Scientific Research Publishing

Engineering, 2026, 18(4), 145-154
https://www.scirp.org/journal/eng
ISSN Online: 1947-394X
ISSN Print: 1947-3931

Real-Time Recognition of Three Facial Emotions ("Surprise, Neutral, Happy") Based on CNN with Augmented Data

Hugues Auguste N'Drin[1,2*], Hyacinthe Kouassi Konan[1], Etienne Téna Soro[1], Olivier Asseu[1,2]

[1]Laboratoire des Sciences, des Technologies de l'Information et de la Communication en Abrégé (LASTIC), Ecole Supérieure Africaine des Technologies de l'Information et de la Communication (ESATIC), Abidjan, Côte d'Ivoire
[2]Institut National Polytechnique Félix Houphouët-Boigny (INPHB), École Doctorale Polytechnique (EDP)-Sciences et Techniques de l'Ingénieur (STI), Yamoussoukro, Côte d'Ivoire
Email: *hugues.ndrin@esatic.edu.ci, hyacinthekonan2000@yahoo.com, etienne.soro@esatic.edu.ci, oasseu@yahoo.fr

How to cite this paper: N'Drin, H.A., Konan, H.K., Soro, E.T. and Asseu, O. (2026) Real-Time Recognition of Three Facial Emotions ("Surprise, Neutral, Happy") Based on CNN with Augmented Data. *Engineering*, **18**, 145-154.
https://doi.org/10.4236/eng.2026.184010

Received: February 25, 2026
Accepted: April 20, 2026
Published: April 23, 2026

Abstract

Emotion recognition from facial expressions has become essential for applications such as human-computer collaboration, robot communication, and interactive interfaces [1]. This work proposes a real-time recognition system (*i.e.*, the system produces a prediction with a latency low enough for smooth interaction, typically less than 100 ms per image or greater than 10 frames per second) capable of classifying three emotions: surprise, neutrality, and joy from facial images. The model is based on a convolutional neural network (CNN) optimized by data augmentation techniques applied to the FER2013 dataset [2] (Data augmentation was applied only to the training subset, not before distribution). The CNN has three convolutional layers, four fully connected layers, and uses ReLU and Softmax functions. The proposed approach achieves a validation accuracy of 89%, maintains high and balanced recognition rates for each class, and is capable of processing slightly distorted faces (*i.e.*, faces with small geometric or photometric variations, such as rotations, translations, scaling changes, or partial expressions) [3]. These results demonstrate the feasibility of fast, robust emotion recognition applicable to real-time interactive scenarios.

Keywords

Convolutional Neural Network, Data Augmentation, Validation Accuracy, Emotion Detection

1. Introduction

Facial expressions convey a person's emotions through facial muscle movements

and are a reliable indicator of mental state. Facial expression analysis has numerous applications, including lie detection, social robotics, and data-driven animation [4]. For an intelligent agent or robot to interact effectively with humans, accurate emotion recognition is crucial [5]. Research on facial recognition has shown considerable progress, but challenges remain, particularly in maintaining a balanced recognition rate (defined as consistent performance across different classes, typically measured by averaging recalls per class to prevent a dominant class from biasing results) between different emotions [6]. This study focuses on three key emotions (surprise, neutral, happy) to optimize real-time recognition, simplify the classification problem, and reduce errors related to less represented classes [7] [8].

The use of CNNs, combined with data augmentation techniques, forms the core of this approach [3]. The following sections describe the related work, methodology, data collection and preprocessing, augmentation, system implementation, results, and finally, conclusions and future prospects.

Previous work

Facial expression recognition has been the subject of research for several years. CNNs have become the dominant approach due to their ability to automatically extract discriminating features. FER2013 was used to classify seven emotions, achieving acceptable accuracy. However, some classes, such as "disgust" and "fear", exhibited very low recognition rates (45% and 41%, respectively) [8]. Recent models combining deep CNNs and residual blocks have improved overall accuracy (85.24%) on CK+ and JAFFE, but remain limited by the small number of datasets and the inability to handle distorted faces [4]. Data augmentation has proven crucial for improving model robustness, as demonstrated by recent work on FER2013 and RAF-DB [5]. This work shows that combining an efficient CNN with targeted data augmentation allows for robust performance while maintaining a balanced recognition rate for all classes [3]. For balanced performance, it is important to supplement overall accuracy with class-specific metrics. Specifically, for each of the three emotions, the following should be reported: 1) Accuracy (the proportion of correct predictions among those assigned to a given class), 2) Recall (the proportion of correctly identified examples among all real-life examples of that class), 3) The F1 score (the harmonic mean of accuracy and recall, offering a compromise between the two).

These metrics should be calculated from the confusion matrix, using the standard formulas:

- Accuracy = TP/(TP + FP)
- Recall = TP/(TP + FN)
- F1 = 2 × (accuracy × recall)/(accuracy + recall)

2. Methodology

The developed CNN model comprises (**Figure 1** and **Table 1**):

- 3 convolutional layers (32, 64, 128 filters, 3 × 3 kernels, ReLU activation),
- 4 fully connected layers (750, 850, 850, 750 nodes, ReLU activation),

- 0.5 dropout after each dense layer,
- 3-node output layer, Softmax function.

The input image is 48 × 48-pixel grayscale. Normalization and face preprocessing ensure homogeneous input. Optimization is performed via SGD, with a learning rate of 0.01 and a cross-entropy loss function. Callbacks such as EarlyStopping, ReduceLROnPlateau, and ModelCheckpoint were used.

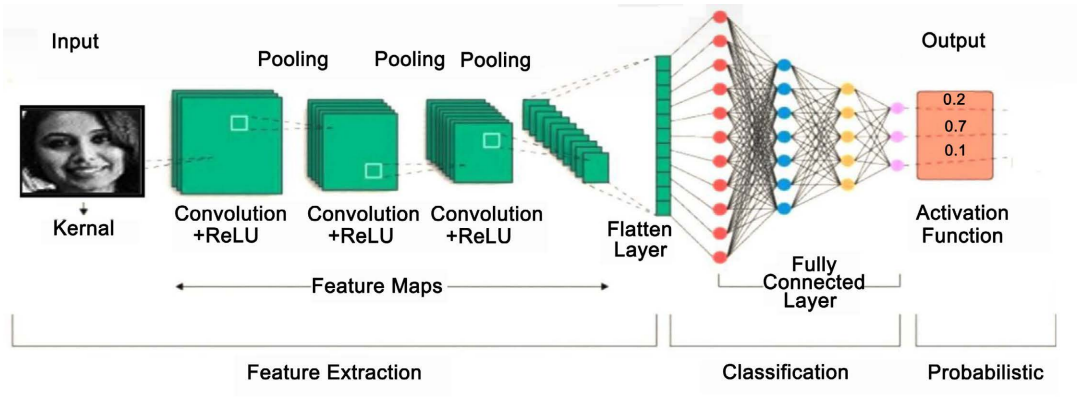


Figure 1. The convolutional neural network.

Table 1. System architecture.

Template contents	Details
First convolution layer	32 filters, 3 × 3 size, ReLU, input size 48 × 48
First layer of max pooling:	size 2 × 2
Second convolution layer:	64 filters, 3 × 3 size, ReLU
Second layer of max pooling:	size 2 × 2
Third convolution layer:	128 3 × 3 size filters, ReLU
Third layer of max pooling:	size 2 × 2
First fully connected layer:	750 knots, ReLU
Dropout layer:	random exclusion of 50% of neurons
Second fully connected layer:	850 knots, ReLU
Dropout layer:	random exclusion of 50% of neurons
Third fully connected layer:	850 knots, ReLU
Dropout layer:	random exclusion of 50% of neurons
Fourth layer fully connected:	750 knots, ReLU
Dropout layer:	random exclusion of 50% of neurons
Output layer:	3 nodes for 3 classes, SoftMax
Optimization function:	stochastic gradient descent (SGD)
Learning rate:	0.01
Callback functions:	EarlyStopping, ReduceLROnPlateau, ModelCheckpoint, TensorBoard

3. Data Collection and Preprocessing

To ensure the robustness and generalizability of the model, we used the FER2013 dataset, widely recognized in the facial recognition community. This dataset was selected for its wide diversity of expressions, angles, and lighting conditions.

Unlike classic multi-dataset approaches, we chose to focus the model on three emotions: surprise (training data: 1171 images; test data: 831 images), neutrality (training data: 4965 images; test data: 1233 images) and joy (training data: 7215 images; test data: 1774 images), in order to simplify the classification task and improve real-time accuracy. Examples of images used for training are shown in Figure 2.

Figure 2. Examples from the dataset.

Preprocessing proceeds as follows:

1) Face detection and cropping

Facial registration involves locating the face in each image to eliminate background noise. Detection was performed using the OpenCV cascade classifier. Once detected, the face is cropped to reduce spatial complexity and facilitate CNN model training.

2) Grayscale Conversion

All images were resized to 48 × 48 pixels and then converted to grayscale (one channel) to reduce computational complexity and accelerate network training. This conversion simplifies data representation while preserving essential information about facial expressions.

3) Image Normalization

Pixel values were normalized to range from 0 to 1. Normalization improves learning convergence and stabilizes model training by harmonizing image intensity values.

4) Data Augmentation

To enhance the CNN's generalization capabilities and address the limited number of examples per class, we applied data augmentation using the Keras Im-

ageDataGenerator API. This technique generates new images from existing ones by applying random transformations, including: 1) rotations, 2) horizontal and vertical translations, 3) shearing, 4) random zooms, and 5) horizontal flips.

This augmentation allows for the creation of a richer dataset, reducing overfitting and improving the model's robustness to variations in pose or lighting conditions, which is crucial for real-time recognition (Figure 3).

Figure 3. Data preprocessing.

4. Experimenting with Data Augmentation

To improve the robustness and generalizability of the CNN model, we applied data augmentation techniques using the Keras ImageDataGenerator API. This function generates new images from the existing dataset by applying random transformations, such as: 1) Rotation around the image center (±15˚); 2) Shearing to simulate pose variations; 3) Random zoom to represent faces that are closer or farther away; 4) Horizontal flipping; 5) Horizontal and vertical shifting.

Before augmentation, the FER2013 dataset used for the three targeted emotions comprised 12,040 images, or approximately 4013 images per class (surprise, neutral, happy). Since CNNs are highly data-dependent models, the dataset was enriched using the aforementioned transformations. After the increase, the dataset grew to 36,120 images, or approximately 12,040 images per class, significantly expanding the variety and coverage of use cases.

This increased data is particularly important for facial expression recognition because it:

1) Allows the model to better learn the natural variations of faces (pose, lighting, orientation).

2) Reduces the risk of overfitting on the original images.

3) Improves real-time accuracy and stability, essential for interactive applications.

For training, 80% of the images were used for training and 20% for validation. To test the model's robustness under more challenging conditions, a second scenario was tested: 65% of the images for training and 35% for testing (Figure 4). This allows us to assess the model's ability to generalize to a larger dataset not seen during training.

This approach has made it possible to verify that the CNN model optimized for three emotions maintains stable performance, even when the proportion of test data increases, while maintaining the prediction speed necessary for real-time processing.

Figure 4. Data augmentation.

5. System Implementation

The system was implemented in Python, using the Spyder IDE. The main libraries used are TensorFlow, Keras, NumPy, OpenCV, PIL, and Matplotlib. TensorFlow handles the neural network execution and manages CPU/GPU-optimized computational operations. Keras provides built-in functions for creating CNN layers, activation functions, optimizers, and training management. OpenCV is used for image preprocessing, including face detection via the cascade classifier, cropping, grayscale conversion, and normalization. ImageDataGenerator (Keras) manages data augmentation to enrich the dataset and improve generalization. Matplotlib is used to visualize the results, including confusion matrices and performance curves.

It is important to explicitly distinguish the preprocessing applied to the images from the FER2013 dataset from that used in the deployed web interface. In the case of FER2013, the images are already faces that have been detected, aligned, converted to grayscale, and resized to 48 × 48 pixels. Therefore, preprocessing during training is generally limited to normalizing the intensities (e.g., scaling pixels between 0 and 1); possibly standardization; and applying data augmentation techniques (rotations, translations, zooms, etc.). In contrast, in the web interface, the input images come from real streams (camera or uploaded images) and require a more complete pipeline including: 1) face detection via OpenCV (for example with a Haar Cascade type classifier), 2) extraction of the region of interest (ROI), 3) conversion to greyscale, 4) resizing to 48 × 48 pixels, 5) then normalization identical to that used in training.

To make the system accessible to end users, a web-based graphical user interface (GUI) was developed using HTML, CSS, and JavaScript, and connected to the model via a Flask server (Python). The user can select a local image and then click "Predict" to display the detected class. When an image is provided: 1) It is resized to 48 × 48 pixels to match the CNN input. 2) The OpenCV cascade classifier detects the facial region. The face is cropped to isolate the region of interest and remove the background. 3) The image is converted to grayscale, as the model was trained on a single channel. 4) Normalization is applied to harmonize the pixel values between 0 and 1. 5) The preprocessed image is then sent to the custom CNN, which predicts the emotion class from among surprise, neutral, and happy. This implementation ensures real-time compatibility, stable prediction for novel images, and the ability to be integrated into interactive systems or web applications. Figure 5 shows several screenshots of the interface, demonstrating the model's ability to accurately detect and classify images from diverse sources.

Figure 5. Real-time validation.

6. Results and Discussion

The proposed model, a convolutional neural network (CNN) with data augmentation, was evaluated on the FER2013 dataset, limited to the three targeted emotions: surprise, neutral, and happy. Training was performed with a split ratio of 80% for training and 20% for validation.

To ensure the reproducibility of the results, it is necessary to specify the main hyperparameters and training conditions. In particular, the model is trained with a batch size of 15351 images, a stochastic gradient descent (SGD) optimizer with a momentum parameter set to m (m = 0.9), and an initial learning rate of η (η = 0.01), adjusted according to a planning strategy (e.g., a reduction by a factor of γ (γ = 0.5) after p epochs without improvement in the validation loss). An early stopping mechanism is used with a patience of k (k = 10) epochs (*i.e.*, training is stopped if the validation metric does not improve for k consecutive epochs, according to a rule based on minimizing loss or maximizing accuracy). The experiments are initialized with a fixed random seed (s = 42) to ensure the reproducibility of the data partitions and weight initialization. Finally, the hardware used for training is specified (e.g., CPU or GPU, with the exact model), and the total training time or the number of epochs performed is indicated.

After 50 epochs, the model achieved a validation accuracy of 89%, demonstrating the effectiveness of the CNN architecture combined with increased data. Accuracy remained high and nearly constant for all three classes, even when images exhibited slight geometric distortions (pose or facial tilt). The evolution of accuracy over epochs was tracked using TensorBoard (**Figure 6**), where the x-axis represents the number of epochs and the y-axis the recognition rate.

The impact of data augmentation was clearly observed. The augmented model reached 89% accuracy in just 50 epochs. The applied transformations (rotation, translation, shear, zoom, and flip) introduced relevant variations into the dataset, improving the model's ability to generalize to new images. Dropout (0.5) and early stopping mechanisms were used to limit overfitting and ensure stable convergence.

To test the model's robustness, a second scenario was evaluated with a 65/35 split ratio (65% training, 35% testing). In this case, the model achieved 87% accuracy, slightly lower than the 80/20 split but still very satisfactory considering the larger volume of test data. This demonstrates that the model maintains stable performance even when the number of unseen examples increases, which is crucial for real-time applications.

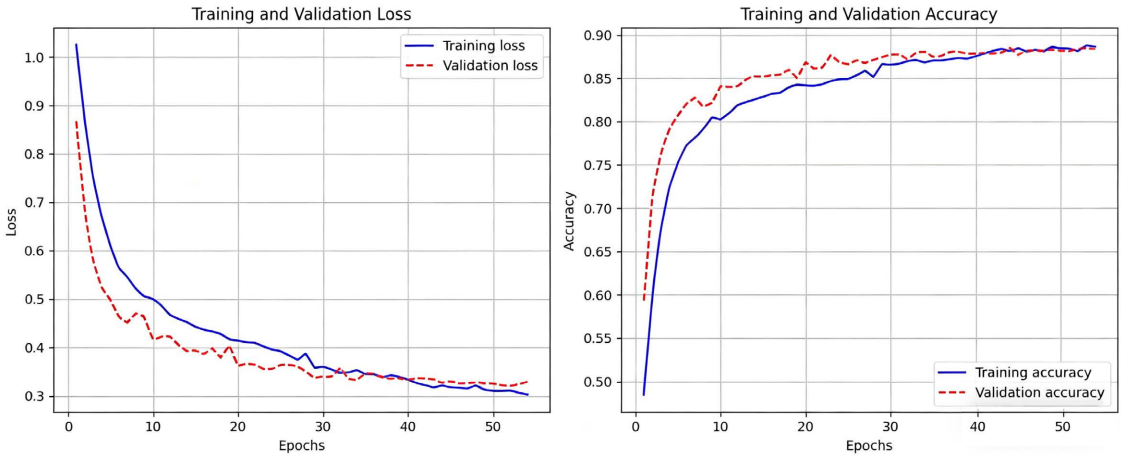
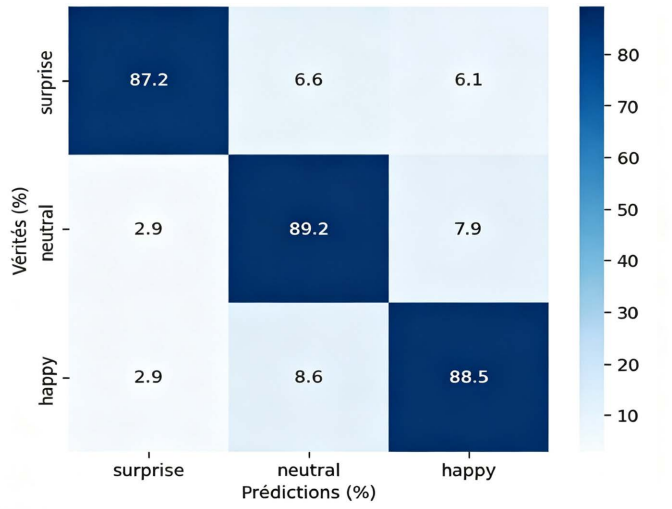


Figure 6. Training and validation curve.

Finally, the analysis of the results highlights that the model is able to classify the three emotions with a high and balanced recognition rate; process slightly distorted or rotated faces and work effectively on novel images thanks to the combination of data augmentation, dropout and normalization.

These results confirm that simplifying the problem (3 classes) and optimizing the CNN for real-time use makes it possible to achieve high accuracy, while maintaining low latency suitable for interactive or embedded systems (See **Figure 7** below).



	precision	recall	f1-score	support
surprise	0.89	0.88	0.88	831
neutral	0.84	0.88	0.86	1233
happy	0.91	0.89	0.90	1774
accuracy			0.88	3838
macro avg	0.88	0.88	0.88	3838
weighted avg	0.88	0.88	0.88	3838

Figure 7. Confusion matrix.

7. Conclusion and Future Outlook

In this research, we developed a real-time facial expression recognition system based on a data-augmented, optimized convolutional neural network (CNN). Focusing on three key emotions—surprise, neutral, and happy—the model achieves 89% validation accuracy on the FER2013 dataset. The results show that data augmentation, combined with regularization techniques such as dropout and early stopping, maintains a high and balanced recognition rate for each class. Furthermore, the model is capable of classifying images with slight geometric distortions, making it suitable for interactive, real-time applications.

Experiments conducted with different data splitting ratios (80/20 and 65/35) demonstrate that the model maintains robust performance even as the test data volume increases, confirming the stability and generalizability of the proposed CNN. These results are consistent with recent observations in the facial recognition literature and the importance of increasing data for limited datasets.

Several research avenues can be explored for the future: 1) Extension to more emotion classes: While focusing on three emotions has improved accuracy and stability, including other emotions such as sadness or fear could enrich the system's applicability, particularly in contexts of emotional monitoring or social interaction. 2) Optimization for real-time video streams: The current implementation has been validated on static images. Adapting it to video streams with continuous face detection and tracking would allow for the evaluation of performance over time sequences and the handling of emotional transitions. 3) Integration of advanced deep network techniques: The use of newer models, such as Residual CNNs (ResNet) or Transformers for vision, could improve the system's ability to capture subtle facial features while maintaining real-time performance. 4) Deployment on embedded and mobile platforms: Simplifying the model and optimizing the real-time CNN pave the way for integration on embedded devices, such as social robots or mobile applications, where latency and power consumption are critical.

In conclusion, this work demonstrates that combining a simple yet effective CNN, targeted data augmentation, and appropriate preprocessing enables the design of a robust and fast facial recognition system for interactive applications. This approach provides a solid foundation for future extensions to more comprehensive systems and dynamic environments.

Declaration

The platform used to create or produce the portraits and images in this article is provided via the links: https://mediacy.com/blog/ai-essentials-cnns-microscopy/, https://fr.freepik.com/photos/personne-triste, https://www.google.com/search?sca_esv=2a704d97ad930897&sxsrf=ANbL-n4Eq6k_Jp2mFwIoTm9EYUhYdWAoAg:1776756796080&udm=2&fbs=ADc_l-bpk8W4E-qsVlOvbGJcDwpnHC5OJXXTJvmMu2n9YYx-G8xzgQk24aW1N_FyIND5zVDd4bb14119C8nZHL5l4Fe3Q78DM888EmtVm1l7Ggrb1XBl29I-upxH2ZKiusq_Iw2q9oUHOoAZBYuy8EaAc-

NGNbMYqqXay6V_L7kQfc6l4SAS5l_Dqujgh0OOfmu5n67ZfjXhn9IJG0UpXFHUvFBgsdhB5UQ&q=les+diff%C3%A9rents+%C3%A9motions+sur+les+visages&sa=X&ved=2ahUKEwi25Lrttv6TAxWyWUEAHXXaARoQtKgLegQIFhAB&biw=1920&bih=919&dpr=1.

Conflicts of Interest

The authors declare no conflicts of interest regarding the publication of this paper.

References

[1] Li, S. and Deng, W. (2022) Deep Facial Expression Recognition: A Survey. *IEEE Transactions on Affective Computing*, **13**, 1195-1215. https://doi.org/10.1109/taffc.2020.2981446

[2] Goodfellow, I.J., Erhan, D., Luc Carrier, P., Courville, A., Mirza, M., Hamner, B., *et al.* (2015) Challenges in Representation Learning: A Report on Three Machine Learning Contests. *Neural Networks*, **64**, 59-63. https://doi.org/10.1016/j.neunet.2014.09.005

[3] Echoukairi, H., El Ghmary, M., Ziani, S. and Ouacha, A. (2023) Improved Methods for Automatic Facial Expression Recognition. *International Journal of Interactive Mobile Technologies* (*iJIM*), **17**, 33-44. https://doi.org/10.3991/ijim.v17i06.37031

[4] Guo, A. (2025) Enhancing Facial Expression Recognition with Robust CNN Architectures and Adaptive Preprocessing Techniques. *Applied and Computational Engineering*, **100**, 137-145. https://doi.org/10.54254/2755-2721/2025.20426

[5] Ajitha, V. (2024) CNN-Driven Enhancement in Facial Emotion Recognition Systems. *International Journal of Intelligent Systems and Applications in Engineering*, **12**, 2343-2350. https://ijisae.org/index.php/IJISAE/article/view/6620

[6] Liu, Y. (2023) The Study of Performance Related to Classical Convolutional Neural Networks in the Field of Facial Emotion Recognition. *Applied and Computational Engineering*, **8**, 470-474. https://doi.org/10.54254/2755-2721/8/20230248

[7] Xie, Y., Tian, W. and Yu, Z. (2023) Robust Facial Expression Recognition with Transformer Block Enhancement Module. *Engineering Applications of Artificial Intelligence*, **126**, Article ID: 106795. https://doi.org/10.1016/j.engappai.2023.106795

[8] Goodfellow, I., *et al.* (2013) Challenges in Representation Learning: Facial Expression Recognition Dataset (FER2013). *Neural Networks*, **27**, 45-55.