14th Annual International Conference on Industrial Engineering and Operations Management Dubai United Arab Emirates (UAE), February 12-14, 2024

Publisher: IEOM Society International, USA DOI: 10.46254/AN14.20240540
Published: February 12, 2024

Module-based Convolutional Neural Network Structure Optimization

Ivan Kristianto Singgih
Assistant Professor, Department of Industrial Engineering
University of Surabaya, Indonesia
The Indonesian Researcher Association in South Korea (APIK)
Seoul, 07342, South Korea
Kolaborasi Riset dan Inovasi Industri Kecerdasan Artifisial (KORIKA)
Jakarta, Indonesia
ivanksinggih@staff.ubaya.ac.id

Evan Sutrisno and Michael Abednego
Undergraduate Student, Department of Industrial Engineering
University of Surabaya
Indonesia
evansutrisno02@gmail.com, s160320023@student.ubaya.ac.id

Dina Natalia Prayogo
Lecturer, Department of Industrial Engineering
University of Surabaya
Indonesia
dnprayogo@staff.ubaya.ac.id

Abstract

Convolutional Neural Network (CNN) is used for processing image data in various applications, including logistics systems, production, etc. This research proposes a method for optimizing the structure of CNN. Many CNN models already use various structures, but there is no certain well-known framework for designing high-accuracy CNN models yet. Unlike previous studies, this study identifies combinations of various types and number of layers in CNN. Considering a limited number of alternative layer combinations when designing CNN models would reduce the computation time of the model generation (and increase the performance of the CNN model because it can consider more design alternatives during the model generation phase, given the available time). This study (1) extracts the combination of layers from many state-of-the-art CNN models that ensure the best information is obtained and (2) proposes a module-based framework to optimize the CNN structure. The numerical experiment shows that the proposed framework produces better accuracy than the compared best-known models. This research will provide insights on good combinations of layers and help researchers develop new CNN structures with high accuracy.

Keywords
Convolutional Neural Network, image processing, metaheuristics, MNIST, optimization.

1. Introduction

Implementation and development of Industry 4.0 and intelligent systems are increasing very rapidly. These systems collect a lot of data and use it for real-time decision-making. One of the widely used technologies is the prediction of data values using the Convolutional Neural Network (CNN) method. The CNN method automatically identifies important factors from image data (Wu et al. 2023).

The CNN method has been used to identify image data (Capua et al. 2023), identify component damage in the remanufacturing process within reverse logistics systems (Schlüter et al. 2021), identify ships in various weather conditions (Liu et al. 2021), etc. CNN is not only used for image data analysis but is also used in solving other cases, such as the prediction of train travel time (Guo et al. 2022), integration of supply optimization and demand prediction in logistics systems (Ren et al. 2020), prediction of demand for food distribution in urban areas (Crivellari et al. 2022), etc.

An example of the CNN structure is shown in Figure 1 (Wang and Su 2022). CNN is composed of several types of layers with different functions. Each layer has parameters. Values of the parameters need to be optimized to improve the accuracy of the CNN prediction results. The layers include the convolutional, pooling, and fully connected layers (Taye 2023). A calculation method based on a certain activation function is required to produce a fully connected layer.

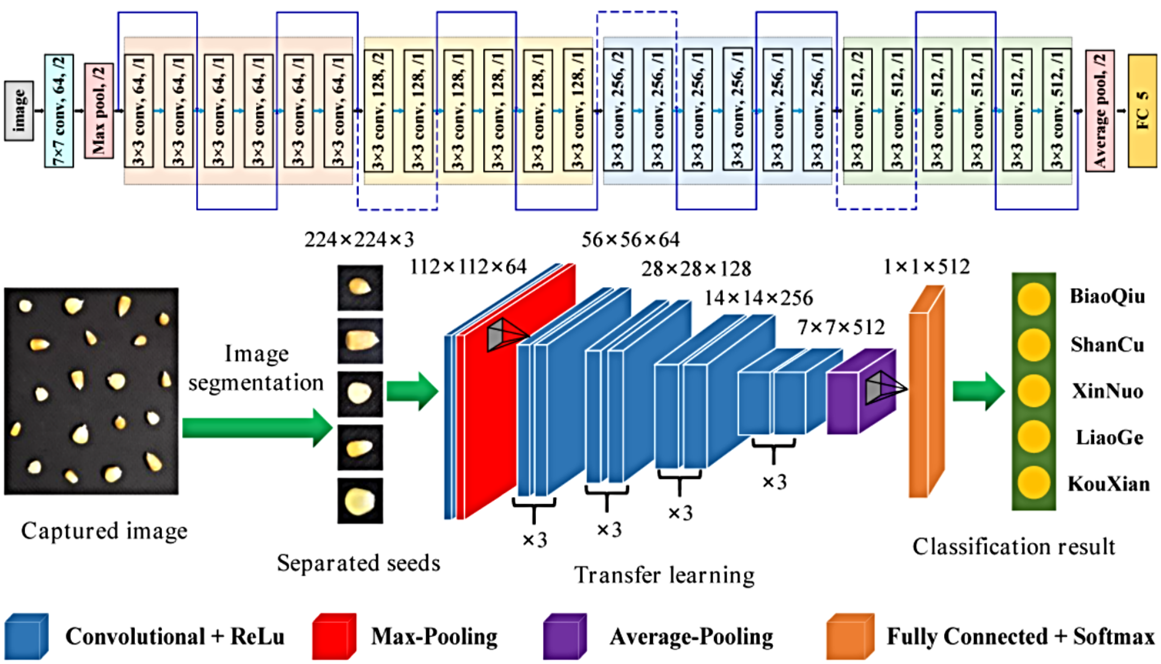


Figure 1. An example of Convolutional Neural Network structure

Until now, the structure of the CNN method continues to develop to increase prediction accuracy. An example of a comparison of the quality of human predictions using several CNN structures, measured in accuracy (AUC) and prediction error (EER), is shown in Batchuluun et al. (2018). The comparison might not only consider the accuracy values for each CNN structure but also the complexity of the model and the number of parameters in the CNN structure (AI Summer 2023). Researchers always try to find the best CNN structure. The CNN structures are developed every year (Medium 2023).

The objective of this study is to (1) identify the combination of layers in state-of-the-art CNN models and (2) propose a module-based framework for optimizing CNN structures. Such a summary could be a good reference for the next researchers when developing their new CNN models.

The structure of this paper is as follows. Section 2 explains the steps used in this study. Section 3 summarizes the CNN structures presented in the related previous studies, the module-based framework used for optimizing the CNN structures, and the experiment settings. Section 4 presents the experiment results. Section 5 concludes the study.

2. Literature Review

Studies on CNN structure design can be divided into: (1) CNN structure parameter optimization with a predetermined sequence of CNN layer types, and (2) hyperparameter optimization for a given complete CNN structure. In the first group of studies, the sequence of layers is predetermined, e.g., convolutional layer(s) than fully connected layer(s). Metaheuristics were used to determine the number of layers for each layer type and the parameters for each layer, e.g., number of nodes, activation function, number of batches, learning rate, etc. Studies classified into this group are Kabir Anaraki et al. (2019) and Tsoumalis et al. (2021). Studies in the second group considered a given complete CNN structure, which does not change. The parameters in the predetermined structure are optimized. Studies in this second group are Ijjina and Chalavadi (2016), Johnson et al. (2020), Rikhtegar et al. (2016), and Vetrimani et al. (2023). The disadvantages of studies in both groups are the performance of the models is limited because the CNN structure change is limited.

This study optimizes the CNN structure differently from studies in both groups above. This study allows any layer sequence generation without predetermining the sequence of layer types, as conducted by the studies in the first group. To maximize the performance of the CNN structure, this study optimizes the parameters of the layers using metaheuristics. Considering various layer types and parameters would allow evaluating more numbers of CNN structures and allow obtaining a better final structure at the end of the metaheuristics search.

3. Methods

The stages of this research are as follows:
1. Selection of parameters to be optimized in CNN.
2. Determination of alternative layer combination restrictions in CNN, considering the current CNN structure (state-of-the-art) to reduce model complexity and enable the search for structures with better accuracies.
3. Proposition of a module-based framework for optimizing CNN structures.

3.1 Selection of Parameters to be Optimized in CNN

In the early stages, a literature study is carried out to understand which parameters affect the quality of the CNN prediction results. This study observes various CNN parameters and their effects on the accuracy of the CNN model predictions. An example of a list of parameters to be considered is the one listed in the complete CNN structure explanation example in Figure 2 (Topbots 2023).

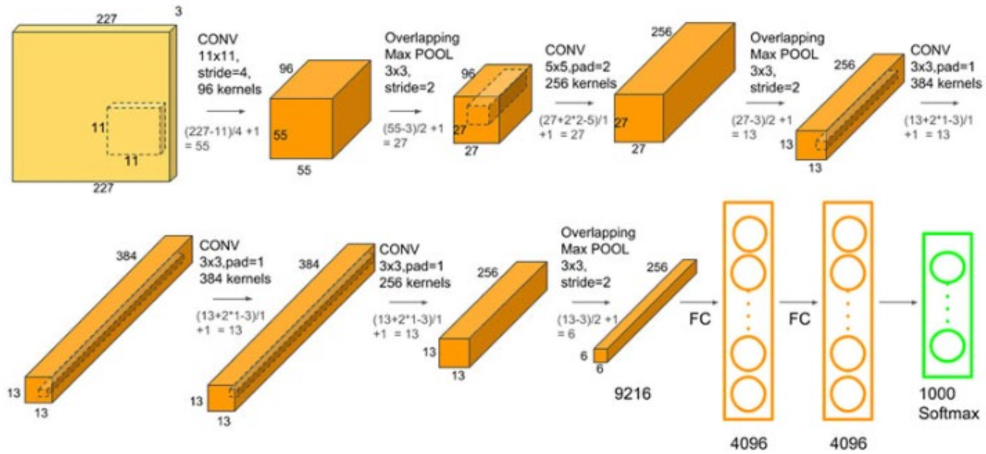
3.2 Summarizing Information of Layers Used in Previous Studies

Limiting layer combinations will reduce the search time for the new CNN structure design. Several previous studies have carried out the approach by limiting the hyperparameters of certain layers (Johnson et al. 2020). Unlike that concept, this study discusses how to identify a combination of the layer structure with limited parameter values obtained from state-of-the-art methods to design good CNN models. This study observes some of the best CNN models used for the prediction using the MNIST data, as shown in Figure 3 (MetaAI 2023).

3.3 Proposing Module-based CNN Structure Optimization Framework

Various optimization methods exist in the operations research field, including metaheuristics like genetic algorithm (GA). Metaheuristic has been proven to efficiently optimize various problems with multiple information types like the ones considered in our study (the CNN layer types and parameters of the specific layer). Examples of such metaheuristic methods are presented in Amirghasemi et al. (2023) and Wen et al. (2023). Such metaheuristic methods represent the solutions in a module-like structure to allow searching for new solutions through exploitation and exploration techniques. Exploitation works by combining good parts of the solution representation, e.g., by exchanging the position of existing parts in an existing solution. Exploration is conducted by introducing a new value into an existing solution, e.g., by generating a random value. GA has been proven to be the best metaheuristics, e.g., when solving traveling salesman problems (Toaza and Esztergár-Kiss 2023) and production scheduling cases (Peiris et al. 2023).

The GA framework is presented in Figure 4. For the initial population, best-known solutions for solving the MNIST dataset are included, and the remaining chromosomes are randomly generated. A single point crossover is performed by selecting two parents randomly, based on their prediction accuracy values; meanwhile, mutation is performed for each chromosome after the whole crossover operations. Three types of mutations are performed: (1) layer removal, (2) layer addition, and (3) layer parameter change. A Boltzmann threshold is applied to allow selecting a slightly worse solution after a mutation operation. During any chromosome generation or modification, the following characteristics of the CNN structure are maintained, e.g., (1) the predetermined minimum and maximum length of layers, (2) the final number of nodes (to have at least ten nodes for the MNIST classification), (3) an increasing number of filters for the convolutional layers, and (4) a decreasing number of nodes for the fully connected layers. Any operation could be restarted if necessary to generate feasible solutions. At the end of each population, an elitism chromosome is stored in the next population to maintain the GA convergence. GA parameters used for the search are listed in Table 1.



Layer	Type	Maps	Size	Kernel size	Stride	Padding	Activation
out	Fully connected	__	1000	__	__	__	Softmax
F9	Fully connected	__	4096	__	__	__	ReLU
F8	Fully connected	__	4096	__	__	__	ReLU
C7	Convolution	256	13*13	3*3	1	Same	ReLU
C6	Convolution	384	13*13	3*3	1	Same	ReLU
C5	Convolution	384	13*13	3*3	1	Same	ReLU
S4	Max Pooling	256	13*13	3*3	2	Valid	__
C3	Convolution	256	27*27	5*5	1	Same	ReLU
S2	Max Pooling	96	27*27	3*3	2	Valid	__
C1	Convolution	96	55*55	11*11	4	Valid	ReLU
In	Input	3(RGB)	227*227	__	__	__	__

Figure 2. List of complete parameters in the AlexNet structure

Table 1. The genetic algorithm parameters

Parameter	Value
population_size	10
num_of_population	10
crossover_rate	0.8
mutation_rate	0.2
percentage_for_gene_addition_or_removal	0.25
initial temperature	50
final temperature	25

Rank	Model	Percentage error	Accuracy	Trainable Parameters	Error rate	Extra Training Data	Paper	Code	Result	Year
1	**Heterogeneous ensemble with simple CNN**	0.09	99.91			×	An Ensemble of Simple Convolutional Neural Network Models for MNIST Digit Recognition			2020
2	**Branching/Merging CNN + Homogeneous Vector Capsules**	0.13	99.87	1,514,187		×	No Routing Needed Between Capsules			2020
3	**EnsNet** (Ensemble learning in CNN augmented with fully connected subnetworks)	0.16	99.84			×	Ensemble learning in CNN augmented with fully connected subnetworks			2020
4	**Efficient-CapsNet**	0.16	99.84	161,824		×	Efficient-CapsNet: Capsule Network with Self-Attention Routing			2021
5	**SOPCNN** (Only a single Model)	0.17	99.83	1,400,000		×	Stochastic Optimization of Plain Convolutional Neural Networks with Simple methods			2020
6	**RMDL** (30 RDLs)	0.18	99.82			×	RMDL: Random Multimodel Deep Learning for Classification			2018
7	**DropConnect**	0.21	99.79			×	Regularization of Neural Networks using DropConnect			2013
8	**MCDNN**	0.23				×	Multi-column Deep Neural Networks for Image Classification			2012

Figure 3. CNN model comparison when solving MNIST data

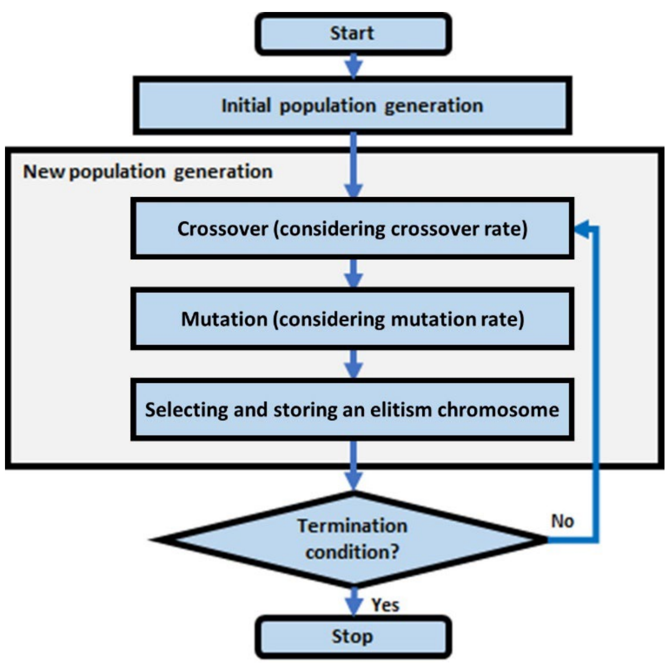


Figure 4. The genetic algorithm framework.

In this study, CNN models with the following layers are considered: convolution layers, activation layers (ReLU, Tanh, sigmoid, etc.), pooling layers, fully connected layers, and softmax functions. The studied CNN models are limited to the ones with these layers because of the following reasons: (1) for an easier and better comparison of the models, considering that CNN models have other various types of layers, and (2) to initially test whether identifying the relationships between these layers could allow the next researchers to design a better CNN model, as suggested by this study.

Among the 91 best CNN models in MetaAI (2023), the five best models with the layers mentioned in this study are shown in Table 2. The summary of the models is shown in Table 3, which lists the used layers and the considered parameter types (number and size of kernels, stride size, padding size, etc.). The maximum and minimum values of each parameter are listed. The # of layers per study refers to the number of each layer type in each study.

In Table 3, based on the initial minimum (*init_min*) and maximum values (*init_max*) of each parameter, a new minimum (*new_min*) and maximum values (*new_max*) are calculated. The new minimum and maximum values are calculated in Equations (1) and (2). For parameters with the same initial minimum and maximum values, the maximum values are additionally modified to allow observing some other potential values. For parameters with no specific number values, all the options for values are considered.

Table 2. Models with the best accuracy

Method Rank	**Model**	**Accuracy (%)**
7	Multi-column Deep Neural Networks	99.77
8	Augmented Pattern Classification (APAC) with Neural Networks	99.77
25	Deep Big Simple Neural Nets	99.6
76	Deep Neural Network-5	97.2
77	Deep Neural Network-3	97

Table 3. Summary of used layers in the studies and their parameter values

Layer Types	# of Layers (Min, Max)	Parameters	Value (Init_Min, Init_Max)	Value (New_Min, New_Max)
Convolutional Layer	(2,2)	Number of Kernels	(20,40)	(15,45)
		Kernel Size	(4,5)	(3,6)
		Stride Size	(1,1)	(1,2)
		Amount of Zero Padding	(0,0)	(0,same)
		Activation Function	scaled hyperbolic tangent, ReLU	
Pooling Layer	(2,2)	Type	max	
		Stride Size	(2,3)	(2,4)
		Activation Function	linear, ReLU	
Fully Connected Layer	(2,5)	Size	(64,2500)	(10,3100)
		Activation Function	scaled hyperbolic tangent, ReLU, ExtendeD Exponential Linear Unit (DELU)	
Softmax	(1,1)	Size	(10,10)	
Total number of layers			(2,7)	(2,9)
Total number of fully connected layers			(2,5)	(1,6)

Table 4. Summary on pairs of subsequent layers of the studies

		Next Layer			
		Convolutional Layer	Pooling Layer	Fully connected Layer	Softmax
Previous Layer	Convolutional Layer	(0,0,0)	(2,2,2)	(0,0,0)	(0,0,0)
	Pooling Layer	(1,1,2)	(0,0,0)	(1,1,2)	(0,0,0)
	Fully Connected Layer	(0,0,0)	(0,0,0)	(1,4,4)	(1,1,5)
	Softmax	(0,0,0)	(0,0,0)	(0,0,0)	(0,0,0)

The relationship between the layers of the selected papers is shown in Table 4. The number of occurrences for each pair of layers is shown by its minimum and maximum values. # of studies refers to the number of studies considering each pair of consecutive layers. Table 4 lists each pair of consecutive layers from the first to the last layer. The purpose is to identify how the layers could be combined to produce a model with good accuracy. Parameter values in Table 3 are then further considered to design better CNN models.

$$new\ minimum\ value = \left\lfloor init\_min - \frac{(init\_max - init\_min)}{2} \right\rfloor \tag{1}$$

$$new\ maximum\ value = \left\lceil init\_max + \frac{(init\_max - init\_min)}{2} \right\rceil \tag{2}$$

Parameters used to train the CNN models within the GA framework are listed in Table 5. The number of epochs used to train each CNN structure is set equal to 1 to reduce the computational time.

Table 5. The CNN model training parameters

Parameter	Value
number_of_epochs	1
parameter_batch_size	2,048

4. Results and Discussion

The convergence of the GA is shown in Figure 5. Such a convergence is guaranteed because of the identification and storage of the elitism chromosomes from each population. The best CNN structure obtained through the search is presented in Figure 6.

Comparison between the obtained best CNN structure and the best known solutions from Table 2 are shown in Table 6. Parameters of each CNN structure were trained for one epoch. Ideally, the evaluation of each CNN structure should be conducted on multiple epochs, but it would be very time consuming. It is shown that the obtained best CNN structure obtained in this study has the best accuracy when compared with all of the best known models.

The obtained final CNN structure was tested for 10 epochs and several batch size values to further assess the quality of the obtained best structure. The accuracy values are shown in Figure 7. The best accuracy was obtained when setting the batch size to 32. With the batch size equal to 32, different number of epochs are tested, and the results are shown in Figure 8. The accuracy increases up to a stable accuracy value of 98.64% within 60 epochs.

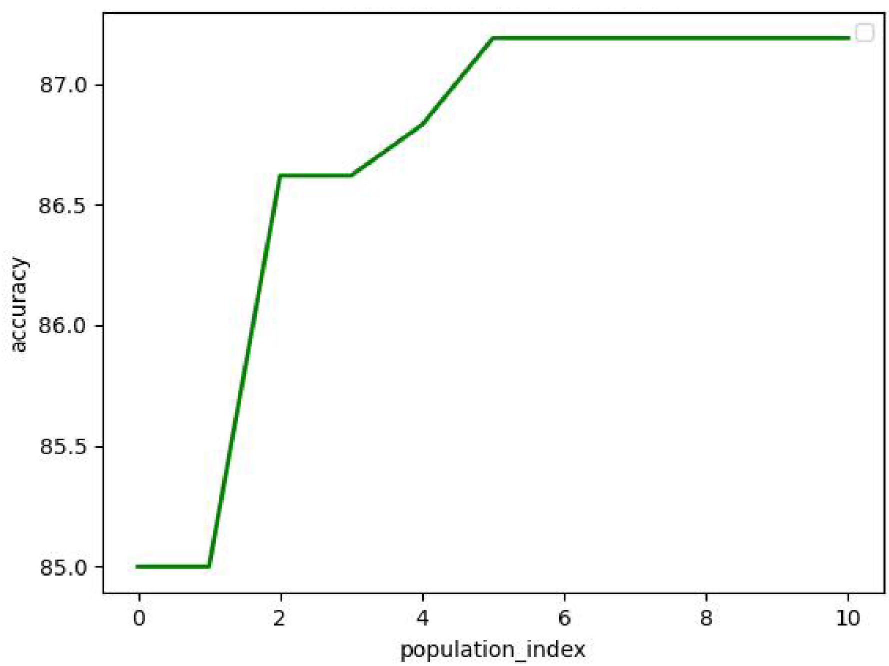


Figure 5. The convergence of the genetic algorithm search.

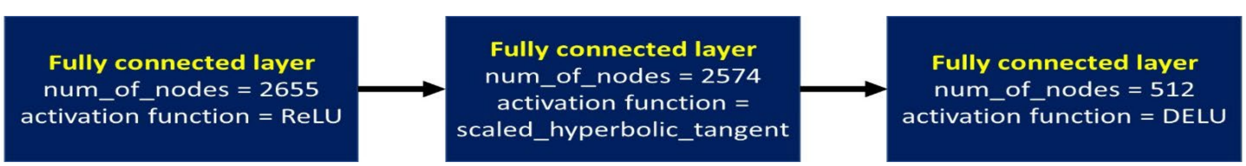


Figure 6. The best CNN structure obtained after running the GA.

Table 6. Comparison between the obtained structure and the best known structures

Model	Accuracy (%)
Multi-column Deep Neural Networks	11.36
Augmented Pattern Classification (APAC) with Neural Networks	79.29
Deep Big Simple Neural Nets	84.44
Deep Neural Network-5	77.73
Deep Neural Network-3	83.34
This study	87.03

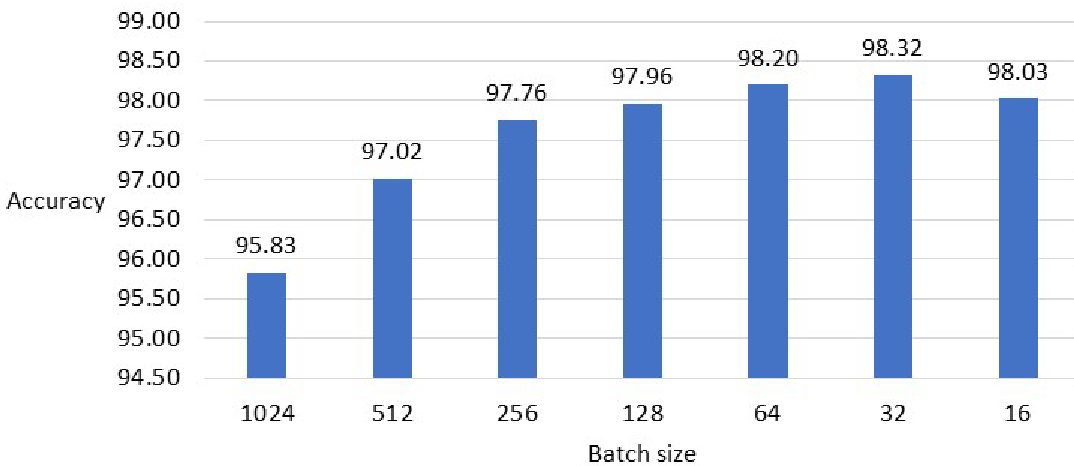


Figure 7. Accuracy values of the best CNN structure when considering different batch sizes (number of epochs = 10).

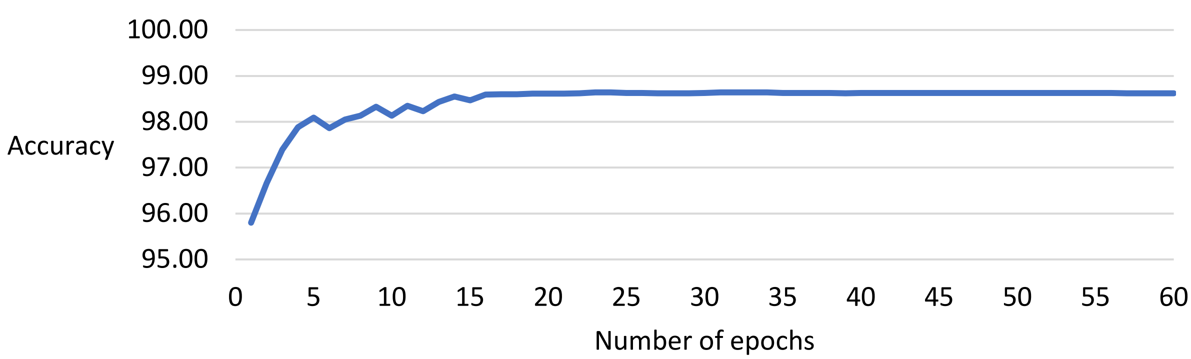


Figure 8. Accuracy values of the best CNN structure when considering different number of epochs (batch size = 32).

The numerical experiment results show the effectiveness of the proposed module-based CNN structure optimization algorithm.

5. Conclusion

This study observes the designs of the best CNN models with the purpose of suggesting a good framework for the next CNN models. Seven studies among the 91 best studies solving the MNIST datasets were discussed. This study extracted and reported the best parameter values and relationships between layers. A module-based framework for optimizing the CNN structure was proposed. The framework includes the exploitation and exploration techniques that introduce and test various CNN structures during the optimization process.

The limitation of this study is the consideration of only one epoch during the algorithm run. It would affect the robustness of the obtained solutions and needs to be resolved, e.g., by increasing the number of epochs. Even though this study proposed a novel way to extract important information about the layers from various studies, another drawback of this study is the limited number of considered studies. Also, the search should be improved to allow increasing the accuracy of the final structure. More layer types could be considered as well to allow proposing CNN models with better accuracy.

Acknowledgement
This research was funded by University of Surabaya with the grant number 022/ST-Lit/LPPM-01/FT/IV/2023.

References
AI Summer, https://theaisummer.com/cnn-architectures/, Accessed on March 16, 2023.

Amirghasemi, M., O'Sullivan, M., and Walker, C., An effective self-adaptive bi-directional multi-layer genetic algorithm for resource and user allocations in networks, *Expert Systems with Applications*, vol. 228, pp. 120283, 2023.

Batchuluun, G., Yoon, H. S., Kang, J. K., and Park, K. R., Gait-Based Human Identification by Combining Shallow Convolutional Neural Network-Stacked Long Short-Term Memory and Deep Convolutional Neural Network, *IEEE Access*, vol. 6, pp. 63164–86, 2018.

Capua, M. D., Ciaramella, A., and De Prisco, A., Machine Learning and Computer Vision for the automation of processes in advanced logistics: the Integrated Logistic Platform (ILP) 4.0, *Procedia Computer Science*, vol. 217, pp. 326–38, 2023.

Crivellari, A., Beinat, E., Caetano, S., Seydoux, A., and Cardoso, T., Multi-target CNN-LSTM regressor for predicting urban distribution of short-term food delivery demand, *Journal of Business Research*, vol. 144, pp. 844–53, 2022.

Guo, J., Wang, W., Tang, Y., Zhang, Y., and Zhuge, H., A CNN-Bi_LSTM parallel network approach for train travel time prediction, *Knowledge-Based Systems*, vol. 256, pp. 109796, 2022.

Ijjina, E. P. and Chalavadi, K. M., Human action recognition using genetic algorithms and convolutional neural networks, *Pattern Recognition*, vol. 59, pp. 199–212, 2016.

Johnson, F., Valderrama, A., Valle, C., Crawford, B., Soto, R., and Nanculef, R., Automating Configuration of Convolutional Neural Network Hyperparameters Using Genetic Algorithm, *IEEE Access*, vol. 8, pp. 156139–52, 2020.

Kabir Anaraki, A., Ayati, M., and Kazemi, F., Magnetic resonance imaging-based brain tumor grades classification and grading via convolutional neural networks and genetic algorithms, *Biocybernetics and Biomedical Engineering*, vol. 39, no. 1, pp. 63–74, 2019.

Liu, R. W., Yuan, W., Chen, X., and Lu, Y., An enhanced CNN-enabled learning method for promoting ship detection in maritime surveillance system, *Ocean Engineering*, vol. 235, pp. 109435, 2021.

Medium, https://towardsdatascience.com/top-10-cnn-architectures-every-machine-learning-engineer-should-know-68e2b0e07201, Accessed on March 16, 2023.

MetaAI, https://paperswithcode.com/sota/image-classification-on-mnist, Accessed on March 16, 2023.

Peiris, A., Hui, F. K. P., Duffield, C., and Ngo, T., Production scheduling in modular construction: Metaheuristics and future directions, *Automation in Construction*, vol. 150, pp. 104851, 2023.

Ren, S., Choi, T.-M., Lee, K.-M., and Lin, L., Intelligent service capacity allocation for cross-border-E-commerce related third-party-forwarding logistics operations: A deep learning approach, *Transportation Research Part E: Logistics and Transportation Review*, vol. 134, pp. 101834, 2020.

Rikhtegar, A., Pooyan, M., and Manzuri-Shalmani, M. T., Genetic algorithm-optimised structure of convolutional neural network for face recognition applications, *IET Computer Vision*, vol. 10, no. 6, pp. 559–66, 2016.

Schlüter, M., Lickert, H., Schweitzer, K., Bilge, P., Briese, C., Dietrich, F., and Krüger, J., AI-enhanced Identification, Inspection and Sorting for Reverse Logistics in Remanufacturing, *Procedia CIRP*, vol. 98, pp. 300–305, 2021.

Taye, M. M., Theoretical Understanding of Convolutional Neural Network: Concepts, Architectures, Applications, Future Directions, *Computation*, vol. 11, no. 3, pp. 52, 2023.

Toaza, B. and Esztergár-Kiss, D., A review of metaheuristic algorithms for solving TSP-based scheduling optimization problems Image 1, *Applied Soft Computing*, vol. 148, pp. 110908, 2023.

Topbots, https://www.topbots.com/important-cnn-architectures/, Accessed on March 16, 2023.

Tsoumalis, G. I., Bampos, Z. N., Chatzis, G. V., Biskas, P. N., and Keranidis, S. D., Minimization of natural gas consumption of domestic boilers with convolutional, long-short term memory neural networks and genetic algorithm, *Applied Energy*, vol. 299, pp. 117256, 2021.

Vetrimani, E., Arulselvi, M., and Ramesh, G., Building convolutional neural network parameters using genetic algorithm for the croup cough classification problem, *Measurement: Sensors*, vol. 27, pp. 100717, 2023.

Wang, Y.-H. and Su, W.-H., Convolutional Neural Networks in Computer Vision for Grain Crop Phenotyping: A Review, *Agronomy*, vol. 12, no. 11, pp. 2659, 2022.

Wen, X., Qian, Y., Lian, X., Zhang, Y., Wang, H., and Li, H., Improved genetic algorithm based on multi-layer encoding approach for integrated process planning and scheduling problem, *Robotics and Computer-Integrated Manufacturing*, vol. 84, pp. 102593, 2023.

Wu, W., Shen, L., Zhao, Z., Harish, A. R., Zhong, R. Y., and Huang, G. Q., Internet of Everything and Digital Twin enabled Service Platform for Cold Chain Logistics, *Journal of Industrial Information Integration*, vol. 33, pp. 100443, 2023.

Biographies

Ivan Kristianto Singgih is an Assistant Professor in the Department of Industrial Engineering at University of Surabaya (Ubaya), Indonesia. He received his B.S. and M.S. degrees in Industrial Engineering from Bandung Institute of Technology, Indonesia, and Ph.D. degree in Industrial Engineering from Pusan National University, Busan, Korea, in 2009, 2010, and 2017, respectively. He was a Postdoctoral Researcher and Research Assistant Professor under the Department of Industrial and Management Engineering at Pohang University of Science and Technology (POSTECH), and a Postdoctoral Researcher at Korea Advanced Institute of Science and Technology (KAIST) and Korea University. His research interests include operations research, smart systems, machine learning, and simulation. He is highly interested in collaborations between academia, practitioners, government, etc. He received the Best Student Paper Award in IJIE 2013 Conference and Korean Government Scholarship Excellent Academic Achievement Award in 2014. He is an active committee member of Institute of Supply Chain and Logistics Indonesia (ISLI) and The Association of Indonesian Researchers in South Korea (APIK).

Evan Sutrisno is an undergraduate student at the Department of Industrial Engineering, Faculty of Engineering, University of Surabaya in Indonesia. His research endeavors primarily concentrate on scheduling problems and the advancement of diverse optimization methods, encompassing the formulation of mathematical models and the design of algorithms. His commitment to exploring these areas not only showcases a profound understanding of industrial engineering concepts but also underscores his passion for addressing real-world challenges through systematic problem-solving approaches.
Currently pursuing a Bachelor's degree in Industrial Engineering at the University of Surabaya, Evan is deeply intrigued by the field of Operations Research. Actively engaged in various research projects alongside esteemed faculty members, he demonstrates an unwavering dedication to expanding his knowledge and contributing to the academic community. Evan's research pursuits extend to scheduling problems, where he employs a combination of mathematical models and innovative algorithms to tackle complex issues in the field.
Looking ahead, Evan aspires to pursue a Master's degree overseas, driven by a desire to deepen his expertise and broaden his perspective in industrial engineering. His academic journey is characterized by his rigorous research involvement. Evan has shared his knowledge by teaching Data Science courses, specializing in Convolutional Neural Network (CNN) applications for image processing. This dual commitment to research and teaching showcases Evan Sutrisno as a dynamic individual poised to make meaningful contributions to the intersection of academia and industry.

Michael Abednego is an undergraduate student at the Department of Industrial Engineering, Faculty of Engineering, University of Surabaya in Indonesia. His researches focus on scheduling problems and development of various optimization methods, including mathematical model and algorithms.

Dina Natalia Prayogo is a lecturer of Department of Industrial Engineering, Faculty of Engineering, University of Surabaya, Indonesia. She earned B.Eng. in Physics Engineering from Institute of Technology Sepuluh November (ITS), Surabaya, Indonesia. Master’s in Industrial and Systems Engineering from The National University of Singapore, and Doctor in Industrial Engineering from University of Indonesia. She has published several journal and conference papers. Her research interests include supply chain management, maritime logistics, and operations research.