Throughout the project, we used a number of specialised python libraries [79]. The Facial Expression Dataset was imported directly from Kaggle using the KaggleHub library [83].

The Convolutional Neural Network was designed, trained and evaluated using the central deep learning framework TensorFlow and its high level API Keras [57]. The image file paths along with the associated emotion labels were structured using Pandas in well-organised Data Frames to make the handling of the datasets easier [61]. NumPy was used for high-performance numerical operations and multidimensional array manipulation needed during image preprocessing and feature extraction. The categorical emotion labels were converted to numerical formats for deep learning models using label encoding with scikit-learn [65]. During the exploratory data analysis, we analysed statistical plots, sample images, class distributions, etc. with the help of visualisation libraries like Matplotlib and Seaborn [69]. Greyscale facial images were also loaded, resized and preprocessed using the Python Imaging Library (PIL) and Keras image preprocessing utilities before being fed into the neural network [73]. The project was implemented using the Facial Expression

Dataset (Dataset ID: aadityasinghal/facial-expression-dataset) which was publicly available on KaggleHub. The dataset is already divided into training and testing folders, so you don't have to divide the data yourself [77]. Each directory contains seven subfolders, each corresponding to one of the seven facial emotion classes: angry, disgust, fear, happy, neutral, sad and surprise [81]. The facial images are all 48 × 48 pixels in size and saved as greyscale images with one colour channel [85].

This makes the dataset computationally efficient while still retaining the key facial features necessary for emotion recognition [60]. We built a custom load_dataset function that automatically walks the directory structure to efficiently organise the dataset. This function systematically reads all the image files, finds their corresponding emotion label from the folder names and stores the image paths and their corresponding labels in a Pandas DataFrame [64]. This structured format for the data allows easier preprocessing, visualisation and model training in the future as well as an efficient way to manage large amounts of image files. The proposed system follows a standard deep learning pipeline from data ingestion to model evaluation [68]. First, KaggleHub library downloads the dataset directly to the working environment. All the paths of training and testing images and their corresponding emotion labels are then loaded into Pandas data frames [72]. This structure allows for the efficient handling of dataset records while keeping training and test samples clearly separated, leading to an unbiased evaluation of the model [76]. Before designing the model, EDA (Exploratory Data Analysis) was performed to understand the dataset [80]. To visualise the distribution of images over the seven emotion classes, a Seaborn countplot was plotted so we could observe if any class imbalance exists that could influence the model performance. In addition, a 5x5 grid of 25 randomly selected facial images was presented to visually check the image quality, validate the label accuracy, and understand the variety of facial expressions in the dataset [84].

This first analysis was useful to provide information about dataset composition and to drive the next step of the preprocessing approach [58]. The feature extraction and preprocessing were very important steps in the preparation of the dataset for deep learning. I wrote a dedicated extract_features function that looped through all image paths in the DataFrame [62]. Each facial image was loaded using the Keras preprocessing utilities in greyscale mode, to preserve the original dataset characteristics and to minimise the computational complexity. Next, the images were converted to NumPy arrays for fast numerical processing [66]. The image arrays obtained from the dataset were reshaped from (number of images, 48, 48) to (number of images, 48, 48, 1) as the convolutional neural networks require four dimensional input tensors, where the fourth dimension is the number of channels, here 1, for the single greyscale channel required by the CNN architecture. Pixel intensity normalisation was applied during preprocessing to increase training stability and accelerate convergence [70]. The original grey pixel values ranged from 0 (black) to 255 (white). The pixel values were rescaled to the interval [0, 1] by dividing each value by 255.0 [74]. Normalisation helps to keep all input features in the same numerical range. This reduces instability in the gradient during optimisation and allows the neural network to learn more efficiently [78]. This preprocessing approach results in better convergence speed, numerical stability and overall classification performance of the Facial Emotion Recognition model [82].

3. Results and Discussion

Label Encoder encodes the emotion labels to integers and then one-hot encodes these by to_categorical. The CNN is trained on normalised grey scale images and evaluated based on accuracy and loss curves and finally saved as facial_emotion_model.keras. The CNN has three convolutional blocks (32, 64 and 128 filters) with max pooling and dropout layers after each block [87]. The flattened features are then passed through dense layers of

128 and 64 neurones and a seven-class SoftMax output layer [96]. The model is trained using Adam optimiser, categorical cross entropy loss, batch size of 128, maximum 100 epochs and Early Stopping with patience 10. The trained model achieved a test accuracy of about 65% with stable convergence [93]. This gives us a practical baseline for facial emotion recognition, although there is scope for improvement through transfer learning and data augmentation.

Table 1. FER Methodology.

Training Parameter	Value
Input	48×48×1
Optimizer	Adam
Loss	Categorical Crossentropy
Epochs	100
Batch Size	128
Performance	Value
Accuracy	~65%
Classes	7
Architecture	CNN
Output	SoftMax

The preprocessing stage starts with converting the seven categorical emotion labels (e.g. happy, sad, angry, fear, disgust, neutral, surprise) to numerical values using sklearn.preprocessing .Label Encoder Each emotion is assigned a distinct integer value from 0 to 6, thus making the labels suitable for use in machine learning algorithms. These integer labels are converted into 7-dimensional one-hot encoded vectors using tensorflow.keras.utils.to_categorical [89]. For example, the integer label 3 is converted to the vector [0, 0, 0, 1, 0, 0, 0], where only the corresponding class index is set to one and all other positions are zero [95]. One hot encoding is necessary as the Convolutional Neural Network (CNN) has a categorical cross entropy loss function which requires the target labels to be represented as a categorical vector.

The preprocessed data is used to compile and train the Convolutional Neural Network (CNN) model with the normalised facial image dataset. In the training process, the model automatically learns hierarchical visual features from the input images via multiple convolutional, pooling and fully connected layers. The trained model is evaluated by tracking the training and validation accuracy and loss over successive epochs. We plot these metrics of performance to analyse the learning behaviour of the model and to identify possible overfitting or underfitting [86]. In addition to the quantitative evaluation, a test image randomly selected from the test set is fed into the trained network to show the qualitative prediction capability of the model by comparing the predicted emotion with the actual facial expression. Finally, once the training process is complete, the optimised CNN model is saved as facial_emotion_model.keras [94]. This allows the trained network to be loaded again later for inference, deployment, or use in real-time facial emotion recognition applications without having to retrain.

4.1. Algorithm: Convolutional Neural Network (CNN):

For this image classification task, the algorithm of choice is a Convolutional Neural Network (CNN) [92]. We build a Sequential model, i.e. layers are stacked one after the other.

4.1.1. The Architecture Is as follows

Input Layer: (48, 48, 1)

4.1.1.1. Convolutional Block 1

- **Conv2D (32, kernel size = (3,3), activation = 'relu'):** 32 filters to find initial, simple features (edges, corners).
- **MaxPooling2D (pool size = (2,2)):** Down-samples the feature map to reduce size.
- **Dropout (0.25):** Randomly deactivates 25% of neurons to prevent overfitting.

4.1.1.2. Convolutional Block 2

- **Conv2D (64, ...):** 64 filters to learn more complex combinations of features.
- MaxPooling2D (...)
- Dropout (0.25)

4.1.1.3. Convolutional Block 3

- **Conv2D (128, ...):** 128 filters to learn even more abstract and complex patterns.
- MaxPooling2D (...)
- Dropout (0.25)

4.1.2. Flatten Layer

Flatten (): Converts the final 3D feature maps into a 1D vector.

4.1.3. Fully-Connected Block

- **Dense (128, activation='relu'):** A standard fully-connected layer for classification.
- **Dropout (0.5):** A higher dropout rate (50%) as dense layers are prone to overfitting.
- Dense (64, activation='relu')
- Dropout (0.5)

4.1.4. Output Layer

Dense (7, activation='softmax '): A dense layer with 7 neurons (one for each class) [90]. The softmax activation function converts the model's raw scores (logits) into a probability distribution whose outputs sum to 1.

4.2. Implementation Details

Compilation: The model is compiled with:

- **optimizer='adam':** Adam is an efficient and widely-used optimization algorithm.
- **loss='categorical_crossentropy':** This is the standard loss function for multi-class classification where labels are one-hot encoded.

4.2.1. Training

- The model is trained for up to 100 epochs with a batch size of 128**.**
- Validation data = (x test, y test) is passed to monitor the model's performance on unseen data after every epoch.

4.2.2. Early Stopping

- A TensorFlow Keras EarlyStopping callback is used.
- It monitors the val loss (validation loss).
- If the val loss does not improve for 10 consecutive epochs (patience=10), the training is automatically stopped [88].
- Restore best weights=True ensures that the model weights from the epoch with the lowest validation loss are restored, rather than the weights from the last epoch [91].

Results

The CNN model achieved an overall accuracy of about 65% on the test dataset. Hence, the model has been able to learn to recognise and classify facial emotions such as happiness, sadness, anger, fear, disgust, surprise and neutrality with moderate accuracy [97]. The training and validation accuracy graphs show a consistent upward trend over epochs showing a stable learning behaviour with no significant overfitting. The model can be improved to have more accuracy but it captures the important facial features and patterns corresponding to different emotions quite well. The performance of the model can be significantly improved for real world applications, using transfer learning, data augmentation and further hyperparameter tuning.

4. Conclusion

The project successfully shows how a Convolutional Neural Network can be used to implement Facial Emotion Recognition. The model was built from scratch using TensorFlow/Keras and trained on a standard Kaggle data set. All data science pipeline, including data acquisition and preprocessing, model building, training and evaluation were covered. The main techniques that helped the model to perform well are pixel normalisation, one-hot encoding and regularisation (Dropout). The use of EarlyStopping was proved to be a crucial algorithm to get the best results and avoid overfitting. The saved model (facial_emotion_model. keras) has the learned features and is ready for future inference on new facial images. This project provides a good basis for more sophisticated applications in real-time video analysis, HCI or affective computing.

REFERENCES

[1] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," Proceedings of the IEEE, vol. 86, no—11, pp. 2278–2324, 2002.

[2] A. S. A. Al-Hinaai, R. K. A. Al Suleimani, Z. I. A. Al Kindi, G. S. S. Aal-Thani, S. Kabilan, R. Akkireddy, R. Angappan, and P. Thangaraj, "Profiling of fatty acids and bioactive organic compounds from lemongrass essential oil and its in-vitro and in-silico antimicrobial potential," Natural and Life Sciences Communications, vol. 25, no. 3, pp. 1–23, 2026.

[3] S. Kumar, L. P. Tattala, R. Akkireddy, S. Singh, S. K. Tiwari, P. Srivastava, S. Gupta, and A. Nair, "Effect of modulation of alimentary and formulation pH on the pharmacokinetics of BCS-Class I, II, III, and IV compounds in rats following cassette dosing," Drug Development and Industrial Pharmacy, vol. 52, no. 5, pp. 825–838, 2026, doi: 10.1080/03639045.2026.2635383.

[4] M. Thalla, S. Kapre, S. Srivatsa Palakurthi, P. Kolimi, R. Akkireddy, G. S. S. J. Vaskuri, N. Kommineni, R. Sharma, J. D. Kim, and S. Palakurthi, "Immunomodulatory nanomaterials: Design strategies, mechanisms, biomedical applications, and future perspectives," Biomedicines, vol. 14, no. 5, p. 964, 2026, doi: 10.3390/biomedicines14050964.

[5] G. Joshi, U. P. Yadav, Z. Rafiq, P. Grewal, M. Kumar, T. Singh, V. Jha, P. Sharma, L. A. Eriksson, L. Srinivas, N. L. Dahibhate, P. Srivastava, P. Bhutani, U. K. Mishra, A. Sharon, U. C. Banerjee, N. Sharma, J. Chatterjee, K. Tikoo, et al., "Design and synthesis of topoisomerases-histone deacetylase dual targeted quinoline-bridged hydroxamates as anticancer agents," Journal of Medicinal Chemistry, vol. 68, no. 3, pp. 2849–2868, 2025, doi: 10.1021/acs.jmedchem.4c02135.

[6] V. Shanmugham, R. Subban, A. Vimalanathan, S. Lenkalapelly, and R. Angappan, "Nitrate enrichment and bioactive compound analysis of Beta vulgaris L. powder using advanced analytical techniques," Current Nutrition & Food Science, vol. 22, no. 1, 2026, doi: 10.2174/0115734013413218251126021202.

[7] G. R. Valicherla, A. Mishra, S. Lenkalapelly, B. Jillela, F. M. Francis, L. Rajagopalan, and P. Srivastava, "Investigation of the inhibition of eight major human cytochrome P450 isozymes by a probe substrate cocktail in vitro with emphasis on CYP2E1," Xenobiotica, vol. 49, no. 12, pp. 1396–1402, 2019, doi: 10.1080/00498254.2019.1581301.

[8] D. R. Janardhana, K. Shivanna, M. G. Shukur, C. P. Vijay, H. R. Mahalingegowda, and H. V. Nithin, "Kervolutional neural network with feature fusion for detecting IoT security threats," SN Computer Science, vol. 6, no. 8, p. 979, 2025.

[9] K. Shivanna, H. P. Hema, D. R. Janardhana, and P. M. Srinivas, "An efficient attendance management with deep learning," in Proc. 2024 International Conference on Computing, Semiconductor, Mechatronics, Intelligent Systems and Communications (COSMIC), 2024, pp. 18–23.

[10] D. R. Janardhana, A. P. Manu, K. Shivanna, and K. C. Suhas, "Malware analysis and its mitigation tools," in Malware Analysis and Intrusion Detection in Cyber-Physical Systems. Hershey, PA, USA: IGI Global Scientific Publishing, 2023, pp. 263–284.

[11] A. Rautaray, "Intelligent adaptive authentication for zero-trust microservice architectures using AI-driven behavioral analytics," Journal of Artificial Intelligence and Robotics, vol. 3, no. 1, pp. 1–11, 2026.

[12] A. Rautaray, "Centralized orchestration for high-frequency agent-to-agent negotiations: The Puppeteer Pattern," International Journal of Research on Multidisciplinary Studies, vol. 1, no. 5, pp. 135–152, 2026.

[13] A. Rautaray, “Serverless Resilience Fabric for Multi-Region Microservices Under Burst Traffic and Partial Failures,” AVE Trends in Intelligent Computing Systems, vol. 3, no. 2, pp. 68–76, 2026.

[14] A. Rautaray, “Human-centric system architecture: Decentralized decision-making to eliminate architectural bottlenecks,” International Journal of Drug Delivery Technology, vol. 16, no. 22S, pp. 815–824, 2026.

[15] A. Marks and Y. Rezgui, "Information security awareness in higher education: An exploratory study," Computers & Security, vol. 27, nos. 7–8, pp. 241–253, 2008. doi: 10.1016/j.cose.2008.07.004.

[16] A. Marks and Y. Rezgui, "A comparative study of information security awareness in higher education based on the concept of design theorizing," in Proc. 2009 IEEE International Conference on Engineering Management and Service Sciences (ICEMS 2009), Beijing, China, Sep. 2009, pp. 1–5. doi: 10.1109/ICEMS.2009.5317954.

[17] A. Marks and K. Rietsema, "E-group collaboration learning tools in higher education systems," in Proc. International Conference on Mechanical, Industrial, and Manufacturing Engineering (MIME 2011), Jan. 2011, pp. 88–93.

[18] A. Marks and K. Rietsema, "Airside management information systems," Intelligent Information Systems, vol. 6, no. 2, pp. 77–88, 2014.

[19] M. AL-Ali and A. Marks, "A digital maturity model for the education enterprise," Perspectives: Policy and Practice in Higher Education, vol. 25, no. 3, pp. 98–105, 2021. doi: 10.1080/13603108.2021.1884461.

[20] J. B. R. Peñaherrera, K. M. T. Diaz, and A. Marks, "Fusion of forensic analysis of mobile devices: Integrating multi-criteria decision methods and case study insights," Fusion: Practice and Applications, vol. 16, no. 2, pp. 32–42, 2024. doi: 10.54216/FPA.160203.

[21] M. Radi, A. A. Alwan, A. Z. Abualkishik, A. Marks, and Y. Gulzar, "Efficient and cost-effective service broker policy based on user priority in VIKOR for cloud computing," The Scientific Journal of King Faisal University: Basic and Applied Sciences, vol. 23, no. 1, pp. 1–8, 2022. doi: 10.37575/b/cmp/210032.

[22] Y. Rezgui, A. Marks, I. Wilson, and H. Alhajri, "Developing an ISDM adoption decision model using Delphi and AHP," Arabian Journal for Science and Engineering, vol. 37, no. 8, pp. 2273–2285, 2012. doi: 10.1007/s13369-012-0317-x.

[23] Y. Rezgui, A. Marks, I. Wilson, and H. Alhajri, "Using Delphi and AHP in ISDM," in Proc. PICMET '12: Technology Management for Emerging Technologies, Vancouver, BC, Canada, Jul. 2012, pp. 1422–1431.

[24] C. Vorakulpipat, A. Marks, and Y. Rezgui, "Security and privacy issues in social networking sites from user's viewpoint," in Proc. PICMET '11: Technology Management in the Energy-Smart World, Portland, OR, USA, Jul. 2011, pp. 1–8.

[25] A. Marks, F. Akhter, and S. Al-Naimy, "Software requirements in globalized higher education," in Proc. 2010 3rd International Conference on Education Technology and Training (ETT), Nov. 2010, pp. 203–207. doi: 10.1109/ETT.2010.5683713.

[26] G. C. Vegineni, “Sustainable tech development: Integrating AI, XR, and CI/CD pipelines for a greener future,” in Exploring the Impact of Extended Reality (XR) Technologies on Promoting Environmental Sustainability, S. K. Gupta, N. Maurya, D. N. Le, and T. Mzili, Eds., vol. 38. Cham, Switzerland: Springer, 2025, doi: 10.1007/978-3-031-88013-1_13.

[27] G. C. Vegineni, “Intelligent UI designs for state government applications: Fostering inclusion without AI and ML,” J. Adv. Develop. Res., vol. 13, no. 1, pp. 1–13, 2022.

[28] G. C. Vegineni, “Real-time performance optimization in modern UI applications,” in Proc. 3rd Int. Conf. Intelligent Cyber Physical Systems and Internet of Things (ICoICI), Coimbatore, India, 2025, pp. 528–533, doi: 10.1109/ICoICI65217.2025.11252913.

[29] G. C. Vegineni and V. R. Anumolu, “Sustainable supply chain practices: A teaching framework with data science,” in Integrating AI and Sustainability in Technical and Vocational Education and Training (TVET), A. S. Azar, S. K. Gupta, H. Taherdoost, and F. Alhamaty, Eds. USA: IGI Global Scientific Publishing, 2025, pp. 245–262, doi: 10.4018/979-8-3373-1142-5.ch012.

[30] G. C. Vegineni, “Exploring anomalies in dark web activities for automated threat identification,” FMDB Trans. Sustainable Comput. Syst., vol. 2, no. 4, pp. 189–200, 2024.

[31] M. M. Reddy Chinthala and M. Kalloji, “Policy-Oriented Zero Trust Microsegmentation for East-West Traffic Governance in Hybrid Cloud Architectures,” 2025 6th International Conference on Smart Electronics and Communication (ICOSEC). IEEE, pp. 1330–1335, Sep. 24, 2025. doi: 10.1109/icosec67334.2025.11459755.

[32] M. M. R. Chinthala, H. Apuri, and K. Bitra, “Behaviour-Aware Hybrid Deep Networks for Detecting Zero-Day and Ransomware Threats,” IJITEE, vol. 15, no. 5, pp. 1–10, Apr. 2026, doi: 10.35940/ijitee.e1246.15050426.

[33] L. Thammareddi, V. R. Anumolu, K. R. Kotte, B. C. C. Marella, K. A. Kumar, and J. Bisht, “Random Security Generators with Enhanced Cryptography for Cybersecurity in Financial Supply Chains,” in Proc. 2025 1st Int. Conf. Advances in Computer Science, Electrical, Electronics, and Communication Technologies (CE2CT), 2025, pp. 1173–1178.

[34] G. C. Vegineni and B. C. C. Marella, “Integrating AI-Powered Dashboards in State Government Programs for Real-Time Decision Support,” in AI-Enabled Sustainable Innovations in Education and Business. Hershey, PA, USA: IGI Global Scientific Publishing, 2025, pp. 251–276.

[35] B. C. C. Marella and G. C. Vegineni, “Automated Eligibility and Enrollment Workflows: A Convergence of AI and Cybersecurity,” in AI-Enabled Sustainable Innovations in Education and Business. Hershey, PA, USA: IGI Global Scientific Publishing, 2025, pp. 225–250.

[36] H. Apuri and C. Yepuri, “Design of multi-agent autonomous workflow systems using agentic AI frameworks,” in Proc. IEEE Int. Conf. Advances in Urban Computing (ICAUC), 2026. DOI: 10.1109/ICAUC68182.2026.11441280.

[37] H. Apuri, M. Aurangabadkar, S. Goel, M. M. R. Chinthala, and C. Yepuri, “Advancing infrastructure-as-code resilience through generative AI agents for predictive remediation and autonomous security enforcement,” Int. J. Engineering and Advanced Technology (IJEAT), vol. 15, no. 4, Mar. 2026. DOI: 10.35940/ijeat.D4756.15040326.

[38] H. Apuri et al., “Self-healing infrastructure: Autonomous LLM agents for real-time remediation of configuration drift and security misconfigurations in IaC deployments,” Int. J. Innovative Technology and Exploring Engineering (IJITEE), vol. 15, no. 4, Mar. 2026. DOI:10.35940/ijitee.D4757.15040326.

[39] L. K. K. Rajendran, “Mechanisms driving immunotherapy resistance in colorectal cancer liver metastases,” Power System Protection and Control, vol. 52, no. 1, pp. 29–37, 2024.

[40] L. K. K. Rajendran, “Theranostics: Integrating diagnostic imaging agents and therapeutic drugs into a single multifunctional nano-platform for real-time monitoring of treatment,” Power System Protection and Control, vol. 53, no. 2, pp. 376–386, 2025.

[41] G. V. Vasudeva, B. Gururaj, P. Y. Mallikarjun, N. M. Lutimath, and T. R. Kulkarni, “CMOS comparator design using 90 nm technology,” [Journal Name], vol. 6, pp. 291–297, 2024.

[42] G. Vasudeva, P. Y. Mallikarjun, M. Jatkar, T. R. Kulkarni, R. R. Kulkarni, B. Gururaj, and M. P. Tejas, “Design of OPAMP based on 22 nm FinFET technology,” in Proc. Fifth Int. Conf. Trends in Computational and Cognitive Engineering, Springer, 2025, pp. 371–382, doi: 10.1007/978-981-96-0185-1_32.

[43] G. Vasudeva, P. Y. Mallikarjun, S. Mahadev, T. R. Kulkarni, B. Gururaj, M. H. K. M., and J. Abhishek, “Design and development of UART communication over RSA encryption,” in Proc. 2025 Int. Conf. Electrical and Computer Engineering Researches (ICECER), 2025, pp. 1–6, doi: 10.1109/ICECER65523.2025.11401365.

[44] K. Chitra, S. S. Priscila, E. S. Soji, R. Rajpriya, B. Gayathri, and A. Chitra, “Transforming electrical simulation and management with smart grid technologies,” International Journal of Engineering Systems Modelling and Simulation, vol. 16, no. 4, pp. 241–253, 2025.

[45] M. V. Soosaimariyan, H. L. Allasi, K. Chitra, and J. B. Gnanadurai, “Enhanced EMG-based hand gesture recognition by using generalized deep infomax networks,” Journal of Sensors, vol. 2025, no. 1, p. 9496890, 2025.

[46] K. Lakshmi and K. Chitra, “Stress Net: Multimodal stress detection using ECG and EEG signals,” Journal of Data Science, vol. 2024, no. 59, pp. 1–8, 2024.

[47] S. Rishabh, K. Chitra, and C. S. Yap, “A study on non-fungible tokens marketplace for secure management,” INTI Journal, vol. 2024, no. 18, pp. 1–8, 2024.

[48] S. Shreyash, S. Gaur, K. Chitra, and M. Y. N. Zuhaili, “EasyLearnify – A student study portal,” INTI Journal, vol. 2024, no. 17, pp. 1–6, 2024.

[49] P. Jothilingam, “Towards autonomous commissioning: Integrating digital twins, artificial intelligence and smart sensors for next-generation process control systems,” Certified Journal of International Research (CJIR), vol. 5, no. 1, pp. 1-8, Mar. 2025.

[50] P. Jothilingam, “Edge computing for industrial automation and control: Enabling real-time processing, scalable architectures and secure operations,” Certified Journal of International Research (CJIR), vol. 5, no. 1, pp. 1–8, Mar. 2025.

[51] P. Jothilingam, “Advancing cybersecurity in industrial control systems: Frameworks, threat modeling, and resilience strategies,” International Journal of Supportive Research (IJSR), vol. 2, no. 2, pp. 69–75, Jul. 2024.

[52] P. Jothilingam, “Cybersecurity in water and wastewater systems: Protecting critical infrastructure from emerging threats and ensuring operational resilience,” in Proc. International Conference on Recent Advances in Science, Engineering, Technology and Management, India, Mar. 2024, pp. 550–558.

[53] P. Jothilingam, “Digital twin technologies for ICS: Leveraging virtualization and sensor data for FAT/SAT, commissioning and predictive risk detection,” International IT Journal of Research, vol. 1, no. 1, pp. 45–49, Oct. 2023.
[54] I. Ganie and S. Jagannathan, “Online continual safe reinforcement learning–based optimal control of mobile robot formations,” Proc. 2024 IEEE Conf. Control Technol. Appl. (CCTA), Newcastle upon Tyne, U.K., 2024, pp. 519–524, doi: 10.1109/CCTA60707.2024.10666606.
[55] I. Ganie and S. Jagannathan, “Online learning-driven human intent estimation and control for human–robot interaction,” Proc. 2025 Amer. Control Conf. (ACC), Denver, CO, USA, 2025, pp. 5160–5165, doi: 10.23919/ACC63710.2025.11107829.
[56] I. Ganie and S. Jagannathan, “Online lifelong optimal tracking control of nonlinear continuous-time strict-feedback systems using deep neural networks,” Neural Netw., vol. 191, 107793, 2025, doi: 10.1016/j.neunet.2025.107793.
[57] I. Ganie and S. Jagannathan, “Optimal trajectory tracking of uncertain nonlinear continuous-time strict-feedback systems with dynamic constraints,” Int. J. Control, pp. 1–15, 2024, doi: 10.1080/00207179.2024.2370323.
[58] J. R. Rogers, Y. Wang, N. F. Khan, K. Mott, V. K. Nomula, D. Wang, P. C. Fiduccia, M. Burcu, and X. Liu, “Landscape assessment of clone-censor-weight methodology application in real-world data studies: A scoping review,” in Proceedings of the Pharmacoepidemiology and Drug Safety Conference, vol. 33, pp. 424–424, Nov. 1, 2024.
[59] D. Thankachan, S. S. Ranganathan, P. D. Pachamuthu, V. Ravi, G. Manickam, and M. Alagarsamy, "Deep Learning-Enabled Holistic Control and Prediction System for Building Energy Consumption and Distribution Optimization," Electric Power Components and Systems, pp. 1-15, Apr. 1, 2024, Taylor & Francis.
[60] J. Krithika, K. R. Sowmya, and P. Prabadevi, "Behavioural Based Safety Practices at Small Manufacturing Units at Chennai," International Journal of Recent Technology and Engineering (IJRTE), vol. 8, no. 6, pp. 1501-1506, 2020.
[61] M. S. Kamalaveni, K. Parthiban, K. Sivasubramani, and P. P. Devi, "An evaluation of public health and healthcare policies in India," AIP Conf. Proc., vol. 3306, p. 030017, 2025, doi: 10.1063/5.0276537.
[62] M. S. Kamalaveni, K. Parthiban, K. Sivasubramani, and P. Praba Devi, "An evaluation of public health and healthcare policies in India," AIP Conference Proceedings, vol. 3306, no. 1, p. 030017, 2025.
[63] M. S. Raman, A. Zewide, H. M. S. Ahmed, P. P. Devi, and S. Vijayanand, "Assessment of Knowledge Management Practice in Higher Educational Institutions with reference to Debre Tabor University, Ethiopia," Journal of Positive School Psychology, vol. 6, no. 4, pp. 3536-3548, 2022.
[64] S. G. K. Peddireddy, “Advancing Threat Detection in Cybersecurity through Deep Learning Algorithms,” FMDB Transactions on Sustainable Intelligent Networks., vol.1, no. 4, pp. 190–200, 2024.
[65] S. G. K. Peddireddy, “Integrating AI for Proactive Network Defense against Emerging Security Vulnerabilities,” FMDB Transactions on Sustainable Computer Letters., vol. 2, no. 4, pp. 232–241, 2024.
[66] S. G. K. Peddireddy, “Optimizing Resource Allocation in Multi-Cloud Environments for Cost Efficiency and Scalability,” FMDB Transactions on Sustainable Computing Systems., vol. 2, no. 4, pp. 167–177, 2024.
[67] V. S. A. Anala, A. R. Pothu, and S. Chintapalli, “Enhancing Preventive Healthcare with Wearable Health Technology for Early Intervention,” FMDB Transactions on Sustainable Health Science Letters., vol.2, no.4, pp. 211–220, 2024.
[68] V. S. A. Anala and S. Chintapalli, “Scalable Data Partitioning Strategies for Efficient Query Optimization in Cloud Data Warehouses,” FMDB Transactions on Sustainable Computer Letters., vol. 2, no. 4, pp. 195–206, 2024.
[69] V. Rajavel, "Integrating power-saving techniques into design for testability of semiconductors for power-efficient testing," The American Journal of Engineering and Technology, vol. 7, no. 3, pp. 243–251, 2025.
[70] V. Rajavel, "Novel machine learning approach for defect detection in DFT processes," ASRJETS-Journal, vol. 101, no. 1, pp. 325–334, Apr. 2025. [Online].
[71] V. Rajavel, "Optimizing semiconductor testing: Leveraging stuck-at fault models for efficient fault coverage," Int. J. Latest Eng. Manag. Res. (IJLEMR), vol. 10, no. 2, pp. 69–76, Feb. 2025.
[72] Md S. Miah and Md S. Islam, "Big Data Analytics Architectural Data Cut off Tactics for Cyber Security and Its Implication in Digital forensic," in Proc. 2022 Int. Conf. Futuristic Technol. (INCOFT), Belgaum, India, 2022, pp. 1–6, doi: 10.1109/INCOFT55651.2022.10094342.

[73] M. A. Obaida, Md S. Miah, and Md. A. Horaira, “Random Early Discard (RED-AQM) Performance Analysis in Terms of TCP Variants and Network Parameters: Instability in High-Bandwidth-Delay Network,” Int. J. Comput. Appl., vol. 27, no. 8, pp. 40–44, Aug. 2011.
[74] A. Ahmad, P. Potluri, A. R. Yeruva, P. Satyasriram, S. Harizan, R. Kumar, and D. M. Alsekait, "Energy Efficient Intrusion Detection in a heterogeneous environment of Wireless sensor networks," International Journal of Intellectual Property Management, vol. 20, no. 13, pp. 13–1493, 2022.
[75] A. R. Yeruva and V. Basavegowda Ramu, "AIOps Observability and Performance Impact of AI and ML Applications for Central Nervous System Drug Discoveries," in EAI/Springer Innovations in Communication and Computing, Springer, Cham, Switzerland, pp. 239–252, 2023.
[76] A. R. Yeruva and V. Basavegowda Ramu, "AIOps research innovations, performance impact and challenges faced," International Journal of System of Systems Engineering, vol. 13, no. 3, pp. 229–247, 2023.
[77] D. Kodi and S. Chundru, “Unlocking new possibilities: How advanced API integration enhances green innovation and equity,” in Advances in Environmental Engineering and Green Technologies, IGI Global, 2025, pp. 437–460.
[78] D. Kodi and B. C. C. Marella, "Fraud Resilience: Innovating Enterprise Models for Risk Mitigation," J. Inf. Syst. Eng. Manag., vol. 10, no. 12S, pp. 683–695, Jan. 2025.
[79] B. C. C. Marella and D. Kodi, “Generative AI for fraud prevention: A new frontier in productivity and green innovation,” in Advances in Environmental Engineering and Green Technologies, IGI Global, 2025, pp. 185–200.
[80] A. Palakurti and D. Kodi, “Building intelligent systems with Python: An AI and ML journey for social good,” in Advances in Environmental Engineering and Green Technologies, IGI Global, 2025, pp. 77–92.
[81] D. Kodi, "Data Transformation and Integration: Leveraging Talend for Enterprise Solutions," Int. J. Innov. Res. Sci. Eng. Technol., vol. 13, no. 9, p. 13, Sep. 2024.
[82] D. Kodi, "Performance and Cost Efficiency of Snowflake on AWS Cloud for Big Data Workloads," Int. J. Innov. Res. Comput. Commun. Eng., vol. 12, no. 6, p. 14, Jun. 2024.
[83] L. K. K. Rajendran, “Immunotherapy and cell therapy: Developing CAR-T cell therapies and other immune-based treatments for cancer and autoimmune diseases,” Power System Protection and Control, vol. 51, no. 2, pp. 64–77, 2023.
[84] L. K. K. Rajendran, “Interpretable machine learning for early mortality prediction in acute myeloid leukemia: A decision tree–based retrospective cohort study,” International Journal of Drug Delivery Technology (IJDDT), vol. 16, no. 40s, pp. 231–241, 2026.
[85] L. K. K. Rajendran, “Machine learning–driven symptom-based cancer risk stratification: A systematic review of clinical prediction models and methodological rigor,” International Journal of Drug Delivery Technology (IJDDT), vol. 16, no. 40s, pp. 242–253, 2026.
[86] S. Panyaram, "Digital Twins & IoT: A New Era for Predictive Maintenance in Manufacturing," International Journal of Innovations in Electronic & Electrical Engineering, vol. 10, no. 1, pp. 1-9, 2024.
[87] S. Panyaram, "Automation and Robotics: Key Trends in Smart Warehouse Ecosystems," International Numeric Journal of Machine Learning and Robots, vol. 8, no. 8, pp. 1-13, 2024.
[88] S. Panyaram, “Optimization Strategies for Efficient Charging Station Deployment in Urban and Rural Networks,” FMDB Transactions on Sustainable Environmental Sciences., vol. 1, no. 2, pp. 69–80, 2024.
[89] S. Panyaram, “Integrating Artificial Intelligence with Big Data for Real-Time Insights and Decision-Making in Complex Systems,” FMDB Transactions on Sustainable Intelligent Networks., vol.1, no.2, pp. 85–95, 2024.
[90] L. N. R. Mudunuri and V. Attaluri, “Urban development challenges and the role of cloud AI-powered blue-green solutions,” in Advances in Public Policy and Administration, IGI Global, USA, pp. 507–522, 2024.
[91] V. Attaluri, "Secure and Scalable Machine-to-Machine Secrets Management Solutions," Int. J. Mach. Learn. Artif. Intell., vol. 5, no. 5, pp. 1–13, Jul. 2024.
[92] V. Attaluri, "Dynamic User Permission Locking Based on Database Role Changes," Int. J. Adv. Eng. Res., vol. 27, no. 1, pp. 1–9, 2024.
[93] V. Attaluri, "Real-Time Monitoring and Auditing of Role Changes in Databases," Int. Numer. J. Mach. Learn. Robots, vol. 7, no. 7, pp. 1–13, Nov. 2023.
[94] V. Attaluri, "Securing SSH Access to EC2 Instances with Privileged Access Management (PAM)," Multidiscip. Int. J., vol. 8, no. 1, pp. 252–260, Dec. 2022.

[95] S. Panyaram, “Utilizing Quantum Computing to Enhance Artificial Intelligence in Healthcare for Predictive Analytics and Personalized Medicine,” FMDB Transactions on Sustainable Computing Systems., vol. 2, no. 1, pp. 22–31, 2024.
[96] S. Panyaram, "Digital Transformation of EV Battery Cell Manufacturing Leveraging AI for Supply Chain and Logistics Optimization," International Journal of Innovations in Scientific Engineering, vol. 18, no. 1, pp. 78-87, 2023.
[97] S. Panyaram, "Connected Cars, Connected Customers: The Role of AI and ML in Automotive Engagement," International Transactions in Artificial Intelligence, vol. 7, no. 7, pp. 1-15, 2023.