Online-Journals.org

iJOE International Journal of Online and Biomedical Engineering

iJOE | eISSN: 2626-8493 | Vol. 21 No. 4 (2025) | OPEN ACCESS

https://doi.org/10.3991/ijoe.v21i04.53289

PAPER

Development of an Ultrasound Image Extraction Method for Detection and Classification of Kidney Abnormalities Using a Convolutional Neural Network

Ruri Hartika Zain(✉), Sumijan, Sarjon Defit

Department of Information Technology, Faculty of Computer Science, Universitas Putra Indonesia YPTK, Padang, Indonesia

rurihartika_zain@upiyptk.ac.id

ABSTRACT

The Kidney is an important organ that filters waste, toxins, and excess fluids from the blood and removes the waste in the urine. Kidney disease is a condition that occurs when it becomes damaged or impaired. In this study, a model was proposed using a convolutional neural network (CNN) and ML technology, which were trained on a dataset of 306 kidney ultrasound images. This development included Canny extraction methods, grey level co-occurrence matrix (GLCM), and principal component analysis (PCA) to obtain kidney abnormalities detection features. Additionally, the new method created was called Canny Grey Principal Component Analysis Pattern (CGPCAP). This CGPCAP method was developed to achieve better results in feature extraction and classification multi detection kidney used scale invariant feature transform (SIFT) to detect extreme, low-contrast, and edge areas in kidney images. Following this discussion, CGPCAP was tested on an image retrieved from the database, with a set of 16 features extracted. CGPCAP achieved high classification accuracy through the experimental use of a CNN classifier. Other objectives included performing feature extraction and classification between normal kidneys and kidney abnormalities. In this study, multi detection kidney was used to detect extreme, low-contrast, and edge areas in kidney images. CNN was used to classify kidney images based on the feature extraction results. Relating to this discussion, the results of the proposed method were compared with automatic feature extraction using Canny, GLCM, SIFT, and PCA. The extracted features were inputted to the CNN classifier, which achieved the highest accuracy of 97.5% compared to other abnormality detection methods. This suggests that the CGPCAP algorithm not only improves the model's ability to make a more accurate prediction but is also more efficient at handling more complex data. Overall, the use of CGPCAP with the CNN test algorithm provides better results in the context of renal abnormality detection.

KEYWORDS

Canny, grey level co-occurrence matrix (GLCM), principal component analysis (PCA), scale invariant feature transform (SIFT) and convolutional neural network (CNN)

Zain, R.H., Sumijan, Defit, S. (2025). Development of an Ultrasound Image Extraction Method for Detection and Classification of Kidney Abnormalities Using a Convolutional Neural Network. *International Journal of Online and Biomedical Engineering (iJOE)*, 21(4), pp. 150–170. https://doi.org/10.3991/ijoe.v21i04.53289

Article submitted 2024-11-12. Revision uploaded 2025-01-20. Final acceptance 2025-01-21.

1 INTRODUCTION

The kidney is a bean-shaped organ in the urinary system of the body, with the main function of removing waste products from the bloodstream in the form of urine [1], [2]. Kidney stones are small, solid crystal growths that can stick to the ureter or pass through it, depending on the size of the stone. Additionally, causes of kidney stones include diet, obesity, certain medical treatments, and dehydration [3], [4]. Kidney cysts are often found in people with advanced chronic kidney disease (CKD). These cysts are a common condition, which is more common in older people. The median prevalence of simple renal cysts varies depending on the study population and imaging method used [5], [6], [7]. When a kidney stone grows larger than five millimeters, it can block the ureter, leading to severe pain in the abdomen or lower back. Moreover, permanent kidney failure or even cancer can occur when treatment is delayed.

Artificial intelligence (AI) technology provides the necessary tools to address kidney issues. Therefore, this study presents a methodology based on deep learning (DL), specifically for the segmentation, detection, and classification of kidney diseases [8]. Detecting kidney stones becomes easier as the automated detection process reduces operator noise and helps to identify as well as classify existing stones [9], [10]. This result, which was inspired by previous studies, has motivated further investigation into the detection and classification of kidney abnormalities [11]. The study aims to support general practitioners in treating patients in the community, in outpatient clinics, and in emergencies, even in the absence of a nephrologist.

Convolutional neural network (CNN) is a type of ML algorithm that is well-suited to computer vision tasks. Modeling CNN is becoming an attractive architectural option for addressing medical imaging problems, creating clinical adjuncts that help doctors train, characterize, and evaluate difficult diagnostic cases [12], [13]. Since most of the ultrasound waves are reflected in the transducer, kidney stones appear brighter and hyperechoic on ultrasound images. The deep convolutional network architecture enables hierarchical feature extraction, enabling DL to become a major focus of AI. Ultrasound (US) imaging is an imaging modality that has many advantages, including affordability, non-radioactivity, non-invasiveness, and real-time, allowing it to be very popular in various radiological applications [14], [15].

Adjusting the parameters across a range of values shifts the result from an under segmented image to an over segmented image in the Canny method. The optimal result is achieved at the center of this range, where the image is most accurately segmented [16], [17]. Following the discussion, ordered edge detection methods such as the Canny algorithm aim to detect edges in images and objects without requiring prior information. For example, this algorithm uses Gaussian filters to improve edge detection accuracy [18], [19]. When using multiple directions to compute gradient magnitude, the adaptive Canny algorithm improves its path edge detection by adding adaptive median filtering and morphological closure operations [20], [21], [22].

Grey level co-occurrence matrix (GLCM) extracts feature points by signifying the spatial characteristics in ultrasound images. The model is widely used in ultrasound image analysis as it uses the correlation between the brightness values of neighboring pixels and the current pixel [23], [24]. Additionally, the GLCM method statistically describes the relationship between two adjacent pixels in a specific area, across the area, or the grey level of pixels in a certain distance [25], [26]. The model classification method is a step in training classifiers whose main function is to extract texture features from images [27]. Haralick defined 14 statistical features of the GLCM for texture classification, and these features are highly correlated [28], [29]. Following this discussion, a total of eight features, including energy, entropy, homogeneity,

entropy sum, variance, dissimilarity, and average sum, are extracted from the image of the machine surface using GLCM to form a data set. For texture classification, the model features extracted from the machine surface image are used as inputs to ML classifiers. GLCM is a statistical image analysis method for examining the pixel shape of an image as a grey level matrix, also called a grey level spatial co-occurrence matrix.

Identifying and grouping objects based on features is part of the object recognition process. The extraction stage is the phase where information is gathered from the features for recognition [30], [31]. Moreover, modern CNN classification algorithms can improve diagnostic performance for pathologists outside the capabilities of conventional CAD architectures, as shown by the success in various classification tasks for diagnosing kidney disease [32], [33]. A comprehensive study of CNN has enabled the development of various CNN architectures based on Lenet, Faster R-CNN, ResNet, and VGGs [34]. Additionally, convolution, pooling, activation, and fully connected layers are the most commonly used layers [35]. In a CNN, the merging layer combines the outputs of several groups of neurons located in the same kernel map [36], [37]. Clustering involves dividing a set of data points into groups (clusters) based on their similarities, with each observation acting as a prototype for the cluster [38]. Several performance measures, including accuracy, sensitivity, specificity, confusion matrix, and positive predictive value, were used to assess the performance of the study [39]. ML is the process of training machines to predict diseases using large amounts of data [40], [41].

2 METHODOLOGY

During the study, classification of renal ultrasound images into normal kidney and abnormal kidney images was proposed, where the abnormal kidney was classified into kidney stone, cyst, cancer, and tumor. Figure 1 showed the proposed research framework.

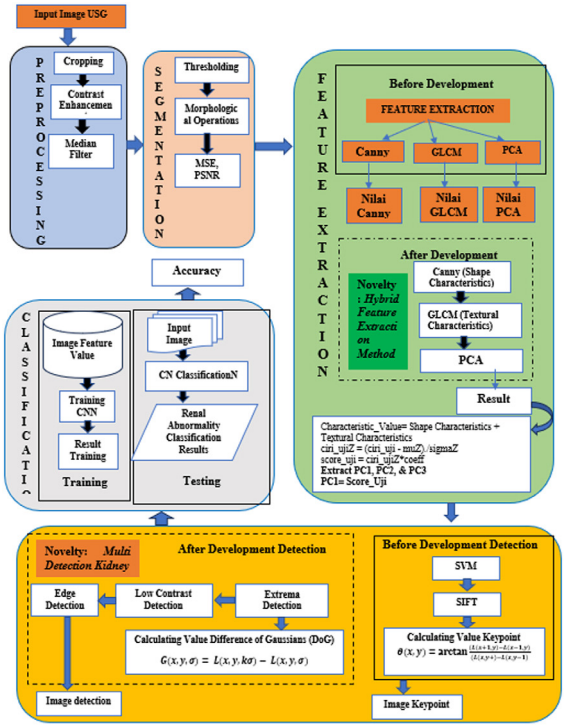


Fig. 1. The proposed research framework

Figure 1 shows the proposed research framework, which consists of several stages. The proposed research framework is the steps that will be taken to solve the problem to be discussed, including:

1. The first is the collection of renal ultrasound image data, which is used as a test image for further analysis.
2. The second stage is the segmentation stage, including thresholding, morphological operations, and calculation of MSE and PSNR values. Previous methods focused on two or three class problems, but the author developed a classification system to detect five categories of kidney images, such as normal, cyst, stone, cancer, and tumor.
3. Previous methods use feature extraction only with Canny or only with GLCM; no one has combined feature extraction with Canny, GLCM, and principal component analysis (PCA) methods. The third stage included the feature extraction stage phase, which represented the contribution (novelty) of the study by developing methods to identify ultrasound images for the detection and classification of kidney abnormalities. In this stage, the value of shape feature extraction using Canny based on area and perimeter was determined. The texture features were also determined using the GLCM method, which included variables such as energy, contrast, correlation, sum of squares, homogeneity, and sum mean. Other parameters included sum variance, sum entropy, entropy, difference variance, difference entropy, information measure of correlation 1, information measure of correlation 2, and maximum correlation coefficient. After obtaining three shape feature values from Canny and 14 texture features from GLCM, the variables were used as feature values for standardization. The features of the kidney image were extracted using PCA followed by a CNN classifier.
4. Following the discussion, the fourth stage was the detection phase, which included several processes, namely extreme, low contrast, and edge detection. The previous detection was SVM, SIFT, and calculating the key point value with a formula so that the key point image output is obtained. While the detection that the author will develop is extreme detection, then low-contrast detection, and finally edge detection, where the output is image detection.
5. The last stage included the classification stage of the kidney ultrasound image, which consisted of the ultrasound image detecting stones, cysts, cancer, and tumors. In the classification stage, CNN was used to train kidney data in the learning program. This process enabled the model to detect kidney image objects by forming layers, applying pooling, and using the dropout method to achieve object detection results according to the objectives of the study. While previous methods focused on two-class or three-class problems, this investigation developed a classification system to detect five categories of kidney images, namely normal, cyst, stone, cancer, and tumor.

3 RESULT

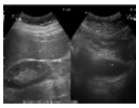
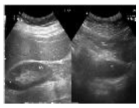
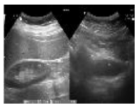
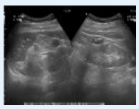
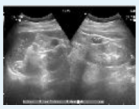
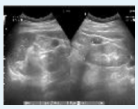
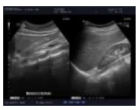
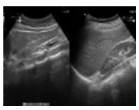
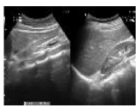
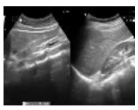
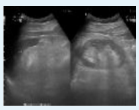
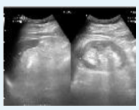
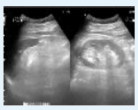
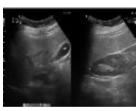
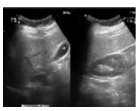
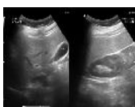
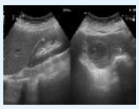
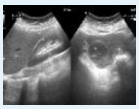
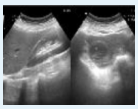
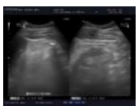
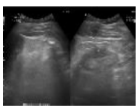
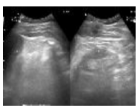
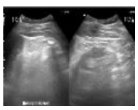
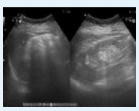
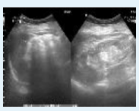
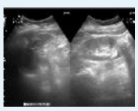
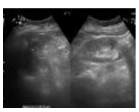
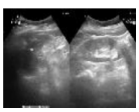
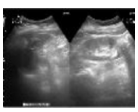
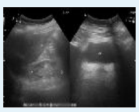
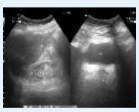
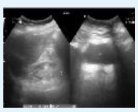
The testing in this study was conducted in several stages according to the study framework using input images in the form of renal ultrasound images. In addition, Matlab R2023 software was used to create programs and test images obtained from renal ultrasound images.

3.1 Image preprocessing

Pre-processing was essential for image development, as poor contrast affected accuracy. The process included hybrid contrast stretching, which used cropping,

contrast improvement, and median filtering. Relating to the discussion, the preprocessing is shown in Table 1.

Table 1. Preprocessing result

Patients	Image Input USG	Image Cropping	Image Contrast Enhancement	Image Median Filter
1				
2				
3				
4				
5				
6				
7				
8				
9				
10				

The preprocessing stage included cropping, which comprised the trimming image to obtain the area of interest and eliminate noise outside the area.

3.2 Image segmentation

Image segmentation aimed to simplify the representation of kidney images, allowing it to be more meaningful and easier to analyze. The segmentation grouped

object pixels into areas that represented specific structures. This stage consisted of Otsu thresholding, morphological operations, and the calculation of MSE and PSNR values. During the process of the examination, a limit called the threshold was used. Image intensity values greater than or equal to the threshold were converted to white (1), while image intensity values less than the threshold were converted to black (0). Therefore, the output of the thresholding result was a binary image, and the outcomes of image segmentation were shown in Table 2 Segmentation of kidney images.

Table 2. Segmentation of kidney images

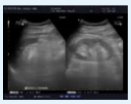
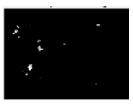
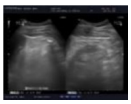
Patients	Image Kidney	Thresholding Otsu	Morphological Operations	MSE	PSNR
1				0.609	15.455
2				0.727	15.592
3				0.684	17.632
4				0.865	18.319
5				0.754	18.680
6				0.981	19.623
7				0.781	18.197
8				0.785	20.107
9				0.829	22.588
10				0.513	16.478

Table 2 shows that as the MSE value reduced, the segmentation results of the renal ultrasound image became better. The results also signified that the closer image was to the original image after processing. Consequently, as the PSNR value increased, the segmented image became closer to the original image.

3.3 Image extraction (CGPCAP)

Feature extraction relied on the local texture property for each image pixel, leading to many statistics. However, studies had developed several hybridization methods to achieve superior results in feature extraction and classification. In the literature, the combination of Canny, GLCM, and PCA methods had not been previously explored. Therefore, this study proposed a new method by combining the three main models for texture feature extraction. Feature extraction relied on the local texture property for each image pixel, leading to many statistics. However, studies had developed several hybridization methods to achieve superior results in feature extraction and classification. A hybrid approach called Canny Grey Principal Component Analysis Pattern (CGPCAP), which combines Canny, GLCM, and PCA to improve the accuracy of feature extraction and increase classification accuracy, outperforms conventional approaches using previous algorithms to extract features from kidney ultrasound images.

Canny edge detection was applied initially to extract two parameters, namely area and perimeter, in this image extraction stage. The Area which was a variable in Canny edge detection, was a measure of the area occupied by an object in the image. This variable was a basic and commonly used shape feature in object recognition. The area provided information about the relative size of objects in the image. Moreover, the perimeter in this investigation was the length of the line that formed the boundary of the object. The variable provided information about the complexity of the object or the sharpness of the edges. In addition, objects with longer perimeters had more complex contours. The results of Canny shape feature values are shown in Table 3.

Table 3. Canny edge result image

Patients	Canny		Image Canny
	Area	Perimeter	
1	26.50	26.03	
2	67.26	61.30	
3	56.71	52.76	
4	46.21	41.80	
5	36.70	34.69	
6	38.35	34.87	

(Continued)

Table 3. Canny edge result image *(Continued)*

Patients	Canny		Image Canny
	Area	Perimeter	
7	55.44	51.07	
8	60.50	54.31	
9	53.69	46.99	
10	69.25	63.02	

Table 3 shows that Canny shape feature extraction produced two feature parameter values, namely area and perimeter. The area was a scalar value that determined the actual number of pixels. Following the discussion, the difference in values was due to variations in the pattern of the pixels. The Canny operator followed a sequence of steps, which started by taking a grayscale image as input and generating an image that showed the position of the intensity discontinuity. Other steps included applying a Gaussian-derived kernel to eliminate noise in the original image and using a hysteresis threshold to distinguish between true and false edges.

The GLCM process was used to characterize the texture of the image by calculating the frequency of pairs of pixels with specific values and spatial relationships. This process led to the formation of GLCM, from which various statistical measures were then extracted. GLCM had several texture features that were extracted from the image and used to distinguish the image of a particular class from other classes. These features included energy, contrast, correlation, sum of squares, homogeneity, and sum average. Other variables included Sum Variance, Sum Entropy, Entropy, Difference Variance, Difference Entropy, Correlation1 Information Measure, Correlation2 Information Measure, and Maximum Correlation Coefficient. Table 4 presents the results of GLCM parameters, namely energy, contrast, correlation, and homogeneity.

Table 4. GLCM result image

No	GLCM				Image GLCM
	Energy	Contrast	Correlation	Homogeneity	
1	0.996	0.002	0.783	0.999	
2	0.961	0.009	0.912	0.997	
3	0.913	0.018	0.920	0.995	

(Continued)

Table 4. GLCM result image *(Continued)*

No	GLCM				Image GLCM
	Energy	Contrast	Correlation	Homogeneity	
4	0.981	0.007	0.878	0.998	
5	0.989	0.008	0.820	0.998	
6	0.976	0.006	0.910	0.998	
7	0.986	0.004	0.916	0.999	
8	0.981	0.005	0.911	0.998	
9	0.981	0.005	0.911	0.998	
10	0.961	0.015	0.867	0.996	

Table 4 showed that the GLCM method was proposed as a statistical method for extracting texture features from images. The method was developed by calculating the transition between pairs of two pixels. Following the discussion, energy, which was a parameter in GLCM, was used to measure the uniformity of the texture. The energy was high when the pixel values were similar, and when the energy was low, it signified that the values of the normalized GLCM were heterogeneous. Moreover, the maximum value of the energy was 1, showing that the distribution of pixels was in a constant state or periodic form (not random). As the image became more homogeneous, the energy value became higher. During the study, contrast measured the spatial frequency of the image and the difference in GLCM moments. This measurement was the difference between the height and depth of a pixel. The contrast was 0 when neighboring pixels had the same value. In this study, homogeneity was called Inverse Difference Moment, which was a measure of the homogeneity of the image. This variable was very sensitive to the values around the main diagonal; it had a high value when all pixels had the same variable. Homogeneity was different from contrast, showing a high value when all pixels had the same value, provided the energy remained constant. Relating to the process, linearity (joint probability) of pairs of pixels was measured. The dimension of the resulting matrix was equal to the maximum value of the original image. In addition, GLCM relied on two parameters, namely distance and angle.

Principal component analysis was a statistical method that aimed to reduce the dimensionality of a dataset while preserving the structure and important information. The method preserved information by projecting high-dimensional data onto a low-dimensional subspace, facilitating the identification of the most important patterns and features in the data. In this study, PCA operated through five main stages, including the first stage, which was known as standardization, where all variables were standardized. The process allowed each variable to have the same contribution in the analysis. The second step was calculating the covariance matrix

to identify the relationship between the variables from the input set. The third stage included calculating the eigenvalues and vectors of the covariance matrix. This study aimed to identify the principal component, as the analysis continued with the feature vectors. By calculating the eigenvalues and vectors, the result showed which components were less significant and could be discarded (components with low eigenvalues). The remaining vector matrix was called the feature vector, and the analysis was then completed by resampling along the principal component axis. Relating to the discussion, Table 5 presents the PCA results of the study.

Table 5. PCA result image

Patients	Characteristics PCA		
	PC1	PC2	PC3
1	–3.778	0.834	0.224
2	2.581	–1.768	–0.465
3	7.360	0.596	0.674
4	–0.452	–0.680	0.083
5	–1.612	0.8239	–0.259
6	0.146	–0.992	1.139
7	–0.689	–2.364	0.525
8	0.044	–2.112	–0.016
9	–0.384	–0.531	–0.398
10	2.968	–0.024	–1.725

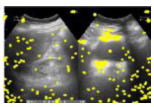
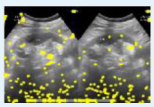
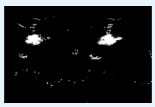
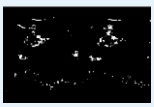
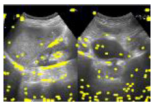
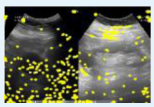
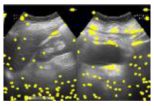
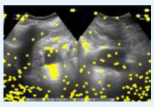
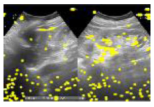
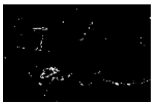
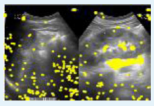
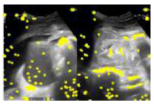
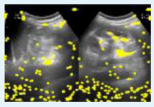
Table 5 showed that PCA captured the essence of the data in a few principal components representing the most variation in the dataset. Firstly, the PCA plot showed sample grouping based on individual similarities. Secondly, the loading plot signified the strength of each characteristic's influence on the principal components. These vectors were pinned to the PC origin (PC1 = 0 and PC2 = 0). The individual projected value in each PC showed how much weight the vectors had. Thirdly, PCA biplot = PCA score plot + loading plot; this PCA biplot simply combined the regular PCA plot with the loading plot. The arrangement included bottom an axis representing the PC1 score, a left axis for the PC2 score, a top axis showing the load on PC1, and a right axis comprising the load on PC2. Fourthly, the scree plot showed how much variation each principal component captured from the data. Consequently, the scree plot was a diagnostic tool to check how effectively PCA represented the data. The comparison results of PC1, PC2, and PC3 values in PCA were shown in the loading plot, signifying the extent to which each characteristic influenced the principal component.

3.4 Image detection

The detection stage comprised several stages, including extreme, low contrast, and edge detection. In extreme detection, maximum or minimum points were identified in the difference of Gaussian space. After obtaining the extrema points, the positions were refined with sub-pixel accuracy using the second-order Taylor expansion of the scale space function to obtain the true extremum position. Moreover,

the low contrast image had minimal difference between the light and dark parts, allowing it to be difficult to distinguish where the object boundary began and the scene background as well. Images that were inadequate for image processing flow were programmatically detected using low-contrast image detection. Following this discussion, edge tracking was an operation used to identify distinct local intensity changes in the image. Gradient was produced by measuring changes in this intensity function, and the image was viewed as a collection of continuous intensity patterns. Relating to the discussion, Table 6 showed the detection result image.

Table 6. Detection result image

Patients	Local Extreme Detection	Low Contrast Detection	Result Edge Detection
1			
2			
3			
4			
5			
6			
7			
8			
9			
10			

According to Table 6, an edge was described as a rapid or significant change in the gray level intensity over a short distance. Edge detection in the image was

used to identify and mark the edges to show image details. This method helped to clarify blurred image detail, which came from errors or limitations during image acquisition. A point (x, y) was said to be an edge of the image when the point had a high difference from its neighbors. Following the discussion, an edge detection operator was a tool used to modify the grey value of a point based on the grey value of the points around it (convolution/neighborhood operation). In the neighborhood operation, points were assigned weights based on the specific operation being performed. Moreover, the number of points used varied, with configurations such as 2×2, 3×3, 5×5, 7×7, and others.

3.5 Image classification

When feature extraction focused on mapping patterns based on the features of the image, classification aimed to the identify image by classifying these features. The classification was used as the process of finding a model or function that described or distinguished concepts or classes of data. This process aimed to estimate the class of an object whose label was unknown. Classification process was divided into two phases, including the learning and testing phases. In the learning phase, a portion of the data of a known class was fed to form an approximate model, while in the test stage, the model that had been formed was tested with some other data to determine the accuracy. When the accuracy was sufficient, the model was used to predict the class of the unknown data. This method classified new data by using previously categorized data to calculate the distance between the features of the template image and the input image.

4 DISCUSSION

The purpose of using CNN in measuring accuracy to evaluate the performance of the CGPCAP result model from a renal ultrasound image.

4.1 Architecture of CNN

The architecture of CNN for detecting kidney abnormalities used a kernel (or filter) specifically designed to analyze kidney images. In addition, the filter was a randomly generated vector in the network consisting of weights and biases. The same weights and biases were shared between the different neurons in CNN, rather than unique weights and biases for each neuron. Many filters were generated, with each filter capturing an exceptional feature of the input. After the convolution step during this study, the ReLU activation function was applied. ReLU was the most commonly used activation function that allowed neural networks to model non-linear relationships effectively. In a specified matrix (x), ReLU set all negative values to zero and kept all other values constant. The CNN architecture consists of eight layers, including five convolutional layers and five fully connected layers. The features presented in the ultrasound kidney image were extracted using the pre-trained AlexNet CNN, which consisted of an input value of 16, including eight convolutional layers and five fully connected (FC) layers. Following this discussion, convolutional, pooling, and FC layers were the five main layers in CNN architecture, as shown in Figure 2.

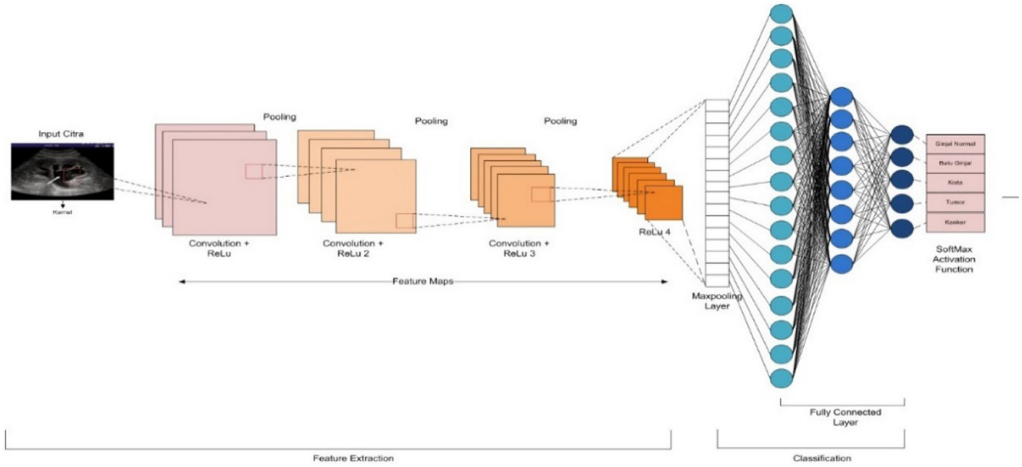


Fig. 2. Architecture of CNN for detecting kidney abnormalities

Figure 2 showed detection and classification of kidney abnormalities with 5 classes, namely normal, stone, cyst, cancer, and tumor. The resulting shape in the CNN architecture is shown in Figure 3.

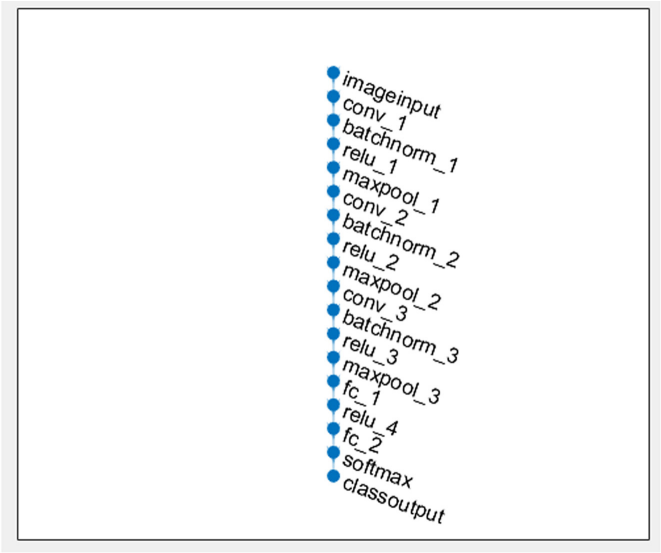


Fig. 3. Architecture CNN

Figure 3 CNN Architecture explains that the proposal of a hybrid approach called CGPCAP, which combines Canny, GLCM, and PCA to improve the accuracy of feature extraction and increase the classification accuracy, provides better results than conventional approaches that use previous algorithms to extract features from kidney ultrasound images.

4.2 Testing detection results with CNN

During the study, the image was grouped according to the file used. The purpose of using CNN in this investigation was to develop a system that automatically identified patterns or special features in kidney images to detect certain medical conditions. Following the process, the features analyzed included energy, contrast,

correlation, and homogeneity. The main purpose of CNN was to perform the task of feature and pattern recognition to classify kidney abnormalities. This process was achieved through the convolutional layer, which enabled the network to effectively extract important features from the image, supporting accurate classification based on the extracted and detected features. In the classification stage, the results of the selected renal ultrasound image were shown in Table 7.

Table 7. Classification result image of patient 1 to 10

Feature Value	Patients									
	1	2	3	4	5	6	7	8	9	10
Area	26.500	67.266	56.714	46.210	36.705	38.350	55.444	60.500	53.692	69.259
Perimeter	26.039	61.305	52.764	41.808	34.691	34.878	51.076	54.315	46.997	63.027
Energy	0.996	0.961	0.913	0.981	0.989	0.976	0.986	0.981	0.985	0.961
Contrast	0.002	0.009	0.018	0.007	0.008	0.006	0.004	0.005	0.009	0.015
Correlation	0.783	0.912	0.920	0.878	0.820	0.91	0.916	0.911	0.862	0.867
Sum of squares	0.979	1.086	1.230	1.029	1.014	1.049	1.019	1.031	1.030	1.090
Homogeneity	0.999	0.997	0.995	0.998	0.998	0.998	0.999	0.998	0.998	0.996
Sum average	2.005	2.062	2.139	2.030	2.021	2.041	2.025	2.032	2.028	2.063
Sum variance	3.982	4.005	4.133	3.999	4.010	4.019	4.021	4.008	4.027	4.013
Sum entropy	0.014	0.1142	0.222	0.059	0.040	0.073	0.044	0.059	0.052	0.114
Entropy	0.015	0.1209	0.236	0.062	0.042	0.077	0.044	0.062	0.055	0.122
D. Variance	0.002	0.009	0.018	0.007	0.008	0.006	0.004	0.005	0.009	0.015
D. Entropy	0.005	0.0267	0.046	0.0171	0.016	0.017	0.008	0.014	0.017	0.035
IMC 1	–0.687	–0.8352	–0.8397	–0.7744	–0.709	–0.825	–0.8357	–0.818	–0.764	–0.755
IMC 2	0.127	0.398	0.538	0.27	0.212	0.320	0.249	0.287	0.256	0.372
MCC	0.998	0.980	0.955	0.990	0.994	0.988	0.993	0.990	0.992	0.9803
Results	**Cysts**	**Normal**	**Normal**	**Cysts**	**Normal**	**Cysts**	**Tumors**	**Cysts**	**Tumors**	**Tumors**

Table 7 showed that classification of renal ultrasound images was divided into five classes, namely normal kidney, stone, cyst, cancer, and tumor. In addition, classification results were acquired from the results of the data stored in net.mat. Following the discussion, CNN model training results are shown in Table 8.

Table 8. CNN model training results

Epoch	Iteration	Time Elapsed	Mini-Batch Accuracy	Mini-Batch Loss	Base Learning
1	1	00:00:01	28.12%	1.5824	0.0010
100	500	00:00:14	46.88%	1.1656	0.0010
200	1000	00:00:24	65.62%	0.9026	0.0010
300	1500	00:00:40	68.75%	0.7076	0.0010
400	2000	00:00:55	68.75%	0.7442	0.0010
500	2500	00:01:09	81.25%	0.4817	0.0010
580	2900	00:01:20	96.88%	0.2870	0.0010
990	4950	00:02:19	90.62%	0.3089	0.0010
1000	5000	00:02:20	97.5%	0.2953	0.0010

Table 8 reveals that the model experienced a significant performance improvement from the beginning to the end of training, with the validation accuracy continuously increasing and the validation loss continuously decreasing, indicating that the model is getting better at generalizing the validation data. The learning rate remains constant throughout the training process at 0.0010, indicating that a small learning rate is used to achieve optimal convergence.

The Training Progress Monitor was a tool used to track the progress of training when using custom training loops. This study used the progress monitor object to animate custom and record custom metrics during training. The tool also showed and recorded training information, stopped training early, and tracked training progress along with elapsed time. More information on configuring the training progress window and an example of how to generate this image was shown in monitoring custom training loop progress. Following the discussion, CNN classification started with training progress as shown in Figure 4.

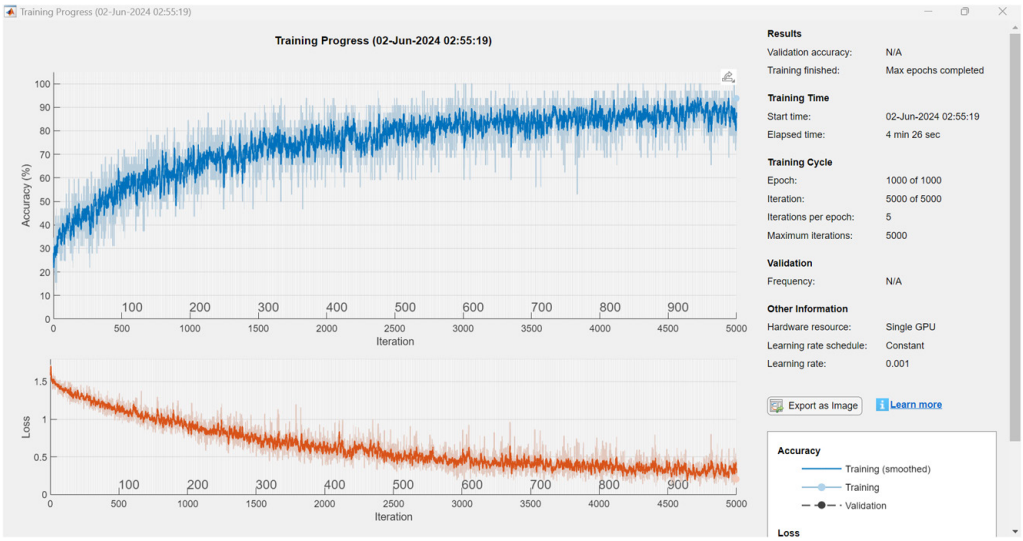


Fig. 4. CNN model training result graph

Figure 4 showed the results of CNN progress, specifically signifying the validation accuracy in the training command window. The training progress monitor in MATLAB was an object used to track the progress of training and to plot the training. Additionally, training accuracy referred to the accuracy obtained when a model was applied to training data. When the model was tested, the training data was used to generate predictions based on the input features. These predictions were then compared to the original labels to measure accuracy. The representation of the training progress graph consisted of several parts, including Epoch, which was the number of times the model passed through all the training data. During the study, iteration represented the number of times the model had updated its parameters or weights based on a portion of the training data. Increasing the number of iterations allowed the model to learn more quickly. However, iterations also raised the risk of overshooting the optimal solution, potentially leading to instability. The mini-batch size in the study was expressed as the amount of data used in the iteration, which affected the speed and quality of model learning. Base Learning Rate described the size of the large number of times the model changed its parameters or weights in an iteration, as the rate affected the speed and quality of model learning. The results obtained in the training process signified that the accuracy value was 95.625%, as shown in Figure 5.

```
Training finished: Max epochs completed.
====== PELATIHAN ======
Jumlah Data  : 160
Presisi      : 93.75 %
Recall       : 93.75 %
F1 Score     : 93.75 %
Akurasi      : 97.5 %
Sensitivitas : 93.75 %
Spesifitas   : 98.4375 %
```

Fig. 5. Training accuracy

Figure 5 showed the training accuracy associated with the development of ultrasound extraction for kidney detection and abnormalities, signifying a validation accuracy of 97.5%. The training data was used to train CNN. As the training data of a model increased, the model made better predictions. During the study, the test data was used to evaluate the performance of the model, which learned from the training data and improved its predictions. Training finished is Max epochs completed. Total data: 160, accuracy: 93.75, recall: 93.75%, F1 score: 93.75%, accuracy: 97.5, sensitivity: 93.75, and specificity: 98.4375.

In order to measure the performance of a classification model, an evaluation metric for determining accuracy is required. Accuracy is defined as the ratio of the number of correct predictions to the total number of predictions made by the model. Correct predictions include two categories: true positive (TP) and true negative (TN). TP refers to true positive predictions, while TN refers to true negative predictions. On the other hand, there are two types of incorrect predictions, namely false positive (FP) and false negative (FN), where FP is when the model incorrectly predicts a positive for a class that should be negative, and FN is when the model incorrectly predicts a negative for a class that should be positive.

$$Akurasi = \frac{TP + TN}{TP + TN + FP + FN} \times 100\% \quad (1)$$

The process is also presented in the form of a confusion matrix, precision, recall and F1 score as a more comprehensive form of evaluation. The prediction results are analyzed using the Confusion Matrix, which provides a detailed overview of the correct and incorrect predictions for each class labeled on each test data. Based on this confusion matrix, important metrics such as recall (the model's ability to detect all positive classes), precision (the model's accuracy in predicting positive classes), and F1 score (the harmonic mean between recall and precision) are calculated to provide a more accurate evaluation of the model's performance, especially on the dataset used in this study, which consists of 160 training images and 40 test images, each labelled according to the conditions of normal kidney, kidney stone, kidney cyst, kidney tumor, and kidney cancer.

4.3 Confusion matrix

The process is also presented in the form of a confusion matrix, precision, recall and F1 score as a more comprehensive form of evaluation. The prediction results are analyzed using the Confusion Matrix, which provides a detailed overview of the correct and incorrect predictions for each class labeled on each test data. Based on this confusion matrix, important metrics such as recall, precision, and F1 score are calculated to provide a more accurate evaluation of the model's performance, especially on the dataset used in this study, which consists of 160 training images

and 40 test images labeled according to the conditions of normal kidney, kidney cyst, kidney stone, kidney tumor and kidney cancer, as shown in Figure 6.

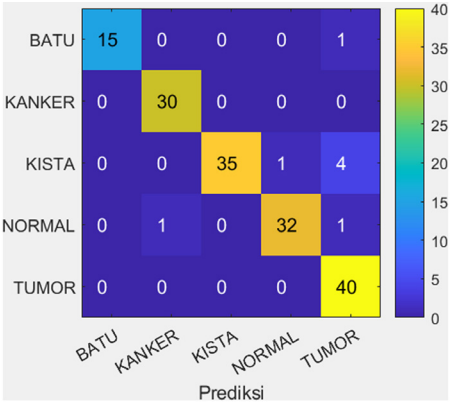


Fig. 6. Confusion matrix

Figure 6 shows the confusion matrix of the results of the five image classes tested, such as normal kidney, kidney stone, kidney cyst, kidney tumor and kidney cancer. The confusion matrix provides an overview of the performance of the model in distinguishing the five classes.

4.4 Receiver operating characteristic

The receiver operating characteristic (ROC) graph shows the relationship between True Positive Rate (TPR) and False Positive Rate (FPR) at different class thresholds. TPR, also known as sensitivity or recall, describes the proportion of positives that are correctly classified, while FPR describes the proportion of negatives that are incorrectly positioned as positives. The calculation process performed in ROC is to determine the TPR value. This process is performed to measure how well the model correctly identifies positive samples. TPR (recall) or sensitivity is expressed using equation 2.

$$TPR = \frac{TP}{TP + FN} \tag{2}$$

The results of the model testing process based on predetermined classes on the training and test data by visually displaying the ROC in Figure 7.

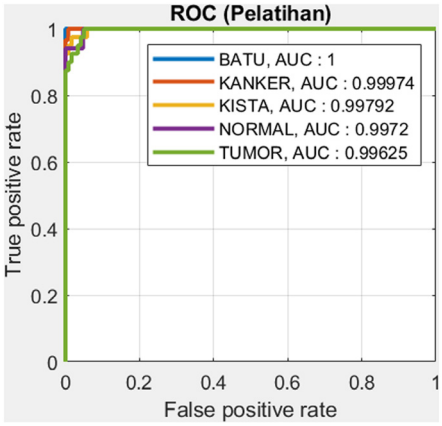


Fig. 7. ROC

Figure 7 shows ROC; based on the results obtained, each class is correctly identified, and there is no overlap between classes. This shows that the CNN model used has optimal performance in the process of predicting images in each of the existing classes, as shown in Figure 8.

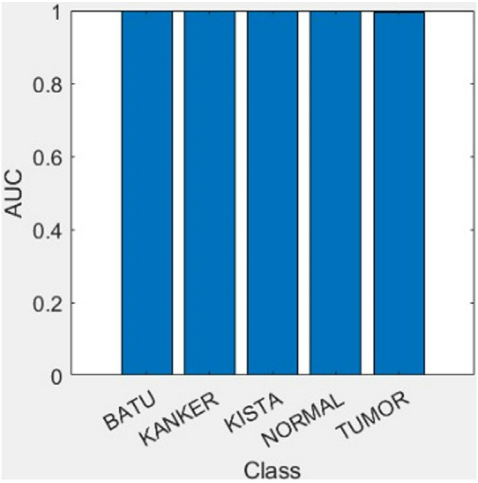


Fig. 8. AUC for each class

Figure 8 shows the AUC (area under the curve) values for the five classes of normal kidney, kidney stone, kidney cyst, kidney tumor, and kidney cancer as a result of the analysis using the ROC curve. This high AUC is a strong indication that the model has good generalizability and can handle the variability in the data very well.

5 CONCLUSION

In conclusion, automated classification methods were very useful for kidney abnormalities detection. The use of pre-trained CNN networks performed well for feature extraction on small data sets. Moreover, automatic feature extraction with Canny, GLCM, SIFT, and PCA methods led to a higher classification rate with an accuracy of 97.5%. This suggests that the CGPCAP algorithm not only improves the model's ability to make a more accurate prediction but is also more efficient at handling more complex data. Overall, the use of CGPCAP with the CNN test algorithm provides better results in the context of renal abnormality detection.

6 REFERENCES

[1] P. Kokil and S. Sudharson, "Computer-aided diagnosis system for the classification of multi-class kidney abnormalities in the noisy ultrasound images," *Computer Methods and Programs Biomedicine*, vol. 205, p. 106071, 2021. https://doi.org/10.1016/j.cmpb.2021.106071

[2] P. Kokil and S. Sudharson, "An ensemble of deep neural networks for kidney ultrasound image classification," *Comput. Methods Programs Biomed.*, vol. 197, p. 105709, 2020. https://doi.org/10.1016/j.cmpb.2020.105709

[3] D. C. H. Harris, S. J. Davies, F. O. Finkelstein, C. W. Yang, and C. Zuniga, "Increasing access to integrated ESKD care as part of universal health coverage," *Kidney Int.*, vol. 95, no. 4, pp. S1–S33, 2019. https://doi.org/10.1016/j.kint.2018.12.005

[4] S. D. Mahalakshmi, "An optimized transfer learning model based kidney stone classification," *Comput. Syst. Sci. Eng.*, vol. 44, no. 2, pp. 1387–1395, 2023. https://doi.org/10.32604/csse.2023.027610

[5] E. A. Hartung and K. M. Dell, "Approach to simple kidney cysts in children," *Pediatr. Nephrol.*, vol. 39, pp. 3389–3395, 2024. https://doi.org/10.1007/s00467-024-06386-6

[6] A. Yogeswaran *et al.*, "Comparison of contemporary risk scores in all groups of pulmonary hypertension," *Pulmonary Vascular: Original Research*, vol. 166, no. 3, pp. 585–603, 2024. https://doi.org/10.1016/j.chest.2024.03.018

[7] V. Malkhasyan *et al.*, "Influence of a simple cyst on kidney function," *Front. Med.*, vol. 11, pp. 1–6, 2024. https://doi.org/10.3389/fmed.2024.1381942

[8] G. Bueno, M. M. Fernandez-Carrobles, L. Gonzalez-Lopez, and O. Deniz, "Glomerulosclerosis identification in whole slide images using semantic segmentation," *Comput. Methods Programs Biomed.*, vol. 184, p. 105273, 2020. https://doi.org/10.1016/j.cmpb.2019.105273

[9] D. Abimanyu, S. Sumarno, F. Anggraini, I. Gunawan, and I. Parlina, "Rancang Bangun Alat Pemantau Kadar pH, Suhu Dan Warna Pada Air Sungai Berbasis Mikrokontroller Arduino," *J. Pendidik. dan Teknol. Indones.*, vol. 1, no. 6, pp. 235–242, 2021. https://doi.org/10.52436/1.jpti.55

[10] T. Borges, A. Rai, D. Raj, D. Ather, and K. Gupta, "Kidney stone detection using ultrasound images," *SSRN Electron. J.*, pp. 1–8, 2022. https://doi.org/10.2139/ssrn.4159208

[11] N. Razmjooy and C. Yan, "Kidney stone detection using an optimized deep believe network by fractional coronavirus herd immunity optimizer," *Biomed. Signal Process. Control*, vol. 86, no. PA, p. 104951, 2023. https://doi.org/10.1016/j.bspc.2023.104951

[12] Y. Fu *et al.*, "A novel MRI segmentation method using CNN-based correction network for MRI-guided adaptive radiotherapy," *Med. Phys.*, vol. 45, no. 11, pp. 5129–5137, 2018. https://doi.org/10.1002/mp.13221

[13] M. N. Islam *et al.*, "Vision transformer and explainable transfer learning models for auto detection of kidney cyst, stone and tumor from CT-radiography," *Sci. Rep.*, vol. 12, 2022. https://doi.org/10.1038/s41598-022-15634-4

[14] S. Biradar and P. T. Akkasaligar, "Classification of medical ultrasound images of kidney," in *Int. Conf. Inf. Commun. Technol.*, 2014, pp. 24–28.

[15] K. Ansarin *et al.*, "Are Doppler ultrasonography parameters symmetric between the right and left kidney?" *In. J. Gen. Med*, vol. 3, pp. 371–373, 2011. https://doi.org/10.2147/IJGM.S14119

[16] M. Radhakrishnan, A. Panneerselvam, and N. Nachimuthu, "Canny edge detection model in MRI image segmentation using optimized parameter tuning method," *Intell. Autom. Soft. Comput.*, vol. 26, no. 6, pp. 1185–1199, 2020. https://doi.org/10.32604/iasc.2020.012069

[17] A. M. Athavale *et al.*, "Development and validation of a deep learning model to quantify interstitial fibrosis and tubular atrophy from kidney ultrasonography images," *JAMA Netw. Open*, vol. 4, no. 5, pp. 1–12, 2021. https://doi.org/10.1001/jamanetworkopen.2021.11176

[18] S. B. Basnet and P. K. Chhetri, "Ultrasonography in patients with chronic kidney disease," *J. Chitwan Med. Coll.*, vol. 11, no. 2, pp. 110–114, 2021. https://doi.org/10.54530/jcmc.342

[19] S. Barburiceanu, R. Terebes, and S. Meza, "3D texture feature extraction and classification using Glcm and Lbp-based descriptors," *Appl. Sci.*, vol. 11, no. 5, pp. 1–26, 2021. https://doi.org/10.3390/app11052332

[20] N. D. Lynn, A. I. Sourav, and A. J. Santoso, "Implementation of real-time edge detection using Canny and Sobel algorithms," *IOP Conf. Ser. Mater. Sci. Eng.*, vol. 1096, p. 12079, 2021. https://doi.org/10.1088/1757-899X/1096/1/012079

[21] E. A. Sekehravani, E. Babulak, and M. Masoodi, "Implementing Canny edge detection algorithm for noisy image," *Bull. Electr. Eng. Informatics*, vol. 9, no. 4, pp. 1404–1410, 2020. https://doi.org/10.11591/eei.v9i4.1837

[22] F. Chen and X. Zhang, "Lane line edge detection based on improved adaptive Canny algorithm," *J. Phys. Conf. Ser.*, vol. 1549, no. 2, 2020. https://doi.org/10.1088/1742-6596/1549/2/022131

[23] F. H. Najjar, H. M. Al-Jawahry, M. S. Al-Khaffaf, and A. T. Al-Hasani, "A novel hybrid feature extraction method using LTP, TFCM, and GLCM," *J. Phys. Conf. Ser.*, vol. 1892, no. 1, 2021. https://doi.org/10.1088/1742-6596/1892/1/012018

[24] S. Y. Ye and D. H. Kim, "Classification of chronic kidney disease in sonography using the glcm and artificial neural network," *Diagnostics*, vol. 11, no. 5, p. 854, 2021. https://doi.org/10.3390/diagnostics11050864

[25] S. A. Althubiti, S. Paul, R. Mohanty, S. N. Mohanty, F. Alenezi, and K. Polat, "Ensemble learning framework with GLCM texture extraction for early detection of lung cancer on CT images," *Comput. Math. Methods Med.*, vol. 2022, no. 1, p. 2733965, 2022. https://doi.org/10.1155/2022/2733965

[26] G. Prasad *et al.*, "A study of dimensionality reduction in GLCM feature-based classification of machined surface images," *Arab. J. Sci. Eng.*, vol. 49, no. 2, pp. 1531–1553, 2024. https://doi.org/10.1007/s13369-023-07854-1

[27] S. A. Althubiti, S. Paul, R. Mohanty, S. N. Mohanty, F. Alenezi, and K. Polat, "Ensemble learning framework with GLCM texture extraction for early detection of lung cancer on CT images," *Computational and Mathematical Methods in Medicine*, vol. 2022, pp. 1–14, 2022. https://doi.org/10.1155/2022/2733965

[28] M. Benco, R. Hudec, P. Kamencay, M. Zachariasova, and S. Matuskal, "An advanced approach to extraction of colour texture features based on GLCM," *Int. J. Adv. Robot. Syst.*, vol. 11, no. 1, 2014. https://doi.org/10.5772/58692

[29] A. S. Vironicka and J. G. R. Sathiaseelan, "Texture feature extraction with medical image using GLCM and machine learning techniques," *ICTACT Journal on Image and Video Processing*, vol. 12, no. 4, pp. 2735–2740, 2022. https://ictactjournals.in/paper/IJIVP_Vol_12_Iss_4_Paper_8_2735_2740.pdf

[30] F. T. Kurniati, "GLCM-based feature combination for extraction model optimization in object detection using machine learning," *J. Ilm. Tek. Elektro Komput. dan Inform.*, vol. 9, no. 4, pp. 1196–1205, 2023. https://doi.org/10.26555/jiteki.v9i4.27842

[31] P. A. Cicalese, A. Mobiny, Z. Shahmoradi, X. Yi, C. Mohan, and H. V. Nguyen, "Kidney level lupus nephritis classification using uncertainty guided Bayesian convolutional neural networks," *IEEE J. Biomed. Heal. Informatics*, vol. 25, no. 2, pp. 315–324, 2021. https://doi.org/10.1109/JBHI.2020.3039162

[32] C. Sabanayagam *et al.*, "A deep learning algorithm to detect chronic kidney disease from retinal photographs in community-based populations," *The Lancet Digit. Heal*, vol. 2, no. 6, pp. e295–e302, 2020. https://doi.org/10.1016/S2589-7500(20)30063-7

[33] P. Wang, X. Zhang, and Y. Hao, "A method combining CNN and ELM for feature extraction and classification of SAR image," *J. Sensors*, vol. 2019, no. 1, p. 6134610, 2019. https://doi.org/10.1155/2019/6134610

[34] R. S. Alharbi, H. A. Alsaadi, S. Manimurugan, T. Anitha, and M. Dejene, "Multiclass classification for detection of COVID-19 infection in chest X-rays using CNN," *Comput. Intell. Neurosci.*, vol. 2022, no. 1, pp. 1–11, 2022. https://doi.org/10.1155/2022/3289809

[35] Y. Eroglu, K. Yildirim, A. Çinar, and D. M. Yildirim, "Diagnosis and grading of vesicoureteral reflux on voiding cystourethrography images in children using a deep hybrid model," *Comput. Methods Programs Biomed.*, vol. 210, p. 106369, 2021. https://doi.org/10.1016/j.cmpb.2021.106369

[36] M. M. Al-Nawashi, O. M. Al-Hazaimeh, and M. K. Khazaaleh, "A new approach for breast cancer detection-based machine learning technique," *Appl. Comput. Sci.*, vol. 20, no. 1, pp. 1–16, 2024. https://doi.org/10.35784/acs-2024-01

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

[38] M. K. Khazaaleh *et al.*, "Handling DNA malfunctions by unsupervised machine learning model," *J. Pathol. Inform.*, vol. 14, p. 100340, 2023. https://doi.org/10.1016/j.jpi.2023.100340

[39] N. Gharaibeh, A. A. Abu-ein, O. M. Al-hazaimeh, K. M. O. Nahar, W. A. Abu-ain, and M. M. Al-nawashi, "Swin transformer-based segmentation and multi-scale feature pyramid fusion module for Alzheimer's disease with machine learning," *International Journal of Online and Biomedical Engineering (iJOE)*, vol. 19, no. 4, pp. 22–50, 2023. https://doi.org/10.3991/ijoe.v19i04.37677

[40] E. M. Rojas, H. B. R. Moreno, M. R. Ramirez, and J. S. M. Palencia, "Contributions of machine learning in the health area as support in the diagnosis and care of chronic diseases," in *Innovation in Medicine and Healthcare Systems, and Multimedia, Smart Innovation, System and Technologies*, Y. W. Chen, A. Zimmermann, R. Howlett, and L. Jain, Eds., Springer Singapore, 2019, vol. 145, pp. 291–269. https://doi.org/10.1007/978-981-13-8566-7_26

[41] D. Carlos, D. Erny Herwindiati, and C. Lubis, "Implementasi Algoritma Convolutional Neural Networks Untuk Klasifikasi Jenis Cat Tembok Menggunakan Arsitektur MobileNet," *Technol. Sci.*, vol. 6, no. 1, pp. 395–402, 2024. https://doi.org/10.47065/bits.v6i1.5322

7 AUTHORS

Ruri Hartika Zain is a Lecturer at the Faculty of Computer Science, YPTK, Putra Indonesia University. She earned her Bachelor of Computer Systems and Master of Information Technology degrees in 2004 and 2009 from Putra Indonesia University YPTK (E-mail: rurihartika_zain@upiyptk.ac.id).

Sumijan is a Lecturer at the Faculty of Computer Science in the Information Technology Program. He obtained a bachelor's degree at Putra Indonesia University YPTK. He did M.Sc. from Universiti Teknologi Malaysia and a Doctor of Information Technology from Gunadarma University in 2015. His work focuses on image processing. He can be contacted at email: sumijan@upiyptk.ac.id.

Sarjon Defit is the Chancellor of Putra Indonesia University YPTK Padang. Currently active as a Lecturer in Computer Science and is interested in data mining, artificial intelligence, decision support systems, and others. He can be contacted at email: sarjon_defit@upiyptk.ac.id.