Indonesian Journal of Electronics, Electromedical Engineering, and Medical Informatics e-ISSN: 2656-8624
Homepage: https://ijeeemi.org/; Vol. 7, No. 3, pp. 475-487, August 2025

RESEARCH PAPER **OPEN ACCES**

Convolutional Kolmogorov-Arnold Network for Pneumonia Detection in Medical Image Analysis

Riechie[1], Vira Jessica[1], Matthew Kurniawan[2], and Feliks Victor Parningotan Samosir[2]

[1] Mathematics Department, Faculty of Science and Technology (FaST), Universitas Pelita Harapan, Tangerang, Indonesia
[2] Informatics Department, Faculty of Information Technology (FIT), Universitas Pelita Harapan, Tangerang, Indonesia

Abstract

Pneumonia is a serious respiratory infection that poses a significant global health burden, particularly in regions with limited access to medical personnel and diagnostic resources. Chest X-ray imaging remains the most common method for pneumonia diagnosis; however, manual interpretation is prone to error and often requires experienced radiologists. To address this challenge, automated diagnostic systems based on deep learning have gained increasing attention. This study aims to evaluate the effectiveness of the Convolutional Kolmogorov-Arnold Network (CKAN) in detecting pneumonia from chest X-ray images and and to compare its performance against a baseline Convolutional Neural Network (CNN) model. The study involved three variations of the CKAN architecture, which combined convolutional layers with Kolmogorov-Arnold-based layers. Both CKAN and CNN models were trained on balanced and imbalanced datasets using data augmentation techniques to improve model robustness. Additional experiments were conducted with and without the application of early stopping mechanisms. Performance evaluation was conducted using five metrics: accuracy, precision, recall, specificity, and balanced accuracy. Loss histories and confusion matrices were also analyzed to assess learning stability and classification behavior. The best-performing CKAN model achieved an accuracy of 83.49%, precision of 79.96%, recall of 98.21%, specificity of 78.59%, and balanced accuracy of 78.59%. In comparison, the best-performing CNN model achieved 81.00%, 77.98%, 97.18%, 75.73%, and 75.73%, respectively. These findings demonstrate CKAN's superior generalization capability and its effectiveness in handling class imbalance. In conclusion, CKAN shows promising potential for improving pneumonia detection from chest X-rays using a more compact and interpretable model structure. Future studies can explore hyperparameter optimization and extend the method to other medical imaging tasks. This work contributes to the development of more accurate and accessible automated diagnostic systems.

Paper History
Received April 6, 2025
Revised July 20, 2025
Accepted August 3, 2025
Published August 8, 2025

Keywords
Convolutional Kolmogorov-Arnold Network;
Convolutional Neural Network;
Pneumonia Detection;
Medical Image Analysis;
Chest X-Rays

Author Email
riechiemistery@gmail.com
virajessica998@gmail.com
kurniawanmatth@gmail.com
feliks.parningotan@uph.edu

I. Introduction

Pneumonia remains one of the leading causes of mortality worldwide, claiming over 4 million lives annually, particularly among children and the elderly in regions with limited access to healthcare services [1]. Timely and accurate detection is critical to reducing these mortality rates, with chest X-ray imaging being the most commonly used diagnostic method for pneumonia, as stated in [2], [3], and [4]. However, X-ray-based diagnoses often rely on radiologists, whose expertise may be unavailable in resource-constrained areas, underscoring the need for automated diagnostic tools.

Deep learning, particularly Convolutional Neural Networks (CNNs), has demonstrated remarkable success in medical imaging applications, including pneumonia detection, due to its ability to identify complex patterns in visual data [5]. In parallel, the Convolutional Kolmogorov-Arnold Network (CKAN) offers a promising alternative. By leveraging spline-based convolutional layers, CKAN can more effectively capture non-linear relationships, thereby improving model interpretability and computational efficiency [6]. Despite its potential, CKAN remains underexplored in medical imaging, presenting an opportunity for innovation in automated disease detection.

This study evaluates and compares the performance of CNN and CKAN in detecting pneumonia from chest X-ray images, with a particular focus on CKAN, whose application in this area is still limited. By examining CKAN's capabilities alongside those of CNN, this study aims to assess its potential as a reliable and efficient diagnostic tool. The results of this study are expected to enhance the understanding of CKAN's role in medical imaging and contribute to the development of scalable automated diagnostic methods suitable for diverse healthcare environments, including resource-limited settings.

The rapid advancement of technology has significantly propelled the integration of machine learning (ML) and deep learning (DL) across various domains, as evidenced in recent studies [7], [8]. In particular, the application of DL for automating disease diagnosis from medical images has been extensively explored in recent years; a few

Corresponding author: Feliks Victor Parningotan Samosir, feliks.parningotan@uph,edu, Informatics Department, Faculty of Information Technology (FIT), Universitas Pelita Harapan, Tangerang, Banten, Indonesia.
DOI: https://doi.org/10.35882/ijeeemi.v7i3.106

examples include [9], [10], and [11]. Among the various DL methods, Convolutional Neural Networks (CNNs) have emerged as the preferred approach due to their ability to extract critical features from medical images with high accuracy [12]. To address challenges such as computational overhead and the risk of overfitting, Rahman et al. [13] utilized transfer learning with pre-trained models such as AlexNet, ResNet18, DenseNet201, and SqueezeNet to detect pneumonia from chest X-ray (CXR) images, achieving a peak accuracy of 98%. Similarly, Zhang et al. [14] proposed a lightweight VGG16-based architecture that outperformed other pre-trained models in metrics such as area under the curve (AUC), precision, and recall.

Ensemble learning methods have also gained prominence in pneumonia detection using CXR. Kundu et al. [15] demonstrated that combining models such as GoogleNet, ResNet-18, and DenseNet121 resulted in superior performance compared to individual models. Meanwhile, Yaseliani et al. [16] introduced a hybrid CNN approach that integrated fully connected layers with Support Vector Machine (SVM) and Logistic Regression (LR), achieving an accuracy of 98.55%. Notably, Harshvardhan et al. [17] found that simpler CNN architectures often outperformed more complex models in pneumonia detection tasks, highlighting the potential of streamlined designs as efficient yet accurate solutions.

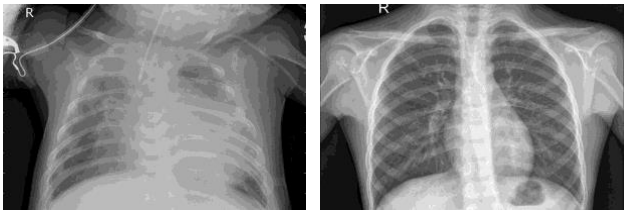
The Kolmogorov-Arnold Network (KAN) algorithm, although widely applied across various research fields [18], [19], [20], has only recently attracted attention as a viable alternative in medical image analysis. Drokin [21] demonstrated that integrating KAN into CNNs to form the Convolutional KAN (CKAN) can enhance both efficiency and accuracy. Cheon and Mun [22] applied CKAN in remote sensing tasks, finding it offered comparable accuracy to pre-trained models while operating at a faster speed. Similarly, Elaziz et al. [23] showcased CKAN's advantages in detecting IoT intrusions, achieving competitive performance with fewer parameters than other DL models.

Although CKAN has shown significant potential in various domains, its application in medical image analysis, particularly for disease detection, remains underexplored. This study seeks to address this gap by evaluating the performance of CNN and CKAN in detecting pneumonia from chest X-ray images, focusing on accuracy, computational efficiency, and model generalization. To ensure a fair comparison, transfer learning is deliberately excluded, given the simplicity of the architectures under evaluation and the limited availability of pre-trained CKAN models at the time of this study.

II. Materials and Methods

A. Dataset

The dataset utilized in this study comprises anterior-posterior (AP) chest X-ray images of pediatric patients aged 1 to 5 years, including images of both healthy individuals and those diagnosed with pneumonia. These data were retrospectively collected from medical records at the Guangzhou Women and Children's Medical Center and were originally obtained as part of routine clinical care. The dataset is publicly available through Mendeley Data under the title "Labeled Optical Coherence Tomography (OCT) and Chest X-Ray Images for Classification", curated by Kermany et al. [24]. Representative examples of normal and pneumonia-affected X-ray images are presented in Fig. 1.



(a) (b)

Fig. 1. Examples of chest X-ray images: (a) normal, (b) pneumonia-affected

B. Data Preprocessing

Before the data can be processed by machine learning models, several preprocessing steps are necessary. In this study, the images were resized to dimensions of 64×64 pixels. This input size was chosen based on prior research, such as [17] and [25], which demonstrated that models with an input resolution of 64×64 pixels achieved optimal performance. Following resizing, the images were converted into tensor format to be compatible with the models, with pixel values normalized to the range of 0 to 1.

Table 1 presents the distribution of training, testing, and validation datasets for each class. An imbalance is evident between the two classes, with only 1,341 normal cases compared to 3,875 pneumonia cases in the training dataset. To address this class imbalance, data augmentation was applied to increase the representation of the minority class. This technique was exclusively applied to the minority class to enhance data diversity, ensuring a balanced dataset. The augmentation process was repeated until both classes contained an equal number of samples.

Table 1. Distribution of training, testing, and validation datasets

	Training	Testing	Validation
Normal	1341	234	8
Pneumonia	3875	390	8
Total	5216	624	16

In addition to balancing the dataset, data augmentation enhances the model's robustness to real-world variations, such as changes in rotation, scale, lighting, or color. This improves the model's generalizability and resilience when exposed to unseen data [26]. Table 2 summarizes the data augmentation techniques employed in this study. All augmentation techniques were implemented using the transforms module from the Torchvision package.

Table 2. Augmentation techniques used in this study

Corresponding author: Feliks Victor Parningotan Samosir, feliks.parningotan@uph,edu, Informatics Department, Faculty of Information Technology (FIT), Universitas Pelita Harapan, Tangerang, Banten, Indonesia.
DOI: https://doi.org/10.35882/ijeeemi.v7i3.106

Augmentation	Parameters
Random Resized Crop	size : 64 scale : (0.8, 1.0)
Random Horizontal Flip	p : 0.5
Random Rotation	degrees : 15
Color Jitter	brightness : 0.2 contrast : 0.2 saturation : 0.2 hue : 0.1

C. Artificial Neural Network

An Artificial Neural Network (ANN), also known as a Multi-Layer Perceptron (MLP), is a computational model inspired by the structure and functioning of the human brain. It consists of interconnected nodes, or artificial neurons, organized into layers [27]. Each node receives input, processes the information using associated weights and an activation function, and produces an output that can be passed to subsequent nodes. ANNs are capable of learning patterns from data through a training process in which the weights of the connections between neurons are iteratively updated to minimize the error between predicted and expected outputs. This optimization process is known as backpropagation. An ANN architecture typically comprises three types of layers: the input layer, hidden layers, and the output layer.

Let a_i^l represent the i-th node in the l-th layer, where $l = 1, 2, \dots L$ and L is the total number of layers in the network. The value of a_i^l is determined by all nodes in the preceding layer and the weights associated with each of those nodes for a_i^l. Let θ_{ij}^l denote the weight connecting the j-th node in the l-th layer to the i-th node in the $l+1$-th layer. In an ANN, for a node in the hidden layer, the linear combination of inputs from the previous layer is processed through an activation function σ to produce the final value of the node. If layer l contains n nodes, the value of a_i^{l+1} is calculated using Eq. (1) [28].

$$a_i^{l+1} = \sigma(\theta_{i0}^l + \theta_{i1}^l a_1^l + \theta_{i2}^l a_2^l + \cdots + \theta_{in}^l a_n^l) \quad (1)$$

D. Kolmogorov-Arnold Network

The Kolmogorov-Arnold Network (KAN) is a neural network architecture based on the Kolmogorov-Arnold representation theorem, which states that any multivariate continuous function can be expressed as a combination of univariate functions [28]. Unlike traditional Multi-Layer Perceptrons (MLPs) that use fixed activation functions at their nodes, KAN implements learnable activation functions at the network's endpoints. These learnable functions replace the weights typically used in conventional MLP architectures. According to the Kolmogorov-Arnold representation theorem, a multivariate function $f(x_1, x_2, \dots, x_n)$ can be expressed as shown in Eq. (2) [28],

$$f(x_1, x_2, \dots, x_n) = \sum_{q=1}^{2n+1} \Phi_q\left(\sum_{p=1}^{n} \phi_{q,p}(x_p)\right), \quad (2)$$

where $\phi_{q,p}$ are univariate functions mapping each input variable x_p, such that $\phi_{q,p}: [0,1] \to \mathbb{R}$, and $\Phi_q: \mathbb{R} \to \mathbb{R}$ are univariate functions as well. Each layer in KAN can be seen as a matrix of these one-dimensional functions. The overall structure of a KAN is defined as in Eq. (3) [28],

$$KAN(x) = (\Phi_{L-1} \circ \Phi_{L-2} \circ \cdots \circ \Phi_0)(x), \quad (3)$$

where L represents the number of layers in the network. KAN has demonstrated promising performance in predicting the parameters of complex systems while using fewer parameters than conventional neural network architectures. Its primary advantage lies in its ability to approximate non-linear relationships with high accuracy through a simpler and more interpretable structure.

E. Convolutional Neural Network

A Convolutional Neural Network (CNN) is a specialized deep learning architecture designed to process data with a grid-like topology, such as images [29]. CNNs consist primarily of two types of layers: convolutional layers and dense layers (also referred to as fully connected layers or ANN layers in this context). An illustration of a typical CNN architecture is presented in Fig. 2.

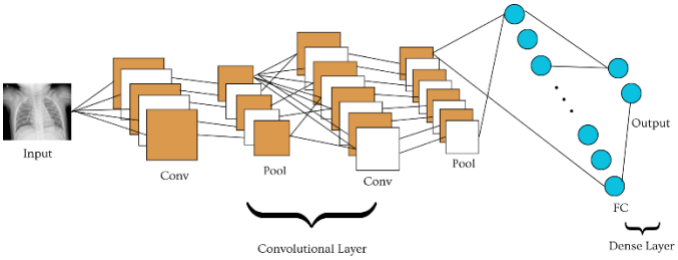


Fig. 2. Illustration of a CNN architecture

The convolutional layer is the cornerstone of CNNs, employing convolution operations to extract key features from input data. Each layer captures visual patterns at varying levels of abstraction, enabling the network to identify edges, textures, and shapes. This operation uses a kernel, a small matrix that slides over the image, to detect spatial patterns. The spatial relationships derived from these convolutions are crucial for distinguishing between the characteristics of different objects within an image. Mathematically, the convolution operation between two functions $f(t)$ and $g(t)$ is defined as in Eq. (4) [17],

$$(f * g)(t) := \int_{-\infty}^{\infty} f(\tau) g(t-\tau) d\tau. \quad (4)$$

Following the convolutional layers, pooling layers are applied to reduce the spatial dimensions of the extracted features while retaining their most critical information. Pooling operations enhance computational efficiency and reduce the risk of overfitting by summarizing feature representations. The processed features are then flattened and passed to dense layers, also known as MLPs. These dense layers organize and interpret the extracted features to generate the final prediction. This hierarchical architecture, composed of convolutional, pooling, and dense layers, has proven highly effective for a range of computer vision tasks, including image

Corresponding author: Feliks Victor Parningotan Samosir, feliks.parningotan@uph,edu, Informatics Department, Faculty of Information Technology (FIT), Universitas Pelita Harapan, Tangerang, Banten, Indonesia.
DOI: https://doi.org/10.35882/ijeeemi.v7i3.106

classification, object detection, and semantic segmentation.

F. Convolutional Kolmogorov-Arnold Network

The Convolutional Kolmogorov-Arnold Network (CKAN) builds upon the Kolmogorov-Arnold Network (KAN) by integrating its principles into the dense layers of a Convolutional Neural Network (CNN). However, Bodner et al. [6] extended this concept by applying the Kolmogorov-Arnold representation theorem to convolutional layers, resulting in the development of KAN convolutional layers. The primary distinction between a KAN convolutional layer and a traditional convolutional layer lies in the kernel structure. In standard CNNs, the kernel consists of fixed weights, whereas in KAN convolutional layers, each kernel element ϕ is a non-linear function that can be learned. These kernel elements are modeled using B-splines, which offer flexibility and the ability to approximate complex non-linear relationships. Mathematically, each kernel element is defined as Eq. (5) [6],

$$\phi(x) = w_1 \cdot spline(x) + w_2 \cdot silu(x), \quad (5)$$

where w_1 and w_2 are learnable weight parameters. Let $K \in \mathbb{R}^{N \times M}$ represent the KAN convolutional kernel, and let an image of dimensions $p \times h$ be represented as a matrix I, as shown in Eq. (6) [6],

$$I = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1p} \\ a_{21} & a_{22} & \cdots & a_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ a_{h1} & a_{h2} & \cdots & a_{hp} \end{bmatrix}. \quad (6)$$

The KAN convolution layer is then defined as Eq. (7) [6],

$$(I * K)_{i,j} = \sum_{k=1}^{N} \sum_{l=1}^{M} \phi_{k,l}(a_{i+k,j+l}), \quad (7)$$

where $\phi_{k,l}$ represents the non-linear function for each kernel element. The integration of learnable non-linear functions into the convolutional kernel enables the KAN convolutional layer to capture complex relationships in image data with greater precision and fewer parameters compared to traditional CNNs. This approach holds significant promise for improving computational efficiency and predictive performance in image-based tasks.

G. Experimental Design

This study employs a Convolutional Neural Network (CNN) as the baseline model, while the proposed model is the Convolutional Kolmogorov-Arnold Network (CKAN). Three CKAN variations were designed and evaluated:

- a combination of convolutional layers and KAN layers, abbreviated as CKAN,
- a combination of KAN convolutional layers and fully connected (MLP) layers, abbreviated as KANC_MLP, and
- a combination of KAN convolutional layers and KAN layers, abbreviated as KKAN.

For each model, experiments were conducted using two architectures with different levels of complexity: a simple architecture and a complex architecture. A detailed overview of the tested models and their respective architectures is provided in Fig. 3. The general structure of each layer used in Fig. 3 is described as follows.

- Convolutional layer: convolution type (standard convolutional or KAN convolutional), input channels, output channels, kernel size.
- Pooling layer: pooling type, kernel size.
- Dense layer: fully connected layer type (ANN or KAN), input size, output size.

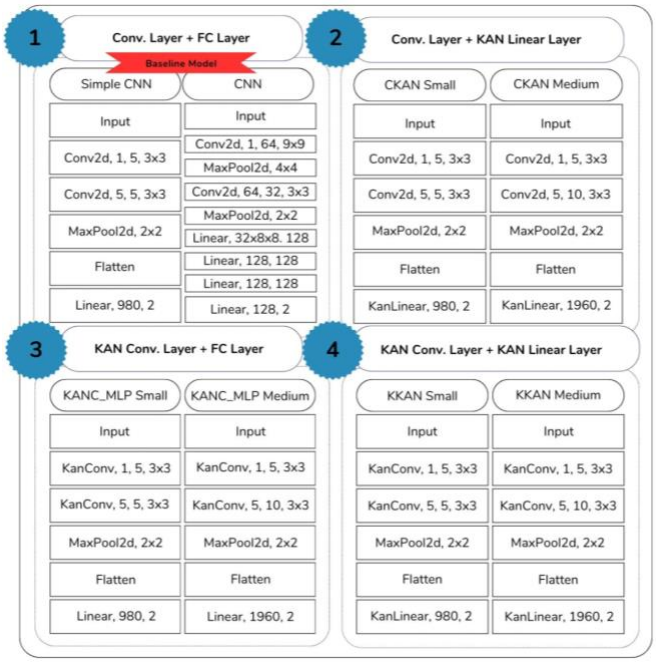


Fig. 3. Proposed model architecture used in experiments

For the CNN models, the Rectified Linear Unit (ReLU) activation function was applied to both convolutional layers and fully connected layers. In contrast, activation functions in the KAN-based models were automatically learned during the training process. The kernel size for both the convolutional layers and KAN convolutional layers was set to 3×3, with a stride of 1 and no padding. For pooling layers, a kernel size of 2×2 was used. In KAN-based models, the grid size for B-spline-based kernels was set to 5, and the B-spline degree was fixed at 3.

These hyperparameter choices were primarily based on the work of Bodner et al. [6], except for the complex CNN architecture, which followed the design outlined by Harshvardhan et al. [17]. In the complex CNN architecture, dropout regularization was applied to mitigate overfitting, and the convolutional and pooling kernel sizes varied across layers to enhance feature extraction capabilities. In the output layer of all models, two nodes were used to represent the two classes: normal and pneumonia. A softmax activation function was applied to the output layer to generate class membership probabilities, enabling direct interpretation of the predictions as probabilities.

Experiments were conducted on both the original dataset and the augmented dataset to determine the significance of augmentation techniques in this context. All models were trained using the Adaptive Moment

Corresponding author: Feliks Victor Parningotan Samosir, feliks.parningotan@uph,edu, Informatics Department, Faculty of Information Technology (FIT), Universitas Pelita Harapan, Tangerang, Banten, Indonesia.
DOI: https://doi.org/10.35882/ijeeemi.v7i3.106

Estimation (Adam) optimizer, with an initial learning rate of 0.001, and categorical cross-entropy as the loss function. These hyperparameters were selected empirically rather than fine-tuned, based on their reliability and widespread adoption in CNN training, as reported by various studies [30]. Furthermore, to reduce the computational burden during the training process, the dataset was fed to the models using a batch size of 32.

To prevent overfitting, an early stopping mechanism was implemented: training was halted if the validation loss did not improve for five consecutive epochs, with the default number of epochs set to 20. Validation performance was monitored using the categorical cross-entropy loss function, with no minimum threshold for improvement, meaning any decrease in validation loss was considered sufficient to reset the early stopping counter. In addition to validation loss, training loss was also monitored to provide a more complete understanding of the model's learning behavior. However, it is possible that the validation data may not adequately represent the testing data, and the early stopping mechanism could potentially hinder the model's learning ability. To address this concern, additional experiments were conducted without applying the early stopping mechanism, although the number of epochs was reduced to 10 to avoid overfitting.

All data preprocessing, model implementation, training, and evaluation in this study were conducted using Python 3.10.12 on Google Colaboratory. The models were developed and trained using the PyTorch library (version 2.0.1+cu118), which was utilized for all stages including data loading, augmentation, model definition, training loops, and evaluation. For deep learning acceleration, the experiments were executed on a paid Google Colab runtime equipped with an NVIDIA T4 GPU (16 GB GDDR6 memory).

H. Evaluation Metrics

To assess the performance of the models, five evaluation metrics were employed: accuracy, precision, recall, specificity, and balanced accuracy. Before delving into these metrics, it is essential to define the concepts of True Positive (TP), True Negative (TN), False Positive (FP), and False Negative (FN).

In a binary classification task, the output consists of two classes, typically labeled as positive and negative. A True Positive (TP) represents the number of correct predictions in which the model accurately identifies the positive class. Similarly, a True Negative (TN) is the count of correct predictions where the model identifies the negative class accurately. A False Positive (FP) occurs when the model incorrectly predicts the negative class as positive, while a False Negative (FN) occurs when the model misclassifies the positive class as negative. Eq. (8) – (12) [13] calculates accuracy, precision, recall, specificity, and balanced accuracy respectively.

$$Accuracy = \frac{TP+TN}{TP+FP+TN+FN}, \quad (8)$$

$$Precision = \frac{TP}{TP+FP}, \quad (9)$$

$$Recall = \frac{TP}{TP+FN}, \quad (10)$$

$$Specificity = \frac{TN}{TN+FP}, \quad (11)$$

$$Balanced\ Accuracy = \frac{Recall+Specificity}{2}. \quad (12)$$

III. Results

This section presents the key findings of the study. The initial experiments were conducted on the dataset without augmentation, followed by experiments incorporating augmented data. The purpose of this comparison was to evaluate whether balancing the dataset through augmentation could significantly improve model performance. After conducting these experiments, the authors suspected that the validation data might not adequately represent the testing data. Therefore, additional experiments using augmented data without applying early stopping were also conducted. The analysis includes the calculation of evaluation metrics such as accuracy (Acc), precision (Prec), recall (Rec), specificity (Spec), and balanced accuracy (Bal Acc), as well as the examination of loss history graphs and interpretation of the confusion matrix.

A. Model Performances

Table 3 presents the performance of models trained without data augmentation, Table 4 summarizes the results with augmented data, and Table 5 outlines the performance with augmented data but without early stopping. For each table, the best training performance is highlighted in orange, while the best testing performance is highlighted in yellow. Overall, the tables show that the proposed models consistently outperformed the baseline model on both the training and testing datasets.

Table 3. Model performances without data augmentation

Model	Dataset	Acc	Pre	Rec	Spec	Bal Acc
Simple CNN	Train	0.96877	0.96990	0.98866	0.91126	0.94996
	Test	0.73718	0.70620	0.99231	0.31197	0.65214
CNN	Train	0.94444	0.93419	0.99536	0.79717	0.89626
	Test	0.70192	0.67770	0.99744	0.20940	0.60342
CKAN Small	Train	0.98774	0.98648	0.99716	0.96048	0.97882
	Test	0.72115	0.69286	0.99487	0.26496	0.62991
CKAN Medium	Train	0.98352	0.98318	0.99484	0.95078	0.97281

Corresponding author: Feliks Victor Parningotan Samosir, feliks.parningotan@uph,edu, Informatics Department, Faculty of Information Technology (FIT), Universitas Pelita Harapan, Tangerang, Banten, Indonesia.
DOI: https://doi.org/10.35882/ijeeemi.v7i3.106

	Test	0.73237	0.70163	0.99487	0.29487	0.64487
KANC_MLP Small	Train	0.97663	0.97690	0.99201	0.93214	0.96207
	Test	0.72917	0.70054	0.98974	0.29487	0.64231
KANC_MLP Medium	Train	0.97739	0.97984	0.98995	0.94109	0.96552
	Test	0.75321	0.72015	0.98974	0.35897	0.67436
KKAN Small	Train	0.98793	0.99533	0.98840	0.98658	0.98749
	Test	**0.77083**	**0.73614**	**0.98718**	**0.41026**	**0.69872**
KKAN Medium	**Train**	**0.99330**	**0.99206**	**0.99897**	**0.97688**	**0.98793**
	Test	0.72436	0.69675	0.98974	0.28205	0.63590

Table 4. Model performances with data augmentation

Model	Dataset	Acc	Prec	Rec	Spec	Bal Acc
Simple CNN	Train	0.96881	0.96190	0.97628	0.96133	0.96881
	Test	0.78205	0.74517	0.98974	0.43590	0.71282
CNN	Train	0.96581	0.95696	0.97548	0.95613	0.96581
	Test	0.75481	0.72150	0.98974	0.36325	0.67650
CKAN Small	Train	0.98129	0.97723	0.98555	0.97703	0.98129
	Test	0.78205	0.74330	0.99487	0.42735	0.71111
CKAN Medium	Train	0.97226	0.95980	0.98581	0.95871	0.97226
	Test	0.76442	0.72710	0.99744	0.37607	0.68675
KANC_MLP Small	Train	0.97368	0.95716	0.99174	0.95561	0.97368
	Test	0.75321	0.71691	1.00000	0.34188	0.67094
KANC_MLP Medium	Train	0.97794	0.97053	0.98581	0.97006	0.97794
	Test	0.76282	0.72830	0.98974	0.38462	0.68718
KKAN Small	Train	0.98877	0.98218	0.99561	0.98194	0.98877
	Test	0.77724	0.73996	0.99231	0.41880	0.70556
KKAN Medium	**Train**	**0.99200**	**0.98573**	**0.99845**	**0.98555**	**0.99200**
	Test	**0.78846**	**0.75000**	**0.99231**	**0.44872**	**0.72051**

Table 5. Model performances with data augmentation and no early stopping

Model	Dataset	Acc	Prec	Rec	Spec	Bal Acc
Simple CNN	Train	0.96826	0.98220	0.95381	0.98271	0.96826
	Test	0.81090	0.77984	0.97179	0.54274	0.75726
CNN	Train	0.97497	0.98345	0.96619	0.98374	0.97497
	Test	0.78686	0.75246	0.98205	0.46154	0.72179
CKAN Small	Train	0.98039	0.97890	0.98194	0.97884	0.98039
	Test	0.78365	0.74854	0.98462	0.44872	0.71667
CKAN Medium	Train	0.97961	0.99128	0.96774	0.99148	0.97961
	Test	**0.83494**	**0.79958**	**0.98205**	**0.58974**	**0.78590**
KANC_MLP Small	Train	0.97277	0.96076	0.98581	0.95974	0.97277
	Test	0.75801	0.72253	0.99487	0.36325	0.67906
KANC_MLP Medium	Train	0.97806	0.97173	0.98477	0.97135	0.97806
	Test	0.76282	0.72830	0.98974	0.38462	0.68718
KKAN Small	Train	0.97806	0.96002	0.99768	0.95845	0.97806
	Test	0.75641	0.72201	0.99231	0.36325	0.67778
KKAN Medium	**Train**	**0.98426**	**0.97255**	**0.99665**	**0.97187**	**0.98426**
	Test	0.75801	0.72171	0.99744	0.35897	0.67821

Corresponding author: Feliks Victor Parningotan Samosir, feliks.parningotan@uph,edu, Informatics Department, Faculty of Information Technology (FIT), Universitas Pelita Harapan, Tangerang, Banten, Indonesia.
DOI: https://doi.org/10.35882/ijeeemi.v7i3.106

Table 6. Summary of best performance for all models

	CNN	CKAN	KANC_MLP	KKAN
Complexity	Simple	Medium	Medium	Medium
Augmentation	TRUE	TRUE	TRUE	TRUE
Early Stopping	FALSE	FALSE	TRUE	TRUE
Accuracy	0.8109	0.83494	0.76282	0.78846
Precision	0.77984	0.79958	0.7283	0.75
Recall	0.97179	0.98205	0.98974	0.99231
Specificity	0.54274	0.58974	0.38462	0.44872
Balanced Accuracy	0.75726	0.7859	0.68718	0.72051

The results indicate that the best baseline performance was achieved by the Simple CNN (SimpleCNN) model, trained on a balanced dataset through augmentation and without early stopping, as shown in Table 5 and Table 6. This model achieved a testing accuracy, precision, recall, specificity, and balanced accuracy of 81.00%, 77.98%, 97.18%, 75.73%, and 75.73%, respectively.

Conversely, the best performance among the proposed models was achieved by the CKAN with a medium-complexity architecture (CKAN_Medium), also trained on a balanced dataset with augmentation and without early stopping, also presented in Table 5 and Table 6. This model outperformed SimpleCNN across all evaluation metrics, with accuracy, precision, recall, specificity, and balanced accuracy of 83.49%, 79.96%, 98.21%, 78.59%, and 78.59%, respectively.

B. Model Loss

To further assess model performance and training behavior, loss history graphs of the best-performing architectures for each model are presented. These visualizations illustrate the progression of training and validation loss across epochs. Fig. 4 shows the loss history of the best-performing CNN model, while Fig. 5 presents the loss curve for the best-performing CKAN model. Similarly, Fig. 6 and Fig. 7 depict the training and validation loss curves for the best-performing KANC_MLP and KKAN models, respectively. These plots provide insights into model convergence, overfitting tendencies, and training stability.

Across all models, training loss consistently decreased over time, indicating effective learning. However, variations were observed in the behavior of validation loss. The CKAN model demonstrated a more stable and steadily declining validation loss, while the CNN and KANC_MLP models showed signs of overfitting in later epochs, as evidenced by rising validation loss.

C. Confusion Matrices

To evaluate how effectively each best-performing model classifies the pneumonia and normal cases, their corresponding confusion matrices are presented. A confusion matrix summarizes prediction results by comparing actual versus predicted classes. It displays the number of TP, FP, TN, and FN in a tabular format. Fig. 8 displays the confusion matrix for the best-performing CNN model, while Fig. 9 shows the confusion matrix for the best-performing CKAN model. Likewise, Fig. 10 and Fig. 11 show the confusion matrix for the best-performing KANC_MLP and KKAN models, respectively.

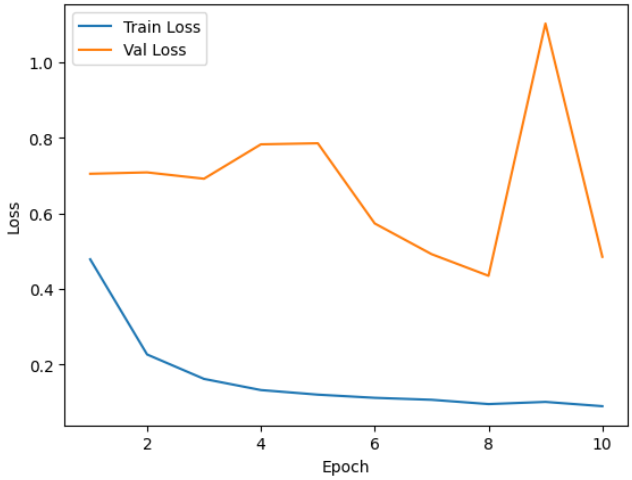


Fig. 4. Loss graph of the best-performing CNN model

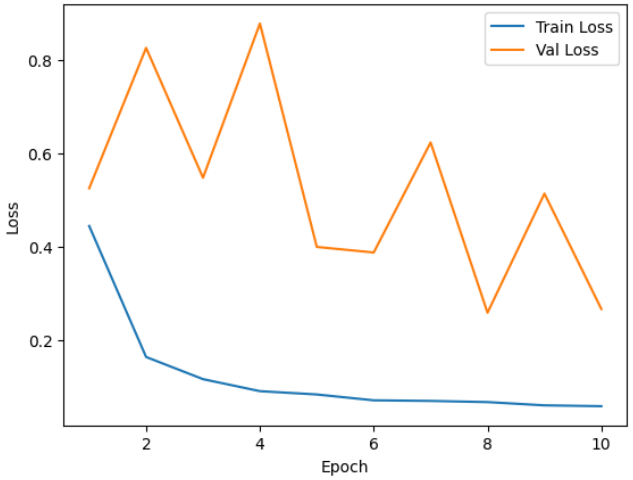


Fig. 5. Loss graph of the best-performing CKAN model

Corresponding author: Feliks Victor Parningotan Samosir, feliks.parningotan@uph,edu, Informatics Department, Faculty of Information Technology (FIT), Universitas Pelita Harapan, Tangerang, Banten, Indonesia.
DOI: https://doi.org/10.35882/ijeeemi.v7i3.106

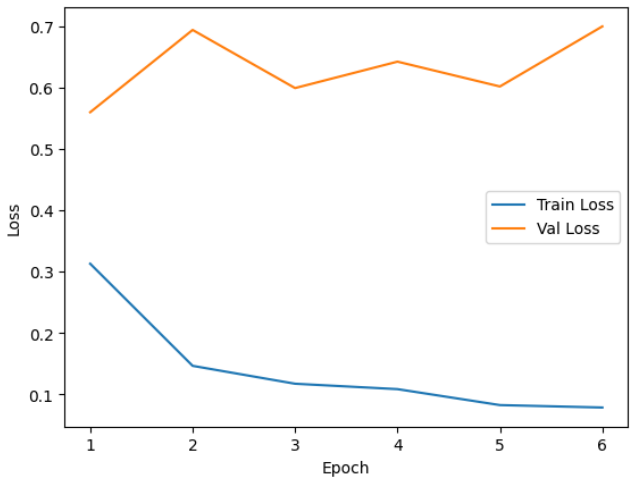


Fig. 6. Loss graph of the best-performing KANC_MLP model

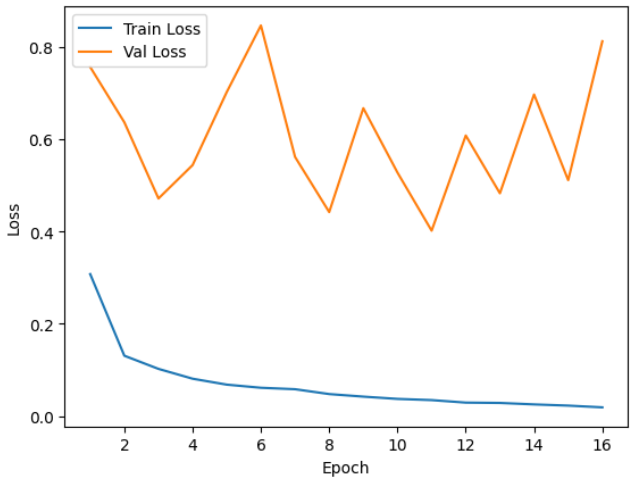


Fig. 7. Loss graph of the best-performing KKAN model

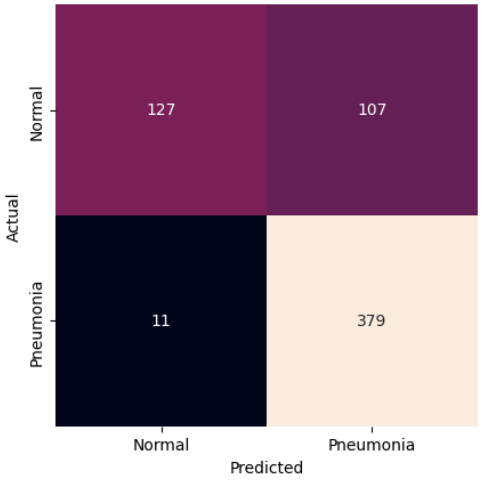


Fig. 8. Confusion matrix of the best-performing CNN model

Across the models, the confusion matrices reveal varying degrees of class-wise prediction accuracy. While all models demonstrated strong performance in identifying pneumonia cases, some exhibited reduced accuracy in correctly classifying normal cases, reflecting trade-offs in sensitivity and specificity which will be further discussed in the discussion section.

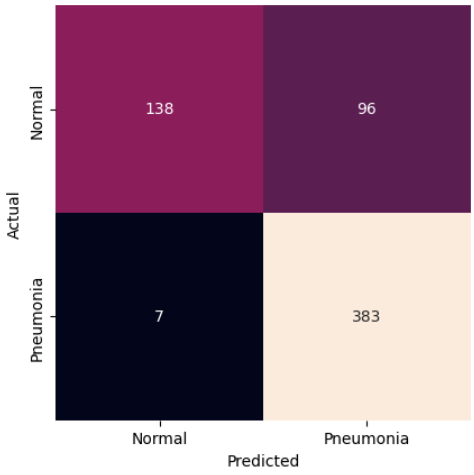


Fig. 9. Confusion matrix of the best-performing CKAN model

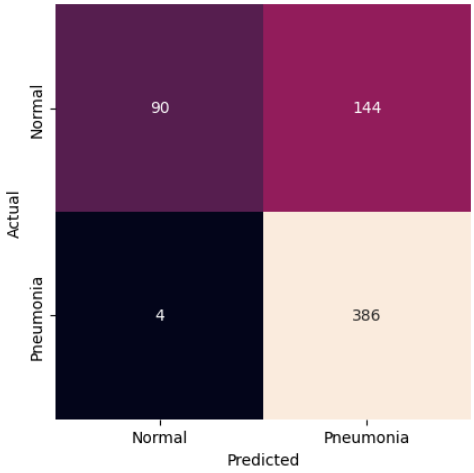


Fig. 10. Confusion matrix of the best-performing KANC_MLP model

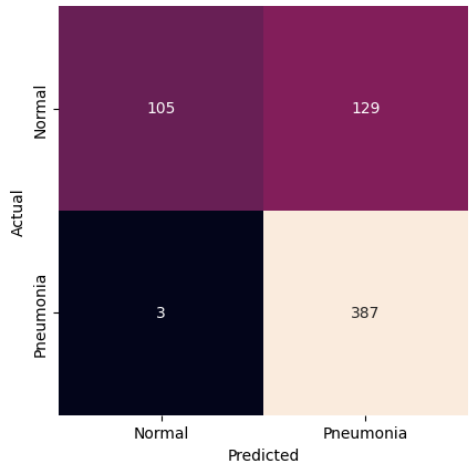


Fig. 11. Confusion matrix of the best-performing KKAN model.

Corresponding author: Feliks Victor Parningotan Samosir, feliks.parningotan@uph,edu, Informatics Department, Faculty of Information Technology (FIT), Universitas Pelita Harapan, Tangerang, Banten, Indonesia.
DOI: https://doi.org/10.35882/ijeeemi.v7i3.106

IV. Discussion

The results of this study provide several key insights into the applicability of the Convolutional Kolmogorov-Arnold Network (CKAN) in medical image analysis. Before discussing the performance of the CKAN-based models in detail, it is worth highlighting some notable findings related to the baseline CNN model. As shown in Table 5 and Table 6, the best-performing CNN variant was the simple CNN architecture. These findings suggest that, for CNNs, a simpler approach can yield competitive performance on a balanced dataset, particularly in detecting pneumonia cases, as evidenced by its high recall, also supported by the findings of Harshvardhan et al. [17]. However, the model's performance in detecting the normal class was limited, as reflected by its lower specificity. This is expected, given the underlying class imbalance. Although data augmentation was applied to oversample the normal class, the resulting samples are synthetic and may not fully capture the variability of real-world normal cases. Consequently, the model may struggle to generalize effectively to actual normal images during testing.

The best performance among the Convolutional Kolmogorov-Arnold Network models was achieved by the medium-complexity architecture, trained on a balanced dataset without the application of early stopping, as shown in Table 5 and Table 6. Notably, this model demonstrated an improvement in specificity, indicating greater effectiveness in identifying the normal class while maintaining excellent recall, demonstrating its robustness in detecting pneumonia cases .

Beyond accuracy and recall, a more comprehensive evaluation was conducted using precision, specificity, and balanced accuracy to account for the imbalanced nature of the dataset. The best-performing CKAN model achieved a precision of 79.96%, indicating a good balance between true positives and false positives when predicting pneumonia. This is important in clinical settings, where overdiagnosis can lead to unnecessary interventions. The specificity of 78.59% further reflects improved detection of normal cases, complementing the model's high recall of 98.21% for pneumonia detection. The balanced accuracy of 78.59% reinforces that the model performs well across both classes. Together, these metrics provide a more complete and fair assessment of the model's diagnostic performance.

From Table 3, Table 4, and Table 5, it can be observed that the best performance on the training dataset was consistently achieved by the KKAN model with a medium-complexity architecture (KKAN_Medium). This result suggests that the KKAN_Medium model has a superior ability to learn patterns from the training data. However, this strong learning capability may also contribute to reduced generalization ability, as indicated by the substantial gap between training and testing performance.

The pronounced differences in both accuracy and balanced accuracy between the training and testing datasets highlight the potential issue of overfitting across the models. Despite this, the CKAN_Medium model demonstrated a narrower gap between its training and testing performance compared to SimpleCNN. This finding suggests that CKAN_Medium exhibits better generalization capabilities, maintaining a higher level of performance on unseen data. These observations align with the findings of Elaziz et al. [23], which emphasize the advantages of CKAN in mitigating overfitting through its efficient parameterization and non-linear modeling. Conversely, the SimpleCNN model appeared more prone to overfitting, likely due to its reliance on a simpler architecture that may lack the flexibility required for robust generalization.

Both the baseline and proposed models exhibited performance improvements when trained on augmented data and without the use of early stopping, with a few exceptions. These improvements can be confidently attributed to the use of augmentation, as the model architectures and training procedures were kept identical across the experiments. For instance, CKAN_Medium showed an increase in accuracy from 73.24% (without augmentation) to 76.44% (with augmentation), with similar gains observed across other evaluation metrics. These findings underline the effectiveness of data balancing via augmentation techniques, a conclusion supported by the work of Yaseliani et al. [16]. By increasing the diversity and representation of the minority class, augmentation likely enhanced the models' ability to handle imbalanced datasets.

The improved performance achieved by omitting the early stopping mechanism further suggests that the validation data used in this study may not adequately represent the testing dataset. This mismatch between validation and testing data could result in premature halting of the training process, thereby limiting the models' potential to fully optimize their parameters. To address this limitation, employing cross-validation techniques could provide a more representative validation process, as demonstrated by Kundu et al. [15]. Cross-validation divides the dataset into multiple folds, ensuring that all subsets contribute to training and validation, which could help reduce the discrepancies observed in this study.

From the loss graphs in Fig. 4 through Fig. 7, it can be observed that all models effectively learned the training data, as indicated by steadily decreasing training loss curves. However, significant differences in validation loss patterns were noted. For the Simple CNN model (Fig. 4), validation loss remained stable during the initial epochs but spiked towards the end of training, indicating overfitting. A similar pattern was observed in the KANC_MLP Medium model (Fig. 6), where a considerable gap between training and validation loss further highlighted overfitting. The KKAN Medium model (Fig. 7) exhibited fluctuating validation loss with a persistent gap, also suggesting overfitting. In contrast, the CKAN Medium model (Fig. 5) showed smaller and more consistent gaps between training and validation loss, with validation loss steadily decreasing despite some fluctuations. This pattern indicates superior generalization compared to the other models.

The results from the confusion matrices in Fig. 8 through Fig. 11 reaffirm that the CKAN Medium model (Fig. 9) outperformed the others in recognizing both the

Corresponding author: Feliks Victor Parningotan Samosir, feliks.parningotan@uph,edu, Informatics Department, Faculty of Information Technology (FIT), Universitas Pelita Harapan, Tangerang, Banten, Indonesia.
DOI: https://doi.org/10.35882/ijeeemi.v7i3.106

normal and pneumonia classes. This is evident from the higher True Positive (TP) and True Negative (TN) values achieved by CKAN Medium, 383 and 138, respectively, compared to 379 and 127 for the Simple CNN (Fig. 8). In contrast, the KANC_MLP Medium and KKAN Medium models (Fig. 10 and Fig. 11) demonstrated stronger performance in detecting pneumonia cases but struggled with the normal class. Their respective TP and TN values were 386 and 90 for KANC_MLP Medium, and 387 and 105 for KKAN Medium. In the context of pneumonia detection, the ability to accurately identify pneumonia cases, represented by the TP value, holds greater clinical importance than recognizing the normal class, due to the critical implications of missed pneumonia diagnoses.

A closer examination of the misclassifications reveals clinically relevant insights. False negatives (FN), instances where pneumonia cases are incorrectly predicted as normal, pose the most critical risk, as they may result in missed or delayed treatment for affected patients. All Kolmogorov-Arnold architectures recorded fewer FN compared to traditional CNN models, with KKAN achieving the fewest FN of 3. This reinforce the reliability of Kolmogorov-Arnold architectures in minimizing diagnostic oversight. On the other hand, false positives (FP), where normal cases are misclassified as pneumonia, can lead to unnecessary follow-up testing, patient anxiety, and increased resource strain. The Simple CNN exhibited a higher FP rate than CKAN_Medium, with values of 107 and 96, respectively, which partly explains its lower specificity. These misclassification trends underscore CKAN_Medium's balanced diagnostic performance, offering a more clinically practical solution by effectively reducing both FN and FP rates.

Overall, the findings highlight the superiority of the CKAN model in capturing complex non-linear relationships through the integration of convolutional layers and KAN linear layers. The significant improvements observed across key evaluation metrics demonstrate the model's effectiveness compared to the baseline Simple CNN. Additionally, the use of data augmentation played a pivotal role in enhancing the performance of both models, particularly in addressing class imbalance. The CKAN_Medium architecture, in particular, excelled in generalization and achieved superior predictions for both the normal and pneumonia classes, making it the most robust and clinically applicable model in this study. To further validate these findings, future studies should apply statistical testing to determine whether the observed differences in predictive performance between CKAN and CNN are statistically significant and not due to random variation.

Beyond performance, model efficiency and deployment feasibility are also important considerations. CKAN introduced greater computational complexity than CNN, with average training times of 164.161 minutes compared to 31.475 minutes for CNN. This increase is due to CKAN's learnable spline-based activations and more complex operations. While it achieved better performance, CKAN also required more memory and had slower inference speeds, which may limit its suitability for real-time or low-resource clinical settings. Future research should explore model optimization techniques to improve CKAN's efficiency and deployment potential.

While this study provides meaningful insights into the performance of CKAN and CNN architectures for pneumonia detection, several limitations should be acknowledged. First, the focus was not on optimizing hyperparameters or fine-tuning architectural configurations to maximize performance. Instead, the primary aim was to conduct a fair and controlled comparison between CNN and CKAN variants under consistent experimental conditions. As a result, the reported accuracy, while competitive, remains lower than that achieved by some recent studies employing more aggressive optimization strategies.

This performance gap is particularly evident when benchmarked against recent state-of-the-art models that also utilized the publicly available dataset provided by Kermany et al. [24]. For instance, Rahman et al. [13] achieved an accuracy of 98% using transfer learning models such as AlexNet, ResNet-18, DenseNet-201, and SqueezeNet. Similarly, Zhang et al. [14] proposed a lightweight VGG-based architecture and achieved an accuracy of 96.07%. Kundu et al. [15] introduced an ensemble approach combining GoogleNet, ResNet-18, and DenseNet-121, reaching an accuracy of 98.81% and a sensitivity of 98.80%. Yaseliani et al. [16] applied a hybrid classification method integrating Support Vector Machine (SVM) and Logistic Regression (LR), achieving a peak accuracy of 98.55%. Despite these differences, the findings from this study highlight the potential of CKAN as a lightweight and interpretable alternative for pneumonia detection, which may be further improved with future architectural and hyperparameter optimization.

The second limitation is that the dataset used in this study was limited to pediatric chest X-ray images collected from a single medical center. This potentially restricts the generalizability of the findings to broader, more heterogeneous patient populations. Incorporating data from multiple sources, including adult patients and different acquisition settings, would strengthen the clinical relevance and applicability of the models. Additionally, the imbalanced nature of the dataset significantly hindered the models' ability to detect the normal class. Therefore, future research is encouraged to explore more effective data balancing strategies, as well as to seek or construct more balanced, real-world datasets to ensure better generalization and class-level performance.

Another limitation lies in the reliance on a fixed training-validation-testing split. As the improved performance from disabling early stopping suggests, the validation set may not have been fully representative of the testing distribution. This mismatch may have prematurely halted training, preventing models from reaching their full learning potential. To mitigate this, future studies should explore cross-validation techniques, which allow for more reliable performance estimation and reduce variance associated with dataset partitioning.

Moreover, this study did not incorporate any explainable AI (XAI) techniques to interpret how the models arrived at their predictions, which limits the

Corresponding author: Feliks Victor Parningotan Samosir, feliks.parningotan@uph,edu, Informatics Department, Faculty of Information Technology (FIT), Universitas Pelita Harapan, Tangerang, Banten, Indonesia.
DOI: https://doi.org/10.35882/ijeeemi.v7i3.106

interpretability of the results. Future research should explore the use of explainability methods, such as Gradient-weighted Class Activation Mapping (Grad-CAM), to visualize and analyze which regions of the chest X-ray images contribute most to the model's decisions. Incorporating such techniques would enhance interpretability and support greater clinical trust in AI-assisted diagnostic systems.

Despite these limitations, the findings have important implications. They demonstrate that CKAN, even without extensive optimization, offers a promising alternative to conventional CNNs for medical image classification tasks. Its ability to model complex non-linear relationships and achieve balanced performance across both classes highlights its potential utility in real-world clinical settings. In particular, the model's high recall in detecting pneumonia suggests it could serve as an effective screening tool to support radiologists by reducing the risk of missed diagnoses. This could be especially valuable in resource-constrained environments or high-volume hospitals, where rapid and accurate assessments are essential. However, the current computational demands and slower inference speed of CKAN may limit its immediate use in time-sensitive or low-resource applications. Future optimizations aimed at improving efficiency could help make the model more viable for real-time clinical deployment. By improving diagnostic support and potentially accelerating patient triage, such models may contribute to more timely interventions and better clinical outcomes.

V. Conclusion

This study evaluated the application of Convolutional Kolmogorov-Arnold Networks (CKAN) for detecting pneumonia in chest X-ray images, comparing its performance to a baseline Convolutional Neural Network (CNN) model. Consistent with the objectives outlined in the introduction, the experimental results demonstrated the effectiveness of CKAN in capturing complex non-linear relationships, resulting in improved performance over the baseline CNN. Among the tested architectures, CKAN_Medium achieved the best performance, with accuracy, precision, recall, specificity, and balanced accuracy of 83.49%, 79.96%, 98.21%, 78.59%, and 78.59%, respectively. These results highlight its superior generalization capabilities and its effectiveness in addressing class imbalance compared to the baseline SimpleCNN model, which achieved 81.00%, 77.98%, 97.18%, 75.73%, and 75.73%, respectively. Moreover, the use of data augmentation significantly improved the models' ability to handle imbalanced datasets, further strengthening the reliability of both CKAN and CNN architectures. These findings validate the potential of integrating KAN principles into convolutional layers to enhance performance in medical image analysis tasks.

As future research, this study can be extended by exploring alternative validation strategies, such as cross-validation, to ensure better alignment between validation and testing datasets and to minimize the risk of overfitting. Additionally, further optimization of CKAN hyperparameters, including grid size and B-spline degree, could lead to even more robust models. The proposed framework could also be applied to other medical imaging modalities, such as CT or MRI, to assess its generalizability across different diagnostic domains. The incorporation of explainable AI (XAI) techniques may further improve model interpretability. These enhancements have the potential to make automated diagnostic systems more reliable, scalable, and effective in real-world clinical settings.

References

[1] S. Sharma and K. Guleria, "A systematic literature review on deep learning approaches for pneumonia detection using chest X-ray images," Multimed Tools Appl, vol. 83, no. 8, pp. 24101–24151, Aug. 2023, doi: 10.1007/s11042-023-16419-1.

[2] Y. Li, Z. Zhang, C. Dai, Q. Dong, and S. Badrigilan, "Accuracy of deep learning for automated detection of pneumonia using chest X-Ray images: A systematic review and meta-analysis," Comput Biol Med, vol. 123, p. 103898, Aug. 2020, doi: 10.1016/j.compbiomed.2020.103898.

[3] S. Nalluri and R. Sasikala, "Pneumonia screening on chest X-rays with optimized ensemble model," Expert Syst Appl, vol. 242, p. 122705, May 2024, doi: 10.1016/j.eswa.2023.122705.

[4] M. Ali et al., "Pneumonia Detection Using Chest Radiographs With Novel EfficientNetV2L Model," IEEE Access, vol. 12, pp. 34691–34707, 2024, doi: 10.1109/ACCESS.2024.3372588.

[5] S. S. Kshatri and D. Singh, "Convolutional Neural Network in Medical Image Analysis: A Review," Archives of Computational Methods in Engineering, vol. 30, no. 4, pp. 2793–2810, May 2023, doi: 10.1007/s11831-023-09898-w.

[6] A. D. Bodner, A. S. Tepsich, J. N. Spolski, and S. Pourteau, "Convolutional Kolmogorov-Arnold Networks," Jun. 2024, [Online]. Available: http://arxiv.org/abs/2406.13155

[7] H. Christine and R. Mandala, "Hybrid of AdaBoost and Optimized LSTM Using Modified Bald Eagle Search in Predicting Concrete Compressive Strength," International Journal on Electrical Engineering and Informatics, vol. 15, no. 4, pp. 527–553, Dec. 2023, doi: 10.15676/ijeei.2023.15.4.2.

[8] V. H. Trong and P. T. Bao, "Late Fusion of Weak Information for Incompleted Face Recognition Using Convolutional Neural Networks: A Novel Approach," International Journal on Electrical Engineering and Informatics, vol. 16, no. 2, pp. 162–179, Jun. 2024, doi: 10.15676/ijeei.2024.16.2.2.

[9] M. S. Ahmed et al., "Joint Diagnosis of Pneumonia, COVID-19, and Tuberculosis from Chest X-ray Images: A Deep Learning Approach," Diagnostics, vol. 13, no. 15, p. 2562, Aug. 2023, doi: 10.3390/diagnostics13152562.

[10] R. Siddiqi and S. Javaid, "Deep Learning for

Corresponding author: Feliks Victor Parningotan Samosir, feliks.parningotan@uph,edu, Informatics Department, Faculty of Information Technology (FIT), Universitas Pelita Harapan, Tangerang, Banten, Indonesia.
DOI: https://doi.org/10.35882/ijeeemi.v7i3.106

Pneumonia Detection in Chest X-ray Images: A Comprehensive Survey," J Imaging, vol. 10, no. 8, p. 176, Jul. 2024, doi: 10.3390/jimaging10080176.

[11] H. Malik, T. Anees, M. Din, and A. Naeem, "CDC_Net: multi-classification convolutional neural network model for detection of COVID-19, pneumothorax, pneumonia, lung Cancer, and tuberculosis using chest X-rays," Multimed Tools Appl, vol. 82, no. 9, pp. 13855–13880, Apr. 2023, doi: 10.1007/s11042-022-13843-7.

[12] M. A. Abdou, "Literature review: efficient deep neural networks techniques for medical image analysis," Neural Comput Appl, vol. 34, no. 8, pp. 5791–5812, Apr. 2022, doi: 10.1007/s00521-022-06960-9.

[13] T. Rahman et al., "Transfer Learning with Deep Convolutional Neural Network (CNN) for Pneumonia Detection Using Chest X-ray," Applied Sciences, vol. 10, no. 9, p. 3233, May 2020, doi: 10.3390/app10093233.

[14] D. Zhang, F. Ren, Y. Li, L. Na, and Y. Ma, "Pneumonia Detection from Chest X-ray Images Based on Convolutional Neural Network," Electronics (Basel), vol. 10, no. 13, p. 1512, Jun. 2021, doi: 10.3390/electronics10131512.

[15] R. Kundu, R. Das, Z. W. Geem, G.-T. Han, and R. Sarkar, "Pneumonia detection in chest X-ray images using an ensemble of deep learning models," PLoS One, vol. 16, no. 9, p. e0256630, Sep. 2021, doi: 10.1371/journal.pone.0256630.

[16] M. Yaseliani, A. Z. Hamadani, A. I. Maghsoodi, and A. Mosavi, "Pneumonia Detection Proposing a Hybrid Deep Convolutional Neural Network Based on Two Parallel Visual Geometry Group Architectures and Machine Learning Classifiers," IEEE Access, vol. 10, pp. 62110–62128, 2022, doi: 10.1109/ACCESS.2022.3182498.

[17] H. Gm, M. K. Gourisaria, S. S. Rautaray, and M. Pandey, 'Pneumonia detection using CNN through chest X-ray', Journal of Engineering Science and Technology (JESTEC), vol. 16, no. 1, pp. 861–876, 2021.

[18] T.-T.-H. Le, Y. Hwang, H. Kang, and H. Kim, "Robust Credit Card Fraud Detection Based on Efficient Kolmogorov-Arnold Network Models," IEEE Access, vol. 12, pp. 157006–157020, 2024, doi: 10.1109/ACCESS.2024.3485200.

[19] D. Carneros-Prado, L. Cabañero-Gómez, E. Johnson, I. González, J. Fontecha, and R. Hervás, "A Comparison Between Multilayer Perceptrons and Kolmogorov-Arnold Networks for Multi-Task Classification in Sitting Posture Recognition," IEEE Access, vol. 12, pp. 180198–180209, 2024, doi: 10.1109/ACCESS.2024.3510034.

[20] A. T. Haryono, R. Sarno, R. N. E. Anggraini, and K. R. Sungkono, "Permuted Temporal Kolmogorov-Arnold Networks for Stock Price Forecasting Using Generative Aspect-Based Sentiment Analysis," IEEE Access, vol. 12, pp. 178672–178689, 2024, doi: 10.1109/ACCESS.2024.3506658.

[21] I. Drokin, "Kolmogorov-Arnold Convolutions: Design Principles and Empirical Studies," Jul. 2024, [Online]. Available: http://arxiv.org/abs/2407.01092

[22] M. Cheon and C. Mun, "Combining KAN with CNN: KonvNeXt's Performance in Remote Sensing and Patent Insights," Remote Sens (Basel), vol. 16, no. 18, p. 3417, Sep. 2024, doi: 10.3390/rs16183417.

[23] M. Abd Elaziz, I. Ahmed Fares, and A. O. Aseeri, "CKAN: Convolutional Kolmogorov–Arnold Networks Model for Intrusion Detection in IoT Environment," IEEE Access, vol. 12, pp. 134837–134851, 2024, doi: 10.1109/ACCESS.2024.3462297.

[24] D. Kermany, K. Zhang, and M. Goldbaum, "Labeled optical coherence tomography (OCT) and Chest X-Ray images for classification," 2018, Mendeley.

[25] A. K. Kushwaha and S. Rastogi, "Solution to OCT Diagnosis Using Simple Baseline CNN Models and Hyperparameter Tuning," 2022, pp. 353–366. doi: 10.1007/978-981-16-3071-2_30.

[26] A. Mumuni and F. Mumuni, "Data augmentation: A comprehensive survey of modern approaches," Array, vol. 16, p. 100258, Dec. 2022, doi: 10.1016/j.array.2022.100258.

[27] R. Aljarrah et al., "Application of Artificial Neural Network-Based Tool for Short Circuit Currents Estimation in Power Systems With High Penetration of Power Electronics-Based Renewables," IEEE Access, vol. 11, pp. 20051–20062, 2023, doi: 10.1109/ACCESS.2023.3249296.

[28] Z. Liu et al., "KAN: Kolmogorov-Arnold Networks," Apr. 2024, [Online]. Available: http://arxiv.org/abs/2404.19756

[29] Z. Li, F. Liu, W. Yang, S. Peng, and J. Zhou, "A Survey of Convolutional Neural Networks: Analysis, Applications, and Prospects," IEEE Trans Neural Netw Learn Syst, vol. 33, no. 12, pp. 6999–7019, Dec. 2022, doi: 10.1109/TNNLS.2021.3084827.

[30] H.-C. Chen et al., "AlexNet Convolutional Neural Network for Disease Detection and Classification of Tomato Leaf," Electronics (Basel), vol. 11, no. 6, p. 951, Mar. 2022, doi: 10.3390/electronics11060951.

Biography of Authors

Riechie received a Bachelor of Mathematics degree in Applied Mathematics from the Mathematics Department at Universitas Pelita Harapan in 2025, with a concentration in computational mathematics. His academic background focuses on machine learning, data science, and mathematical modeling. During his undergraduate studies, he served as a research assistant, working primarily in the fields of machine learning and data science. His research interests include neural networks, time series forecasting, financial

Corresponding author: Feliks Victor Parningotan Samosir, feliks.parningotan@uph,edu, Informatics Department, Faculty of Information Technology (FIT), Universitas Pelita Harapan, Tangerang, Banten, Indonesia.
DOI: https://doi.org/10.35882/ijeeemi.v7i3.106

engineering, and the application of clustering algorithms in stock portfolio optimization. He is particularly passionate about applying data-driven and interpretable models to address real-world problems in finance and healthcare, with a continued interest in advancing both practical and theoretical approaches to intelligent systems and predictive analytics.

Vira Jessica received a Bachelor of Mathematics degree in Applied Mathematics from the Mathematics Department at Universitas Pelita Harapan in 2025, specializing in computational mathematics. Her academic background includes a strong foundation in numerical methods, mathematical modeling, and machine learning. During her undergraduate studies, she actively engaged in research and served as the first author of a paper that received the Best Paper Award at the International Conference on Mathematics, Computational Sciences, and Statistics (ICoMCoS) 2024. Her work reflects a deep interest in applying computational techniques to real-world problems, and she continues to pursue opportunities in applied mathematics, data science, and intelligent systems research.

Matthew Kurniawan is an undergraduate student in the Informatics Study Program at Universitas Pelita Harapan, majoring in Computer Science with a concentration in Data Science. He specializes in computational data science, with academic interests spanning machine learning, data analytics, and algorithmic problem-solving. He has participated in several academic projects focused on predictive modeling, data-driven decision-making, and the development of intelligent systems. He is particularly interested in applying data science methodologies to real-world challenges in fields such as finance, healthcare, and technology, and aspires to contribute to research and innovation in these domains.

Feliks Victor Parningotan Samosir is a lecturer in the Informatics Department, at Universitas Pelita Harapan, specializing in natural language processing, and deep learning. He holds a Bachelor's degree in Educational Informatics from Universitas Negeri Malang and a Master's degree in Computer Science from Universitas Kristen Maranatha, Bandung. He currently serves as the Department Chair of the Informatics Department . In his teaching, he integrates research and practical applications, covering courses such as Introduction to Informatics, Advanced Machine Learning, Natural Language Processing, and Data Science Application Development Studio. His research includes publications on text summarization and sentiment analysis, with current work focusing on bias detection in machine learning, image classification, and the use of transfer learning for low-resource languages.

Corresponding author: Feliks Victor Parningotan Samosir, feliks.parningotan@uph,edu, Informatics Department, Faculty of Information Technology (FIT), Universitas Pelita Harapan, Tangerang, Banten, Indonesia.
DOI: https://doi.org/10.35882/ijeeemi.v7i3.106