Indonesian Journal of Electrical Engineering and Computer Science
Vol. 33, No. 1, January 2024, pp. 442∼449
ISSN: 2502-4752, DOI: 10.11591/ijeecs.v33.i1.pp442-449

Convolutional neural network hyperparameters for face emotion recognition using genetic algorithm

Muhammad Sam'an, Safuan, Muhamma Munsarif
Departement of Informatics, Universitas Muhammadiyah Semarang, Semarang, Indonesia

Article Info

Article history:

Received Sep 21, 2023
Revised Oct 31, 2023
Accepted Nov 13, 2023

Keywords:

Convolution neural networks
Deep learning
Face emotion recognition
Genetic algorithms
Hyperparameter optimization

ABSTRACT

The development of artificial intelligence in facial emotion recognition (FER) is rapidly growing and has been widely applied in various fields. Deep learning (DL) techniques with evolutionary algorithms have become the preferred choice for solving various security, health, gaming, and other related problems. This research proposes the use of a genetic algorithm (GA) as the main method to optimize hyperparameters in the convolutional neural network (CNN) model for FER. The required computation time is approximately 37 hours 57 minutes 55 seconds, with generation 3 taking the longest time at around 16 hours 45 minutes 4 seconds. However, generation 3 achieved an accuracy of 76.11%, which is the highest compared to other generations. The results indicate that the more generations are involved, the higher the achievable accuracy. Furthermore, the proposed CNN-GA model in this study outperforms previous models that have been examined. Thus, this study makes a significant contribution to improving the understanding of using GAs to optimize the performance of CNN models for FER.

Corresponding Author:

Muhammad Sam'an
Departement of Informatics, Universitas Muhammadiyah Semarang
Semarang, Indonesia
Email: muhammad92sam@unimus.ac.id

1. INTRODUCTION

The face plays a crucial role in expressing emotions such as happiness, sadness, anger, and others [1]. Compared to voice and other body movements, the face predominantly conveys a person's emotions [2]. With technological advancements, facial expression recognition is becoming increasingly important, especially in the context of computer, mobile phone, and other technology usage. For instance, facial expression recognition can be utilized to assess customer satisfaction in online shopping. Facial data is captured through a camera and can be categorized based on the conditions of capture, whether ideal or not [3]-[8].

Deep learning (DL) is a part of machine learning that uses artificial neural networks with many hidden layers to automatically learn data representations or features. One of the DL models, convolutional neural network (CNN), is particularly effective in computer vision and can recognize facial features through convolution processes in images [9]-[11]. The advantage of CNN lies in its ability to ignore manual feature extraction [12]-[14] and its capability to retrain tasks for recognizing new objects using existing networks [15],[16]. Various CNN models, including Conv1D, Conv2D, Conv3D, and Conv4D, have shown significant progress in image classification and processing [17]-[19], demonstrating CNN's ability as the best model for addressing object recognition and detection problems [20].

Facial emotion recognition (FER) 2013, introduced by the international conference on machine learning (ICML) in 2013, is a unique emotion recognition dataset that encompasses challenging natural conditions and challenges. This dataset is used to test the performance of CNN models in emotion recognition, yielding results of approximately 60%-75% based on previous studies. Some tested methods include Tang's network structure, achieving an accuracy of 71.2% [21], Caffe-ImageNet with an accuracy of 65.5% [19], and multi-network fusion achieving an accuracy of 70.3% [22]. Nevertheless, previous research indicates that the accuracy performance of CNN for FER using the FER2013 dataset still requires improvement. The significance of proper hyperparameters in the construction of CNN model architecture, especially in the convolution layers, is acknowledged, but precise rules for determining hyperparameters remain unclear, with most determinations made through the intuition of experienced designers or through trial and error methods. Generally, the kernel size used varies, with different models such as LeNet-5 [23], Alexnet [24], and ZFNet employing different sizes [25], while other hyperparameters are determined based on the intuition of the designer.

Hyperparameter optimization is one of the approaches used to improve CNN performance during the learning process [26]-[28]. There are two techniques for hyperparameter optimization, namely manual and automatic. In image recognition, the manual method can improve CNN accuracy [29], but it is less effective due to its narrow parameter range, time-consuming nature, and requirement for very high accuracy [30]. Evolutionary deep learning (EDL) was developed for automatic hyperparameter optimization to improve CNN performance on FER2013 [31]. CNN-genetic algoritm (CNN-GA), a result of EDL development, plays a crucial role in hyperparameter optimization using a large search space of GA [32], representing a set of hyperparameters in CNN. CNN-GA achieved an average accuracy of 99.85% and training speed four times faster than the LeNet-5 model [33], [34]. Based on these advantages and performance, GA was chosen for hyperparameter optimization in the CNN model. In this study, we propose a method for tuning the hyperparameters of the CNN feature extraction step using a CNN-GA, with GA chromosomes representing each hyperparameter such as kernel size, number of filters, activation function, learning rate, and optimizer.

2. METHOD

There are 35,887 images, which are then split into two: training data and testing data. The proportion of the split FER2013 dataset is 80% for the training model, and the remaining 20% is used for evaluation purposes with unseen data. Emotion recognition in FER2013, as shown in Figure 1, is a supervised image recognition problem that is categorized into seven classes: anger, disgust, fear, happiness, sadness, surprise, and neutral. The Python programming language is applied to all hyperparameter optimization approaches. SciKit-Learn is used for matrix calculations. Pandas is used for dataset processing. Numpy is employed to handle all scientific computing. Keras and TensorFlow are utilized to build the CNN model. Distributed evolutionary algorithms in python (DEAP) is used to construct the GA.

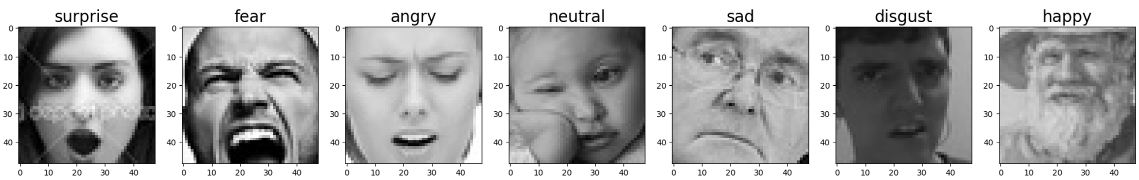

Figure 1. FER2013 dataset

2.1. CNN hyperparameters and architecture

We optimized 5 hyperparameters, with a complete description of the range or value and a list of all hyperparameters shown in Table 1, based on commonly optimized hyperparameters used for FER [35]. The proposed CNN model architecture comprises input, convolution, pooling, dropout, batch normalization, dense, and fully connected layers. The use of six two-dimensional convolutions (Conv2D) with 'same' padding values and 'he_normal' Kernel initialization is accompanied by an automatic hyperparameter optimization approach based on GA for the kernel, filter, and activation functions. Batch Normalization plays a role in stabilizing the input distribution layer, while the pooling layer works to reduce the image dimensions to lessen the neural network's weight. There are three 'MaxPool2D' type pooling layers, which operate by considering higher feature map values to reduce the image size, each with a size of 2x2. The role of the dense layer is to connect the neural network from the specified layer to each previous layer of neurons. The dropout layer is positioned on the fully connected layer and after the pooling layer. Details of the model architecture design are presented in Figure 2.

Table 1. Hyperparameters range and values

Hyperparameter	Description	Range/Values
LR	Learning rate	Min: 1e-3, Max: 1e-2
OP	Optimizer	[‘Adam’, ‘SGD’, ‘RMSprop’, ‘Adadelta’, ‘Adagrad’, ‘Adamax’, ‘Nadam’]
KZ	Kernel size (Convolutional)	Min: 2, Max: 1024
AC	Activation function (Convolutional)	[‘relu’, ‘elu’, ‘tanh’, ‘sigmoid’, ‘hard_sigmoid’, ‘softplus’,‘linear’]
NF	Number of filters (Convolutional)	Min: 1, Max: 9

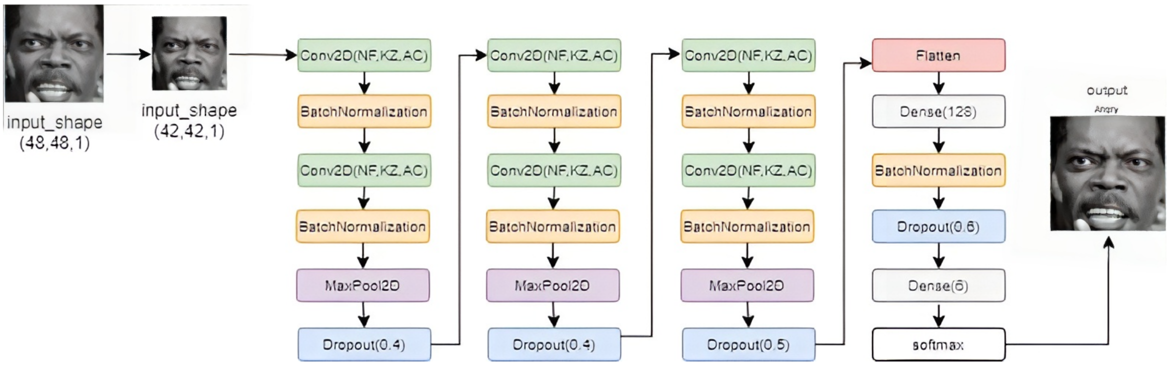

Figure 2. The design of the CNN model architecture

2.2. Hyperparameter tuning of CNN-GA

The flowchart of CNN-GA is shown in Figure 3. The first step involves preprocessing the data containing cropped images, removing duplicate images, resizing the images to 48x48 pixels, and converting them to grayscale. We follow the standard GA process, including population initialization, evaluation, selection, crossover, and mutation [36]. Population initialization is depicted in Figure 4, with the same chromosome length for each individual. We use accuracy score as the fitness function, where individuals with higher scores have a greater chance of being selected for the next generation. CNN-GA utilizes three operators: crossover, mutation, and selection, with crossover depending on the process of selecting parent individuals.

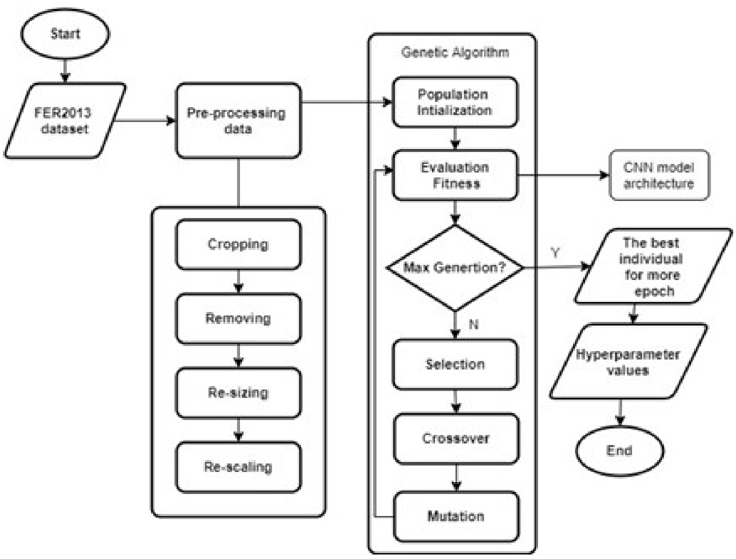

Figure 3. Flowchart of CNN-GA

We use three types of crossover operators: one-point, two-point, and multi-point. One-point crossover alters one point of the parent gene to produce offspring, while two-point uses two crossover points, and multi-point employs multiple crossover points. This prevents duplication of parental genes and enhances fitness values from generation to generation. Determining the incorrect crossover type could lead to premature optimization. Hence, mutation serves as an operator to vary genes and avoid uniform crossovers within the population. A higher mutation rate than the crossover rate could lead to issues in reaching the optimal solution. We set the mutation rate at 0.1 and the crossover rate at 0.2, based on personal instinct. The best individuals are selected using elitism, maintaining the algorithm's evolutionary process towards the best solution.

3. RESULTS AND DISCUSSION

GA is utilized to optimize the performance of the CNN, although the limited tools make it time-consuming. In this study, we set 3 generations (NGen) and 10 populations (Npop) to manage the GA operations. Elitism, which preserves the best individuals from each generation, and multi-parent recombination have proven effective in achieving optimal accuracy. Table 2 provides an overview of the required computational time and the best accuracy score achieved by GA. Although generation 3 requires a longer computational time compared to other stages, it yields better accuracy. It is important to note that after the crossover and mutation operations, there is fluctuation in the fitness values of individuals, which is overcome with the help of multi-parent recombination and elitism, ultimately enhancing performance in generation 3. This analysis highlights the importance of using the GA technique to improve CNN efficiency despite the longer computational time.

Table 2. The best result of hyperparameter

Convolution layer	Values				
	KZ	NF	AF	OP	LR
1	8	3x3	tanh	Adagrad	0.007
2	8	5x5	relu		
3	2	9x9	elu		
4	256	3x3	Sigmoid		
5	16	3x3	linear		
6	128	3x3	tanh		

Figure 5 visualizes the impact of CNN-GA based on the fitness value. During the initialization phase, the fitness value appears to fluctuate. The initialization process in the GA stage demonstrates an increase in accuracy from generation to generation. From the 10th population that yielded an accuracy of 58.72%, generation 1 achieved 71.69% (an increase of 12.97%), generation 2 reached 72.11% (an increase of 1.08%), and generation 3 attained 76.11% (an increase of 3.33%). This increase occurred after the application of multi-parent recombination and elitism operations. The filter function in the convolution layer plays a crucial role in extracting features from the image and ensuring pattern suitability with the image's sub-regions. The increase in the number of filters in the convolution process helps the model capture more features in the image, ultimately enhancing accuracy. Furthermore, the optimizer plays a vital role in the model's learning by updating neuron weights and guiding the process toward convergence. The activation function also plays a significant role in determining the output of each specific neuron. This analysis emphasizes the importance of the proper initialization process in the GA and the key roles of filters, optimizers, and activation functions in enhancing accuracy and model performance. The best hyperparameter results from the top-performing generation are detailed in Table 3.

Furthermore, Table 4 provides an informative description of the model accuracy performance on FER2013 test data. It is clear that all models trained in this study have exceeded the estimated performance of existing studies, which is about 65.5%. However, the success of CNN-GA, which achieved an accuracy performance of 76.11%, highlights the importance of rigorous hyperparameter optimization. These results show that with an efficient and accurate approach, we can optimize the model significantly, even surpassing the performance of existing research in FER. Therefore, hyperparameter optimization via the CNN-GA approach offers promising prospects for improving model capabilities in increasingly complex recognition tasks in the future.

Table 3. The computational time and best accuracy score

Experiment phase	Computational time	Best accuracy score (%)
Initial	70°30'57"	58.72
Generation 1	60°08'37"	71.69
Generation 2	140°18'50"	72.77
Generation 3	160°45'04"	76.11

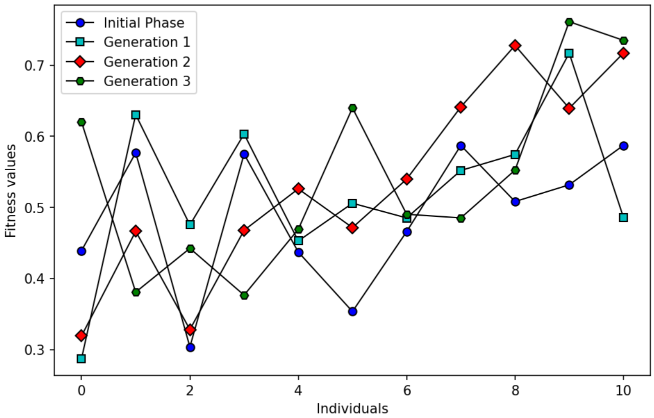

Figure 4. The impact of CNN-GA based on the fitness value

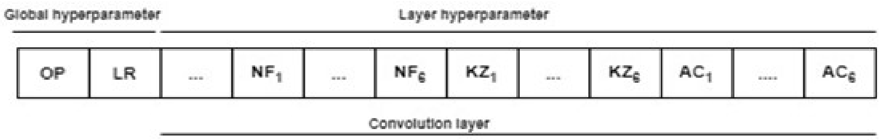

Figure 5. Population initialization of CNN-GA

Furthermore, Table 4 provides an informative description of the model accuracy performance on FER2013 test data. It is clear that all models trained in this study have exceeded the estimated performance of existing studies, which is about 65.5%. However, the success of CNN-GA, which achieved an accuracy performance of 76.11%, highlights the importance of rigorous hyperparameter optimization. These results show that with an efficient and accurate approach, we can optimize the model significantly, even surpassing the performance of existing research in FER. Therefore, hyperparameter optimization via the CNN-GA approach offers promising prospects for improving model capabilities in increasingly complex recognition tasks in the future.

Table 4. TFER2013 testing accuracy comparison

Ref.	Model	Testing accuracy (%)
[21]	Tang's network structure	71.2
[37]	Caffe-imageNet	65.5
[22]	MNF CNN+L2 SVM	70.3
[38]	Raspberry Pi	65.97
[39]	DenseNet	63.50
[40]	VGG16	69.40
[41]	Attention CNN	70.02
[42]	ResNet with gate implementation	71.80
[43]	VGGNet	73.28
[44]	VGG progressive SpinalNet	74.39
	VGG SpinalNet	74.45
[45]	Mini-Xception	66.00
[46]	CNN with transfer learning	72.00
[47]	CNN with HOG feature	75.10
Our model	CNN-GA	76.11

4. CONCLUSION

The development of DL for FER is growing rapidly and is being applied in various fields. Hybrid DL with evolutionary algorithms serves as a problem-solving approach in sectors such as security, health, games, and others. Valid determination of hyperparameters can significantly impact DL performance, particularly for CNN. Hence, we propose CNN-GA for optimizing hyperparameters. The required computation time is 37 hours 57 minutes 55 seconds, with generation 3 taking the longest computation time, namely 16 hours 45 minutes 4 seconds. However, generation 3 achieves an accuracy of 76.11%, which is the best compared to other generations. This suggests that computation time does not affect the resulting accuracy. The accuracy results indicate that the more generations, the higher the achieved accuracy. Furthermore, we compare the proposed model with the results of previous studies. The accuracy results of CNN-GA outperform the existing models. Further research in this direction is necessary to explore its applicability in other domains and to optimize its performance even further.

ACKNOWLEDGEMENTS

The authors would like to thank two anonymous reviewers for their constructive comments and suggestions, which have helped to improve this paper. Furthermore, this research was partially supported by Grant of LPPM 2022 Universitas Muhammadiyah Semarang.

REFERENCES

[1] H. Artanto and F. Arifin, "Emotions and gesture recognition using affective computing assessment with deep learning," *International Journal of Artificial Intelligence (IJAI)*, vol. 12, no. 3, pp. 1419–1427, Sep. 2023, doi: 10.11591/ijai.v12.i3.pp1419-1427.

[2] N. Samadiani *et al.*, "A review on automatic facial expression recognition systems assisted by multimodal sensor data," *Sensors (Switzerland)*, vol. 19, no. 8, p. 1863, Apr. 2019, doi: 10.3390/s19081863.

[3] M. G. Villanueva and S. R. Zavala, "Deep neural network architecture: application for facial expression recognition," *IEEE Latin America Transactions*, vol. 18, no. 7, pp. 1311–1319, Jul. 2020, doi: 10.1109/TLA.2020.9099774.

[4] N. C. Kit, C. P. Ooi, W. H. Tan, Y. F. Tan, and S. N. Cheong, "Facial emotion recognition using deep learning detector and classifier," *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 13, no. 3, pp. 3375–3383, Jun. 2023, doi: 10.11591/ijece.v13i3.pp3375-3383.

[5] S. A. Rizwan, A. Jalal, and K. Kim, "An accurate facial expression detector using multi-landmarks selection and local transform features," *in 3rd International Conference on Advancements in Computational Sciences*, ICACS 2020, Feb. 2020, pp. 1–6, doi: 10.1109/ICACS47775.2020.9055954.

[6] T. T. Kaleekal and J. Singh, "Facial expression recognition using hahn moment on facial patches," *in 2019 4th International Conference on Recent Trends on Electronics, Information, Communication and Technology (RTEICT)*, pp. 1338–1344, 2019, doi: 10.1109/RTEICT46194.2019.9016953.

[7] K. L. Ong, C. P. Lee, H. S. Lim, and K. M. Lim, "Speech emotion recognition with light gradient boosting decision trees machine," *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 13, no. 4, pp. 4020–4028, Aug. 2023, doi: 10.11591/ijece.v13i4.pp4020-4028.

[8] M. M. T. Zadeh, M. Imani, and B. Majidi, "Fast facial emotion recognition using convolutional neural networks and gabor filters," *in 2019 IEEE 5th Conference on Knowledge Based Engineering and Innovation*, KBEI 2019, Feb. 2019, pp. 577–581, doi: 10.1109/KBEI.2019.8734943.

[9] A. S. Sams and A. Zahra, "Multimodal music emotion recognition in Indonesian songs based on CNN-LSTM, XLNet transformers," *Bulletin of Electrical Engineering and Informatics (BEEI)*, vol. 12, no. 1, pp. 355–364, Feb. 2023, doi: 10.11591/eei.v12i1.4231.

[10] R. Chauhan, K. K. Ghanshala, and R. Joshi, "Convolutional neural network (CNN) for image detection and recognition," *in 2018 First International Conference on Secure Cyber Computing and Communication (ICSCCC)*, Dec. 2018, pp. 278–282, doi: 10.1109/ICSCCC.2018.8703316.

[11] W. Mo, X. Luo, Y. Zhong, and W. Jiang, "Image recognition using convolutional neural network combined with ensemble learning algorithm," *Journal of Physics: Conference Series*, vol. 1237, no. 2, p. 022026, Jun. 2019, doi: 10.1088/1742-6596/1237/2/022026.

[12] P. Mishra and P. V. V. S. Srinivas, "Facial emotion recognition using deep convolutional neural network and smoothing, mixture filters applied during preprocessing stage," *International Journal of Artificial Intelligence (IJ-AI)*, vol. 10, no. 4, p. 889, Dec. 2021, doi: 10.11591/ijai.v10.i4.pp889-900.

[13] Y. Wang, L. Dang, and J. Ren, "Forest fire image recognition based on convolutional neural network," *Journal of Algorithms and Computational Technology*, vol. 13, p. 174830261988768, Jan. 2019, doi: 10.1177/1748302619887689.

[14] K. Slimani, M. Kas, Y. El Merabet, Y. Ruichek, and R. Messoussi, "Local feature extraction based facial emotion recognition: a survey," *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 10, no. 4, pp. 4080–4092, Aug. 2020, doi: 10.11591/ijece.v10i4.pp4080-4092.

[15] P. Kulkarni and T. M. Rajesh, "Analysis on techniques used to recognize and identifying the human emotions," *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 10, no. 3, pp. 3307–3314, Jun. 2020, doi: 10.11591/ijece.v10i3.pp3307-3314.

[16] S. Indolia, A. K. Goswami, S. P. Mishra, and P. Asopa, "Conceptual understanding of convolutional neural network- a deep learning approach," *Procedia Computer Science*, vol. 132, pp. 679–688, 2018, doi: 10.1016/j.procs.2018.05.069.

[17] A. A. Elngar *et al.*, "Image classification based on CNN: a survey," *Journal of Cybersecurity and Information Management*, pp. 18-50, 2021, doi: 10.54216/jcim.060102.

[18] I. M. A. Wirawan, R. Wardoyo, and D. Lelono, "The challenges of emotion recognition methods based on electroencephalogram signals: a literature review," *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 12, no. 2, pp. 1508–1519, Apr. 2022, doi: 10.11591/ijece.v12i2.pp1508-1519.

[19] H. Qassim, A. Verma, and D. Feinzimer, "Compressed residual-VGG16 CNN model for big data places image recognition," *in 2018 IEEE 8th Annual Computing and Communication Workshop and Conference*, CCWC 2018, Jan. 2018, vol. 2018-January, pp. 169–175, doi: 10.1109/CCWC.2018.8301729.

[20] M. Sornam, K. Muthusubash, and V. Vanitha, "A survey on image classification and activity recognition using deep convolutional neural network architecture," *in 2017 9th International Conference on Advanced Computing*, ICoAC 2017, Dec. 2018, pp. 121–126, doi: 10.1109/ICoAC.2017.8441512.

[21] Y. Tang, "Deep learning using linear support vector machines," *Arxiv*, 2013, [Online]. Available: http://arxiv.org/abs/1306.0239.

[22] V. Tumen, O. F. Soylemez, and B. Ergen, "Facial emotion recognition on a dataset using convolutional neural network," *in 2017 International Artificial Intelligence and Data Processing Symposium (IDAP)*, Sep. 2017, pp. 1–5, doi: 10.1109/IDAP.2017.8090281.

[23] 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, 1998, doi: 10.1109/5.726791.

[24] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," *Communications of the ACM*, vol. 60, no. 6, pp. 84–90, May 2017, doi: 10.1145/3065386.

[25] Y. Lecun and M. Zeiler, "Visualizing and understanding convolutional networks," *Computer Vision - Eccv 2014*, Pt I, vol. 8689, pp. 1–11, 2015.

[26] X. Xiao, M. Yan, S. Basodi, C. Ji, and Y. Pan, "Efficient hyperparameter optimization in deep learning using a variable length genetic algorithm," *Arxiv*, 2020, [Online]. Available: http://arxiv.org/abs/2006.12703.

[27] R. Andonie and A. C. Florea, "Weighted random search for CNN hyperparameter optimization," *International Journal of Computers, Communications and Control*, vol. 15, no. 2, pp. 1–11, Mar. 2020, doi: 10.15837/IJCCC.2020.2.3868.

[28] F. M. Talaat and S. A. Gamel, "RL based hyper-parameters optimization algorithm for convolutional neural network," *Journal of Ambient Intelligence and Humanized Computing*, vol. 14, no. 10, pp. 13349–13359, Oct. 2023, doi: 10.1007/s12652-022-03788-y.

[29] S. Albahli, F. Alhassan, W. Albattah, and R. Ullah, "Handwritten digit recognition: hyperparameters-based analysis," *Applied Sciences (Switzerland)*, vol. 10, no. 17, p. 5988, Aug. 2020, doi: 10.3390/app10175988.

[30] Z. Fouad, M. Alfonse, M. Roushdy, and A. B. M. Salem, "Hyper-parameter optimization of convolutional neural network based on particle swarm optimization algorithm," *Bulletin of Electrical Engineering and Informatics (BEEI)*, vol. 10, no. 6, pp. 3377–3384, Dec. 2021, doi: 10.11591/eei.v10i6.3257.

[31] H. Iba and N. Noman, "Deep neural evolution: deep learning with evolutionary computation," *Natural Computing Series*, 2020, [Online]. Available: https://link.springer.com/book/10.1007%2F978-981-15-3685-4.

[32] S. Loussaief and A. Abdelkrim, "Convolutional neural network hyper-parameters optimization based on Genetic Algorithms," *International Journal of Advanced Computer Science and Applications*, vol. 9, no. 10, pp. 252–266, 2018, doi: 10.14569/IJACSA.2018.091031.

[33] N. M. Aszemi and P. D. D. Dominic, "Hyperparameter optimization in convolutional neural network using genetic algorithms," *International Journal of Advanced Computer Science and Applications*, vol. 10, no. 6, pp. 269–278, 2019, doi: 10.14569/ijacsa.2019.0100638.

[34] W. Lu, H. Mao, F. Lin, Z. Chen, H. Fu, and Y. Xu, "Recognition of rolling bearing running state based on genetic algorithm and convolutional neural network," *Advances in Mechanical Engineering*, vol. 14, no. 4, 2022, doi: 10.1177/16878132221095635.

[35] S. S. Bellamkonda, "Facial emotion recognition by hyper parameter tuning of convolutional neural network using genetic algorithm," *Blekinge Institute of Technology*, pp. 1–104, 2021.

[36] R. M. Loses, "An introduction to genetic algorithms," *Information Processing and Management*, vol. 33, no. 3, p. 407, May 1997, doi: 10.1016/s0306-4573(97)88274-x.

[37] Y. Jia *et al.*, "Caffe: Convolutional architecture for fast feature embedding," *in MM 2014 - Proceedings of the 2014 ACM Conference on Multimedia*, Nov. 2014, pp. 675–678, doi: 10.1145/2647868.2654889.

[38] L. Zahara, P. Musa, E. Prasetyo Wibowo, I. Karim, and S. B. Musa, "The facial emotion recognition (FER-2013) dataset for prediction system of micro-expressions face using the convolutional neural network (CNN) algorithm based raspberry pi," *in 2020 5th International Conference on Informatics and Computing*, ICIC 2020, Nov. 2020, pp. 1–9, doi: 10.1109/ICIC50835.2020.9288560.

[39] K. C. O'Nell, R. Saxe, S. Anzellotti, K. C. O. Nell, R. Saxe, and S. Anzellotti, "Recognition of identity and expressions as integrated processes," *PsyArXiv*, no. May, pp. 1–18, 2019, doi: 10.31234/osf.io/9c2e5.

[40] G. P. Kusuma, A. Jonathan, and P. Lim, "Emotion recognition on FER-2013 face images using fine-tuned VGG-16," *Advances in Science, Technology and Engineering Systems*, vol. 5, no. 6, pp. 315–322, 2020, doi: 10.25046/aj050638.

[41] S. Minaee, M. Minaei, and A. Abdolrashidi, "Deep-emotion: facial expression recognition using attentional convolutional network," *Sensors*, vol. 21, no. 9, p. 3046, Apr. 2021, doi: 10.3390/s21093046.

[42] F. Wang, H. Chen, L. Kong, and W. Sheng, "Real-time facial expression recognition on robot for healthcare," *in 2018 International Conference on Intelligence and Safety for Robotics*, ISR 2018, Aug. 2018, pp. 402–406, doi: 10.1109/IISR.2018.8535710.

[43] Y. Khaireddin and Z. Chen, "Facial emotion recognition: state of the art performance on FER2013," *Arxiv*, 2021, [Online]. Available: http://arxiv.org/abs/2105.03588.

[44] B. E. Santoso and G. P. Kusuma, "Facial emotion recognition on FER2013 using Vggspinalnet," *Journal of Theoretical and Applied Information Technology*, vol. 100, no. 7, pp. 2088–2102, 2022.

[45] O. Arriaga, M. Valdenegro-Toro, and P. G. Plöger, "Real-time convolutional neural networks for emotion and gender classification," *ESANN 2019 - Proceedings, 27th European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning*, pp. 221–226, 2019.

[46] J. Li, Y. Wang, J. See, and W. Liu, "Micro-expression recognition based on 3D flow convolutional neural network," *Pattern Analysis and Applications*, vol. 22, no. 4, pp. 1331–1339, Nov. 2019, doi: 10.1007/s10044-018-0757-5.

[47] V. S. Amal, S. Suresh, and G. Deepa, "Real-time emotion recognition from facial expressions using convolutional neural network with FER2013 dataset," *in Smart Innovation, Systems and Technologies*, vol. 243, 2022, pp. 541–551.

BIOGRAPHIES OF AUTHORS

Muhammad Sam'an received Bachelor Degree from Universitas Negeri Semarang and Master Degree from Universitas Diponegoro in Mathematics 2010 and 2016 respectively. His research interests are in optimization, fuzzy mathematics and computational mathematics. He can be contacted at email: muhammad92sam@unimus.ac.id.

Safuan received the Master Degree in Informatics Engineering from Dian Nuswantoro University (UDINUS) in 2015. Currently, he is a lecturer in informatics Engineering at Muhammadiyah University, Semarang (UNIMUS). His research interests include data mining, programming and web security. He can be contacted at email: safuan@unimus.ac.id.

Muhammad Munsarif received the Master Degree in Computer science from Dian Nuswantoro University (UDINUS) in 2002. Currently, he is a lecturer in informatics Engineering at Muhammadiyah University, Semarang (UNIMUS). His research interests include computer vision, data science and technopreneuership. He can be contacted at email: m.munsarif@unimus.ac.id.