



Application of Convolutional Neural Network (CNN) Method in Fluctuations Pattern

Melinda Melinda^{1*}, Yunidar Yunidar¹, Nur Afny Catur Andryani²

¹Department of Electrical Engineering and Computer, Engineering Faculty, Universitas Syiah Kuala, Banda Aceh, Indonesia

²Computer Science Program, School of Computer Science, Bina Nusantara University, Jakarta, Indonesia

*Correspondence: melinda@usk.ac.id

SUBMITTED: 23 June 2023; REVISED: 21 July 2023; ACCEPTED: 24 July 2023

ABSTRACT: In the acquisition of amplitude data, the inaccuracy of a signal with the occurrence of an unstable peak value of the amplitude in the data is called a fluctuation. This study uses High-High Fluctuation (HHF) signal data from the acquisition of Multi-Spectral Capacitive Sensors (MSCS) with Hydrogen Dioxide (H₂O) and Hydrogen Dioxide (H₂O) objects mixed with Sodium Hydroxide (NaOH) that have been organized into a matrix form. The data acquisition results in previous studies have several parts that are difficult to distinguish with the naked eye. The method used in this study applies the CNN method for image recognition of signal fluctuations of type HHF from H₂O and H₂O mixed with NaOH, using the architecture known as AlexNet. Then, the H₂O and H₂O data groups mixed with NaOH were grouped into training data groups of 280 image data for each data type, and 70 image data for test data for both groups. During the training stage, the number of epochs used is 20. However, by the time the number of epochs reaches 15, the accuracy rate is already high, reaching 98%. Furthermore, at the testing stage, the CNN program can correctly recognize the entire 70 image data for both materials, achieving perfect recognition for the total amount of the two materials.

KEYWORDS: Fluctuation pattern; High-high fluctuation; convolutional neural; network

1. Introduction

In the process of computer image processing, there is often a condition of signal inaccuracy and a state of rise and fall of the peak amplitude value during data acquisition; this is referred to as fluctuation. The acquisition process causes the acquired data to mix with noise and produce fluctuations, making these fluctuations very complex [1]. Generally, the acquisition process uses a large amount of data, so recognizing the information in the data requires a lot of work. Therefore, an analysis is needed to process signal fluctuations and produce fluctuation results from the data [2-4].

In recent years, fluctuation signal processing methods have been widely developed. Processing signal fluctuations is quite difficult to achieve satisfactory results using ordinary methods. The analysis method for processing multi-spectral fluctuation signals in previous studies involves using Wavelet and Fourier transforms [1]. This research shows that the

information content is important to reflect the state of a material, and the material has a non-stationary nature. Therefore, fluctuation is an inevitable event [1, 4].

This study uses data obtained from previous research, namely pure water (H_2O) and pure water materials mixed with alkaline solutions (H_2O mixed with $NaOH$); the data will be represented not in the form of images but in the form of 2D matrices with pixel size [650 x 420] [5]. The fluctuation pattern of the material is generated from the acquisition process of a multi-spectral capacitive sensor (MSCS) [6]. Then, three different fluctuation patterns are obtained, namely Mean Fluctuation (MF), High Fluctuation (HF), and High-High Fluctuation (HHF) [7, 8]. This study focuses on the HHF fluctuation pattern because it produces the most contrasting pattern based on the signal amplitude.

The application of the CNN method contributes to obtaining a method of processing fluctuation pattern data by classifying H_2O fluctuation pattern matrix data and H_2O mixed with $NaOH$, which has been converted into image form. In the CNN method, the main purpose of convolution and subsampling operations is to extract features from raw input data. The convolution operation involves the multiplication of a small kernel matrix and a certain area of the two-dimensional matrix input [9]. The reason for choosing the CNN method is that there has been no previous research proposing this method to observe the similarity of the fluctuation pattern matrix data; the classification is not so clear between H_2O and H_2O mixed with $NaOH$ because the fluctuation patterns of the two materials cannot be distinguished by the naked eye or at first glance, as they are almost the same. Therefore, using the CNN method is expected to enable the computer to distinguish the two fluctuation patterns. Previous research using one convolution layer resulted in a higher accuracy value (81.5%) compared to using 2 convolution layers (70.7%) [10]. Additionally, previous research using 7 convolution layers for face recognition resulted in an accuracy rate of 99.6% [11].

Each process in the CNN method involves several layers, such as a convolution layer, a pooling layer to increase the position variance of a feature, a fully connected layer to connect the neurons before and after, and a dropout layer to speed up the learning process. These steps allow the algorithm to recognize the characteristics of the material based on its fluctuation pattern with a high level of accuracy [12–15]. Determining the CNN architecture and the number of epochs affect accuracy. In previous research, the MF, HF, and HHF data patterns from H_2O and H_2O mixed with $NaOH$ were challenging to identify directly, so using the CNN method can facilitate the identification process of the two materials. The benefit of this research in the future was that it could recognize signal fluctuation patterns by processing signals using image processing on a material with a CNN method that utilized spectral-based material.

2. Materials and Methods

The problem-solving approach in this study is to apply the stages as shown in Figure 1, as follows: fluctuation pattern data preparation, modeling (preprocessing and classification), evaluation and measurement of system performance (accuracy, precision, recall, F-score).

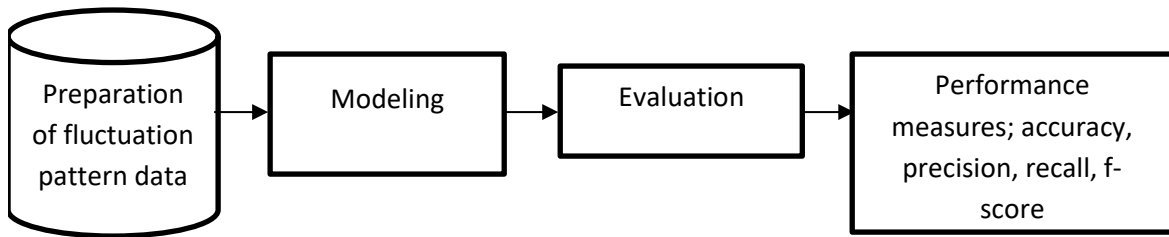


Figure 1. Research stage.

2.1. Materials.

This research takes the data used in previous studies, chemical data from fluctuations using MSCS [1–4]. In this study, the matrix data type of the HHF fluctuation pattern image totaling 350 data was used; for H₂O material (data A), 175 images were used, and for H₂O mixed with NaOH material (data B), 175 images are used as shown in Figure 2, which is a sample of the two materials. This study uses a computer device with specifications sufficient to run the classification process (Table 1).

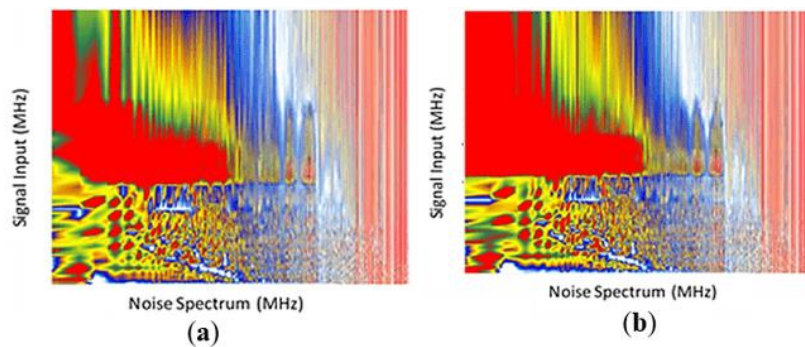


Figure 2. Image display of HHF fluctuation pattern (a) H₂O material (b) H₂O material mixed with NaOH

Table 1. Equipment used.

No	Equipment	Specifications	Justifications
1.	Laptop	Intel(R) Core(TM) i5-4210U CPU @ 1.70GHz 2.40 GHz, RAM 8 GB, Graphic card NVIDIA GEFORCE 920M	Run the CNN program
2.	Matlab @2018	-	Run the program
3.	Python 3.6	Matplotlib, Numpy, Seaborn, Pickle, Scikit-learn, Tensorflow	Pre-processing, Running the CNN proses CNN, and do the testing stage
4.	Visual Studio Code	-	As a code editor

2.2. Preprocessing.

2.2.1. Cropping.

In image processing, a cropping process was needed to get certain parts needed to facilitate the next process. If the cropping stage was not done, then the computer analyzed all objects in the image [16].

2.2.2. Augmentation.

The purpose of augmentation was to enable the system to recognize the basic variances in the object image that would be processed in the subsequent stage. Additionally, there exist

translational data and reflection data, which form integral parts of the augmentation process. Translational data represents the original data shifted in a straight line, while reflection data comprises reflections or mirror images of the original data [17].

2.3. Convolutional Neural Network (CNN).

The Convolutional Neural Network (CNN) constitutes a feature extraction and classification technique. Yann Lecun was the pioneer who introduced CNN in 1988. CNN stands as one of the methods that have paved the way for the development of Deep Learning. In terms of image classification, CNN processes an input image prior to classifying it into a specific category [18]. CNN is a development of Multilayer Perceptron (MLP) intended to process two-dimensional data [13]. Convolutional neural networks aim to process data that possess a network shape, such as topology. The CNN stages can be observed in Figure 3.

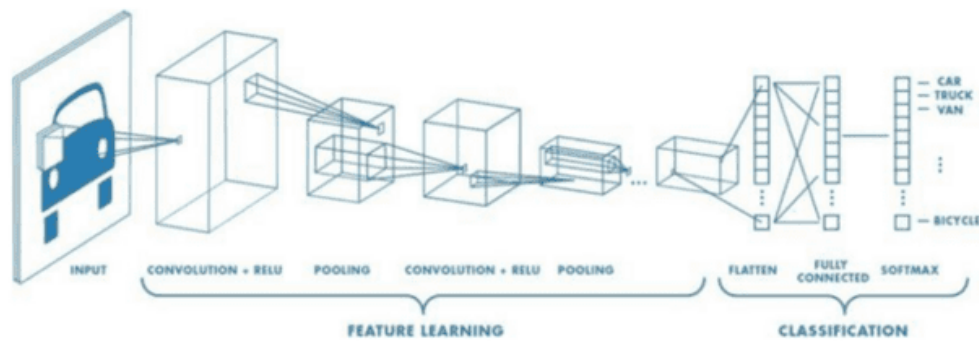


Figure 3. CNN architecture [9].

2.3.1. Convolution layer.

In this layer, the filter size is considered the most significant factor in CNN, each layer has a different filter size. In image processing, convolution can be defined as applying a kernel to an image at all possible offsets. The weights in the layer specify the convolution kernel used so that the kernel can be trained based on the input given to the CNN [19].

2.3.2. Fully connected layer.

The fully connected layer connects the neurons generated in the previous layer and the neurons generated in the next layer [20]. Fully connected layers can improve the performance of CNN models and can reduce the time needed to choose between different architectures, such as deeper and shallower layers.

2.3.3. Pooling layer.

The pooling layer serves to reduce or reduce the number of parameters by using down-sampling operations to improve the positional invariance of a feature [13]. Based on Figure 4, the feature map obtains convolution results using a type of max pooling operation with a smaller size, namely a 2x2 matrix size [20].

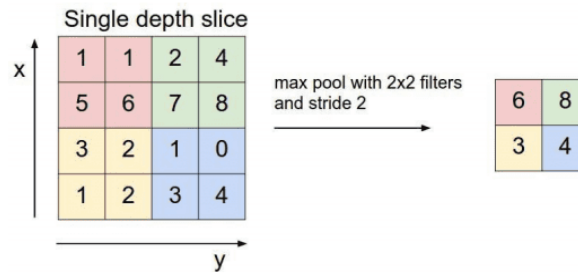


Figure 4. Max pooling [20].

2.3.4. Dropout.

The purpose of augmentation is so that the system can recognize the basic variance in the object image that will be processed to the next stage. In addition, there are translational data and reflection data, which are part of the augmentation process (Figure 5). Translational data is the original data shifted in a straight line, while reflection data is a reflection or reflection of the original data [17].

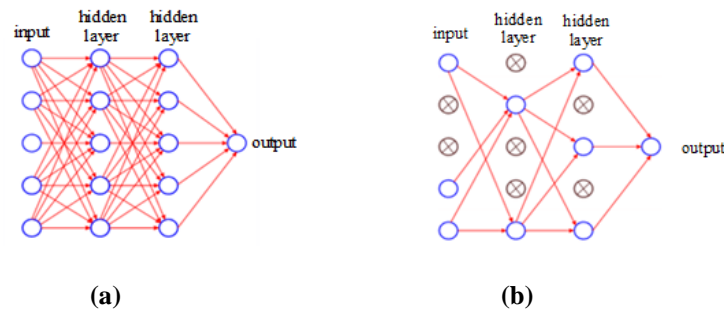


Figure 5. (a) Before drop-out (b) After drop-out [22].

2.4. Accuration.

By considering recall and specificity simultaneously [23]. The results are analyzed by calculating the accuracy, recall, precision, and F-score values from Equation (1) to Equation (4) [24].

$$Accuracy = \frac{(TP+TN)}{(TP+TN+FP+FN)} \quad (1)$$

$$Recall = \frac{TP}{(TP+FN)} \quad (2)$$

$$Precision = \frac{TP}{(TP+FP)} \quad (3)$$

$$F - Score = 2 \frac{(Recall \times Precision)}{(Recall+Precision)} \quad (4)$$

Where:

- TP : True positive
- TN : True Negative
- FP : False Positive
- FN : False Negative

3. Results and Discussion

3.1. Program Implementation.

This section describes the application of the CNN method before and after the training and testing processes on data sets A and B, as shown in Figure 2. The data obtained from previous research was converted from the .bat extension to .txt and entered into Excel with a matrix of 31 x 8192. Then, a program simulation was carried out using MATLAB to produce fluctuation graphs of data groups A and B as images. For the training and testing processes, the program simulation used PYTHON for both groups of materials. If the accuracy level in the training process was not as expected, the number of epochs was increased, and if the accuracy level was as expected, then the testing stage proceeded.

Based on Figure 6, the initial stage of this study was to obtain images of the signal fluctuation pattern using Matlab programming. Furthermore, the images underwent a pre-processing stage using Python programming. After experiencing the pre-processing stage, the A and B data that had been grouped underwent the training stage using Python programming. Furthermore, the last stage, the testing stage in this study, also used Python programming to support the implementation of the expected program. It required software and hardware specifications that could run the program.

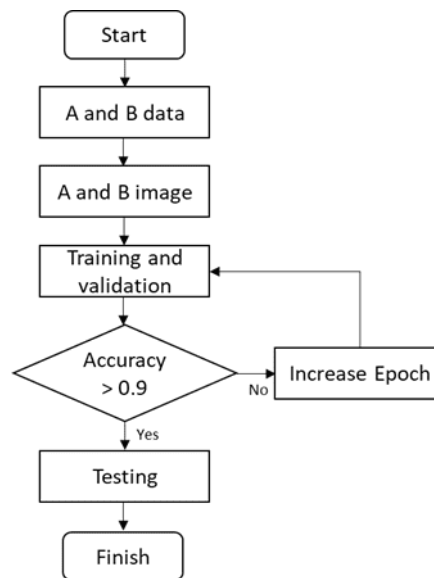
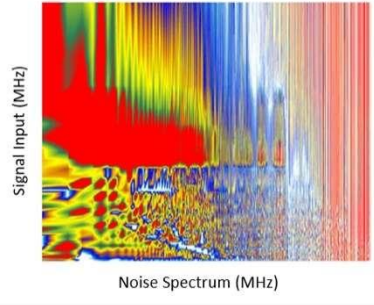
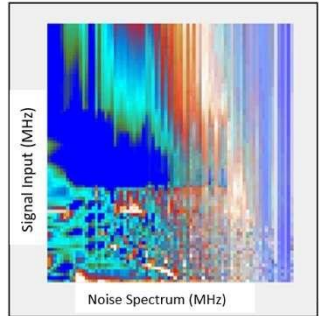
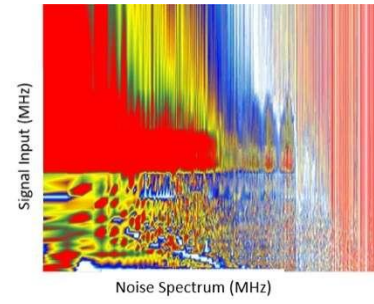
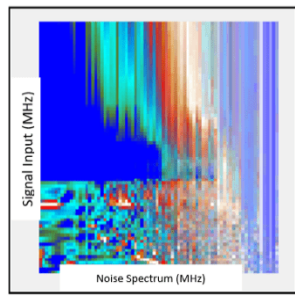
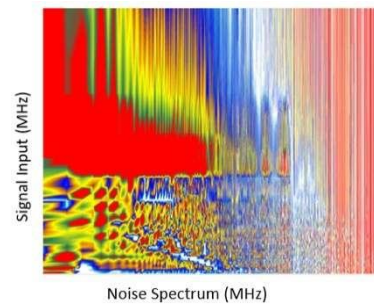
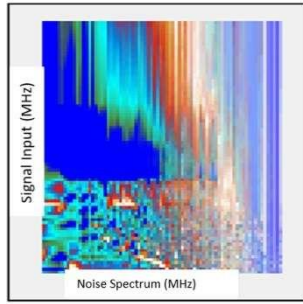
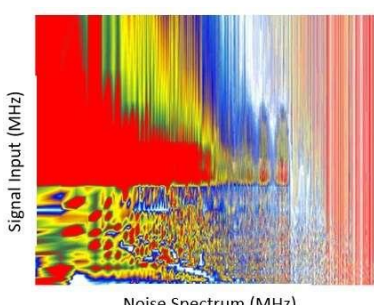
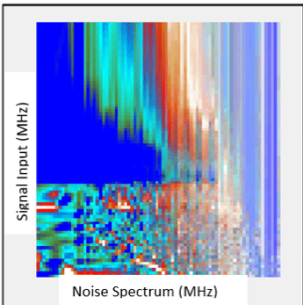


Figure 6. Flowchart of CNN Stage.

3.2. Pre-processing results of H_2O and H_2O material data mixed with $NaOH$.

In the pre-processing section, the results of image changes on both research materials, namely data A and data B, were obtained by changing color, scale, and cropping using the PYTHON programming language installed on the Windows 10 operating system. The input of data A and data B was an image with a pixel size of 560 x 420 per unit image before undergoing pre-processing. In this pre-processing stage, the original image went through a down-sampling process so that the image changed to a smaller pixel size into a matrix with dimensions of 100 x 100. Then, the matrix was plotted into an image using the matplotlib library.

Table 2. Pre-processing results of H_2O and H_2O material data mixed with $NaOH$

Sequence	Data	Before Pre-Processing	After Pre-processing
141	A		
	B		
142	A		
	B		

This process was done so that the image could be easily recognized in the subsequent training and testing stages. The results before and after the pre-processing stage can be seen in Table 2, which consists of data images A and B in the order of 141 and 142. It can be observed that the image, after pre-processing, had its pixel size reduced, and the color change occurred due to the use of the matplotlib library in the Python program. Among all the images obtained through the pre-processing stage, there was no significant difference when viewed directly.

3.3. Architecture determination result.

The CNN architecture used in this study is a modified AlexNet, for more details can be seen in Table 3.

Table 3. Parameter used.

Layer (type)	Output Shape	Activation	Parameter
Conv2D	(None, 98, 98, 32)	Re-Lu.	896
MaxPooling2D	(None, 49, 49, 32)	Re-Lu.	0
Conv2D	(None, 47, 47, 32)	Re-Lu.	9248
MaxPooling2D	(None, 23, 23, 32)	Re-Lu.	0
Dropout(0.25)	(None, 23, 23, 32)	-	0
Flatten	(None, 16928)	-	0
Dense	(None, 128)	Re-Lu.	2166912
Dropout(0.5)	(None, 128)	-	0
Dense	(None, 2)	softmax	258
Total Parameter			2.177.314

Based on Table 3, it is evident that the data underwent two convolutions using the Re-Lu activation function. Additionally, the max-pooling technique was utilized to determine the value of the new matrix for processing in the subsequent stage. Afterward, the data was processed in the fully connected layer. The flattening technique was applied to convert the multi-dimensional matrix measuring $23 \times 23 \times 32$ into a one-dimensional matrix measuring 16,928. Subsequently, the 16,928 matrices were transformed into a size of 128 using the Re-Lu activation function. Further, the 128-sized matrix was converted into a 2-sized matrix using the softmax activation function. In this system, the dropout technique was also employed to randomly select and eliminate neurons, thereby preventing overfitting and expediting the learning process. Throughout the learning process, the total parameters or weights amounted to 2,177,314.

3.2. Training results and dataset validation.

After the pre-processing stage, data A and B were subjected to the training and validation stage. At this stage, the two materials were divided into two groups: test data and training data. The training data was further divided into two groups, namely the training group and the validation data group, each containing 224 images from each material, resulting in a total of 112 images. The validation group consisted of 56 images from each material, with a total of 28 images for each material. During the training and validation of datasets from data materials A and B, the relationship between epochs and the accuracy and loss of the CNN model was analyzed, as depicted in Figure 7. In Figure 7(a), the training accuracy graph (blue) showed that at 14 epochs, the accuracy level during the training process was very high, reaching an accuracy value close to 100%. Meanwhile, the accuracy validation graph (red) indicated that it took 13 epochs to achieve an accuracy value of 98%. Based on the obtained graphs, it was evident that the higher the number of epochs used during the training and validation of the CNN model, the higher the accuracy level. Similarly, in Figure 7(b), the training loss graph (blue) demonstrated that at 15 epochs, the loss rate during the training process was remarkably low, with a loss value below 10%. On the other hand, the loss validation graph (red) revealed that it took 15 epochs to reach a loss value below 10%. Based on these graphs, it was evident that the more

epochs used during the training and validation of the CNN model, the lower the resulting error rate would be. The validation results of data A and B are shown in Figure 8.

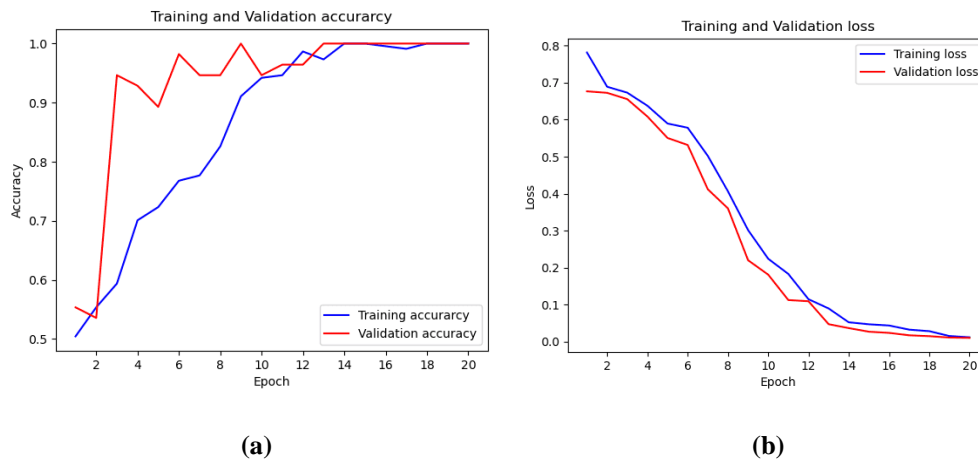


Figure 7. Epoch relationship with accuracy (a) and Epoch relationship with loss (b).

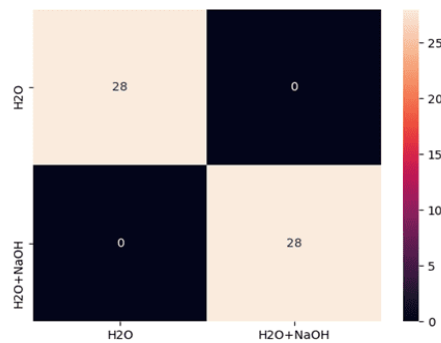


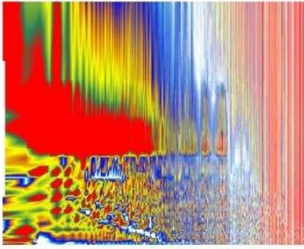
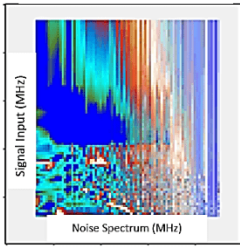
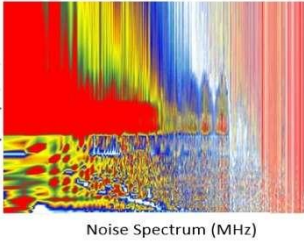
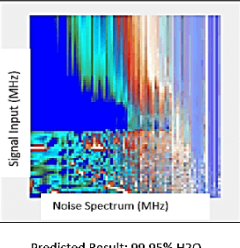
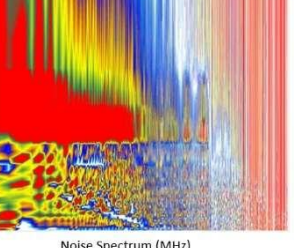
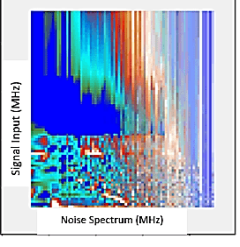
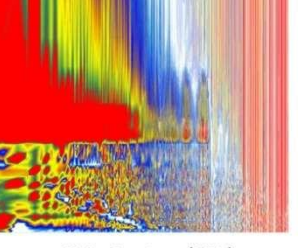
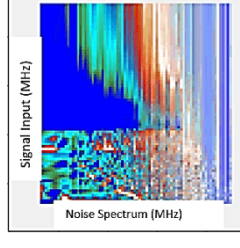
Figure 8. Validation results of H₂O and H₂O mixed with NaOH materials.

Based on Figure 8, the validation results of the H₂O and H₂O mixed with NaOH materials indicated that the image was divided into 4 parts, and then further divided into two groups of parts. The black part represents the portion of error or loss at the validation stage for data A and B. It can be observed from the image that neither of the two materials experienced errors in recognition during the validation stage. The soft pink group represents the data group that correctly recognized the two materials at the validation stage. From the image, it can be seen that a total of 28 images for data A and all 28 images for data B were correctly recognized during the validation stage. The recognition rate of the two materials was correct at the validation stage, with no errors in their recognition.

3.5. Testing result.

During the testing stage, a total of 70 image data were used, with each material having 35 image samples. The testing process resulted in an accuracy rate of 100% for both materials. Each of the 35 test images for data A and B was correctly recognized. Table 4 displays the test results for data A and B from 2 sample images used for each data. It can be observed that both data A and B had a total of 35 test images each, and both achieved an accuracy rate of 100%.

Table 4. Testing results of H₂O and H₂O mixed with NaOH material data.

Sequence	Data	Input Image	Test Results
141	A		Predicted Result: 99.85% H ₂ O 
	B		Predicted Result: 99.97% H ₂ O+NaOH 
142	A		Predicted Result: 99.95% H ₂ O 
	B		Prediction Result: 99.98% H ₂ O+NaOH 

Based on Table 4, the test results of H₂O and H₂O mixed with NaOH demonstrated that the image was divided into 4 parts and further categorized into two groups of parts. The black part represents the portion of error or loss at the testing stage for data A and B. It can be observed from the image that neither of the two materials experienced errors in recognition during the testing phase. The soft pink group represents the data group that correctly recognized the two materials during the testing stage. It can be seen from the image that both data A and B had all 35 images correctly recognized during the testing stage. At this stage, the recognition rate of the two materials was correct, with no errors in their recognition.

Based on Figure 9, accuracy, recall, precision, and F-Score were calculated for data A and B using Equations 1 to 4. By applying 7 convolution layers from CNN, as in previous research [11], this study obtained test results with higher accuracy in detecting fluctuation patterns, as shown in Table 5. The program successfully recognized data A and B correctly as a whole, as evidenced by the accuracy, recall, precision, and F-Score values, all reaching 100%.

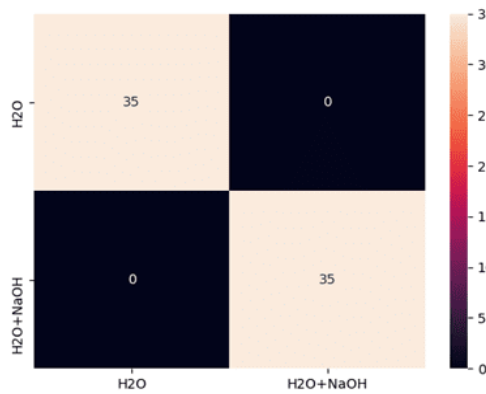


Figure 9. Test results of H₂O and H₂O mixed with NaOH.

Table 5. Hasil perhitungan accuracy, recall, precision dan F-Score.

No	Evaluation Parameter CNN	Formula	Value
1	Accuracy	$\frac{TP + TN}{TP + TN + FP + FN}$	$\frac{35 + 35}{35 + 35 + 0 + 0}$
2	Recall	$\frac{TP}{TP + FN}$	$\frac{35}{35 + 0}$
3	Precision	$\frac{TP}{TP + FP}$	$\frac{35}{35 + 0}$
4	F-Score	$2 \times \frac{Recall \times Precision}{Recall + Precision}$	$2 \times \frac{100\% \times 100\%}{100\% + 100\%}$

4. Conclusions

The image of the signal fluctuation pattern of H₂O and H₂O mixed with NaOH could be recognized using the CNN architecture developed in this study, achieving an accuracy rate of 100%. At first glance, distinguishing the fluctuation pattern images of the two materials with the naked eye seemed challenging, making this CNN method even more significant in demonstrating its ability to recognize the images accurately. The research yielded CNN evaluation parameters, including accuracy, precision, recall, and F-Score, all reaching 100%. These exceptional results were obtained because both H₂O and H₂O mixed with NaOH test data were recognized correctly. The model was trained for 20 epochs; however, the level of accuracy significantly increased after just 15 epochs. For future work, this study could expand the application of the CNN method to identify other materials, such as H₂O mixed with HCl or images of fluctuation patterns of various other materials. Additionally, the quality of materials could be assessed using the Peak Signal to Noise Ratio (PSNR) as a measure. This extension would contribute to enhancing the potential applications and usefulness of the CNN-based approach in material identification and quality evaluation.

Acknowledgment

We would like to acknowledge to the universitas Syiah Kuala that has supported our study.

References

- [1] Syahrial, S.; Melinda, M.; Irhamsyah, M.; Sheiba, T. (2022). Performance of Elliptic Filter on Multi-Spectral Signal Based on Divided Zone. Proceedings of the International Conference on Electrical Engineering and Informatics, pp. 178–183. <http://doi.org/10.1109/ICELTICs56128.2022.9932079>.

- [2] Melinda, M.; Tamsir, A. S.; Bastian, A.; Gunawan, D. (2018). Comparison Analysis of Data Groupings of Fluctuation Patterns based on the Amplitude Representation Value (ARV). Proceedings of the 2nd 2018 International Conference on Electrical Engineering and Informatics, ICELTICS 2018, pp. 25–29. <http://doi.org/10.1109/ICELTICS.2018.8548853>.
- [3] Melinda, M.; Sianturi, P.; Tamsir, A. S. (2019). Comparative Analysis of Material Fluctuation Response based on Data Set Groups. *IOP Conference Series: Materials Science and Engineering*, 620, 012092. <http://doi.org/10.1088/1757-899X/620/1/012092>.
- [4] Irhamsyah, M.; Melinda, M.; Yunidar, Y.; Syahril, S. (2022). Parameters Quality Performance of Signal Fluctuation Based on Data Grouping. Proceedings of the 2022 International Conference on Electrical Engineering, Computer, and Information Technology, ICEECIT 2022, pp. 226–231. <http://doi.org/10.1109/ICEECIT55908.2022.10030593>.
- [5] Liu, C.; Zhou, X.; Zhou, Y.; Akbar, A. (2020). Multi-temporal monitoring of urban river water quality using UAV-borne multi-spectral remote sensing. *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences - ISPRS Archives*, 43, pp. 1469–1475. <http://doi.org/10.5194/isprs-archives-XLIII-B3-2020-1469-2020>.
- [6] Melinda, M.; Tanjung, A.; Tamsir, A. S.; Basari, B.; Gunawan, D. (2017). Grouped data analysis of H₂O and H₂O mixed with NaOH on multi-spectral high fluctuation pattern. Proceedings of the 2017 International Conference on Electrical Engineering, Informatics, Advanced Knowledge, Research, Technology, and Humanities - ICELTICS 2017, pp. 184–188. <http://doi.org/10.1109/ICELTICS.2017.8253266>.
- [7] Tamsir, A. S.; Melinda, M.; Hariadi, M.; Basari, B.; Gunawan, D. (2017). Self-comparison performance analysis of H₂O on multi-spectral fluctuation pattern. Proceedings of the 2017 International Conference on Electrical Engineering, Informatics, Advanced Knowledge, Research, Technology, and Humanities - ICELTICS 2017, pp. 263–268. <http://doi.org/10.1109/ICELTICS.2017.8253273>.
- [8] Melinda, M.; Tamsir, A.S.; Afifah, S.; Gunawan, D. (2018). Impedance Influence Analysis on Multi Spectral Capacitive Sensor. Proceedings of the 2nd 2018 International Conference on Electrical Engineering and Informatics - ICELTICS 2018, pp. 19–24. <http://doi.org/10.1109/ICELTICS.2018.8548794>.
- [9] Dewa, C.K.; Fadhillah, A.L.; Afiahayati, A. (2018). Convolutional Neural Networks for Handwritten Javanese Character Recognition. *Indonesian Journal of Computer and Cybernetics Systems*, 12, 83–94. <http://doi.org/10.22146/ijccs.31144>.
- [10] Dzulqarnain, M.F.; Suprpto, S.; Makhrus, F. (2019). Improvement of Convolutional Neural Network Accuracy on Salak Classification Based Quality on Digital Image. *Indonesian Journal of Computer and Cybernetics Systems*, 13, 189–198. <http://doi.org/10.22146/ijccs.42036>.
- [11] Chauhan, R.; Ghanshala, K.K.; Joshi, R.C. (2018). Convolutional Neural Network (CNN) for Image Detection and Recognition. 1st International Conference on Security, Cybernetics, and Communications - ICSCCC 2018, pp. 278–282. <http://doi.org/10.1109/ICSCCC.2018.8703316>.
- [12] Hassan, R.J.; Abdulazeez, A.M. (2021). Deep Learning Convolutional Neural Network for Face Recognition: A Review. *International Journal of Science and Business*, 5, 41–55. <http://doi.org/10.5281/zenodo.4471013>.
- [13] Wahyuni, E. S.; Hendri, M. (2019). Smoke and Fire Detection Base on Convolutional Neural Network. *ELKOMIKA Journal of Electrical Engineering, Energy, Telecommunication, Electronics*, 7, 455–465. <http://doi.org/10.26760/elkomika.v7i3.455>.
- [14] Dzierżak, R.; Omiotek, Z. (2022). Application of Deep Convolutional Neural Networks in the Diagnosis of Osteoporosis. *Sensors*, 22, 8189. <http://doi.org/10.3390/s22218189>.
- [15] Wang, Q.; Yu, Z.; Zhang, Y. (2021). Face recognition Using Convolutional Neural Networks. 2nd International Conference on Machine Learning and Computational Applications - ICMLCA 2021, pp. 144–148.

- [16] Mishra, B.K.; Thakker, D.; Mazumdar, S.; Neagu, D.; Gheorghe, M.; Simpson, S. (2020). A novel application of deep learning with image cropping: a smart city use case for flood monitoring. *Journal of Reliability and Intelligent Environments*, 6, 51–61. <http://doi.org/10.1007/s40860-020-00099-x>.
- [17] Shorten, C.; Khoshgoftaar, T.M. (2019). A survey on Image Data Augmentation for Deep Learning. *Journal of Big Data*, 6, 60. <http://doi.org/10.1186/s40537-019-0197-0>.
- [18] Hadiyoso, S.; Aulia, S. (2023). Automatic Leukocytes Classification using Deep Convolutional Neural Network, *ELKOMIKA Journal of Electrical Engineering, Energy, Telecommunication, Electronics*, 11, 195–206. <https://doi.org/10.26760/elkomika.v11i1.195>.
- [19] Mukherjee, S. (2019). Impact of Fully Connected Layers on Performance of Convolutional Neural Networks for Image Classification. *Neurocomputing*, 378, 112–119. <http://doi.org/10.1016/j.neucom.2019.10.008>.
- [20] Lv, Q.; Zhang, S.; Wang, Y. (2022). Deep Learning Model of Image Classification Using Machine Learning. *Advances in Multimedia*, 2022, 3351256. <http://doi.org/10.1155/2022/3351256>.
- [21] Zhang, Y.; Pan, C.; Sun, J.; Tang, C. (2018). Multiple sclerosis identification by convolutional neural network with dropout and parametric ReLU. *Journal of Computer Science*, 28, 1–10. <http://doi.org/10.1016/j.jocs.2018.07.003>.
- [22] Poernomo, A.; Kang, D. (2018). Biased Dropout and Crossmap Dropout: Learning towards effective Dropout regularization in convolutional neural network. *Neural Networks*, 104, 60–67. <http://doi.org/10.1016/j.neunet.2018.03.016>.
- [23] He, C.; Ma, M.; Wang, P. (2020). Extract interpretability-accuracy balanced rules from artificial neural networks: A review. *Neurocomputing*, 387, 346–358. <http://doi.org/10.1016/j.neucom.2020.01.036>.
- [24] Roslidar, R.; Saddami, K.; Arnia, F.; Syukri, M.; Munadi, K. (2019). A study of fine-tuning CNN models based on thermal imaging for breast cancer classification. 2019 IEEE International Conference on Cybernetics and Computational Intelligence (CyberneticsCom), pp. 77–81. <http://doi.org/10.1109/CYBERNETICSCOM.2019.8875661>.



© 2023 by the authors. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).