Amrita School of Artificial Intelligence, Coimbatore
Amrita Vishwa Vidyapeetham, India

Kidney Disease Classification using Deep Learning Architectures

Prepared By:
Batch – A – Group - 14
M.D.S. Rama Saran - [CB.EN.U4AIE21034]
Akshayaa BK - [CB.EN.U4AIE21002]
R. Sai Raghavendra - [CB.EN.U4AIE21049]
Poornima N - [CB.EN.U4AIE21042]

Supervised by:
Dr. Abhijith M S

An End Semester Project Report submitted to the Amrita School of Artificial Intelligence as a part of course evaluations of **Mathematics for Intelligent Systems – 6** for B. Tech in **Computer Science Engineering** – **Artificial Intelligence.**

ACKNOWLEDGMENT

The satisfaction and euphoria that accompany the successful completion of any task would be incomplete without mentioning about the people whose constant guidance and encouragement made it possible. We take pleasure in presenting before you, our project, which is result of studied blend of both research and knowledge.

We express our earnest gratitude to our internal guide, Dr Abhijith M S, Department of Artificial Intelligence, our project guide, for his constant support, encouragement and guidance. We are grateful for his cooperation and his valuable suggestions.

Finally, we would also like to thank our group mates for the constant encouragement and the team work which was put throughout the project. We could learn about the various concepts in the project with the each other's help and support.

DECLARATION

We the undersigned solemnly declare that the project report is based on our own work carried out during the course of our study under the supervision of Dr. Abhijith M S, Department of Artificial Intelligence. We assert the statements made and conclusions drawn are an outcome of our research work. We further certify that

I. The work contained in the report is original and has been done by us under the general supervision of our supervisor.

II. We have followed the guidelines provided by the university in writing the report.

III. Whenever we have used materials (data, theoretical analysis, and text) from other sources, we have given due credit to them in the text of the report and giving their details in the references.

Mummidi Devi Siva Rama Saran

Akshayaa B K

R Sai Raghavendra

Poornima N

TABLE OF CONTENTS

ABSTRACT

Kidney diseases, such as cysts, stones, and tumors, present significant diagnostic challenges and can lead to severe health consequences if not identified and treated promptly. The advent of deep learning has opened new avenues for automating medical image analysis, offering potential improvements in diagnostic accuracy and efficiency. This study explores the application of various deep learning models, specifically Convolutional Neural Networks (CNN), Squeeze Net, and ResNet-18, for the classification of kidney diseases using CT scan images. The dataset comprises labeled images of normal kidneys, cysts, stones, and tumors, which are preprocessed to enhance model performance. Cross-entropy loss is employed as the loss function during training. The effectiveness of the models is evaluated, with ResNet-18 demonstrating superior performance in accurately classifying the different types of kidney conditions. The success of ResNet-18 underscores its potential for practical application in medical image analysis. Our findings highlight the significant potential of deep learning models in improving the diagnostic process for kidney diseases, thereby enhancing patient outcomes and reducing the burden on healthcare professionals.

Signature of the Guide
Name: Dr. Abhijith M S

INTRODUCTION

Kidney diseases, such as cysts, stones, and tumors, pose significant diagnostic challenges and can lead to severe health consequences if not detected and treated promptly. Traditional diagnostic methods, including ultrasound and CT scans, depend heavily on the expertise of radiologists, resulting in variability in interpretations and potential delays in diagnosis. With the increasing availability of medical imaging data and advancements in computational power, deep learning has emerged as a promising tool to automate and enhance the diagnostic process. Deep learning, a subset of machine learning, leverages neural networks with multiple layers to model complex patterns in data. In medical imaging, deep learning models, particularly Convolutional Neural Networks (CNNs), have shown remarkable success in tasks such as image classification, segmentation, and anomaly detection. These models can learn to identify intricate features in medical images, offering a consistent and objective approach to diagnosis.

This study focuses on the application of deep learning models to classify kidney diseases using CT scan images. We explore three neural network architectures: Convolutional Neural Networks (CNN), Squeeze Net, and ResNet-18. These models are selected for their varying complexity and efficiency, providing a comprehensive evaluation of their performance in this specific medical imaging task. The methodology involves preprocessing a labeled dataset of CT kidney images, which includes categories of normal kidneys, cysts, stones, and tumors. Data preprocessing steps such as normalization, augmentation, and resizing are applied to ensure the quality and consistency of the input data. The models are trained using cross-entropy loss, a standard loss function for multi-class classification, to optimize their predictive capabilities. Through this approach, we aim to demonstrate the feasibility and effectiveness of deep learning models in automating the classification of kidney diseases. The findings from this study have the potential to enhance diagnostic accuracy and efficiency, ultimately improving patient outcomes and reducing the workload on radiologists. The results indicate that deep learning models, particularly ResNet-18, hold promise as reliable tools for kidney disease classification, paving the way for their integration into clinical practice.

MOTIVATION OF OUR PROJECT

The early and accurate detection of kidney diseases, such as cysts, stones, and tumors, is crucial for effective treatment and management, potentially preventing severe health complications. Traditional diagnostic methods rely heavily on manual interpretation of medical images by radiologists, which can be time-consuming, prone to human error, and dependent on the availability of highly trained specialists. Given the growing prevalence of kidney diseases worldwide, there is an urgent need for automated, reliable, and efficient diagnostic tools to assist healthcare professionals. Deep learning, particularly Convolutional Neural Networks (CNNs), has demonstrated remarkable success in various image classification tasks, offering an innovative solution to automate the analysis of medical images.

Our project is motivated by the potential of deep learning to revolutionize medical diagnostics, particularly in resource-constrained settings where access to experienced radiologists is limited. By leveraging advanced model architectures such as ResNet-18, SqueezeNet, and traditional CNNs, we aim to develop a robust system capable of identifying and classifying kidney diseases with high precision. Automating the detection process promises to enhance diagnostic accuracy and efficiency, reducing the burden on radiologists and enabling timely interventions. This project contributes to the broader field of medical image analysis, advancing the application of machine learning technologies in healthcare, and ultimately improving patient outcomes and reducing healthcare costs.

Dataset Description and Preparation

The dataset consists of medical imaging data collected from Picture Archiving and Communication Systems (PACS) at various hospitals in Dhaka, Bangladesh. It focuses on four specific radiological findings: kidney tumor, cyst, normal, and stone. The data collection involved patients who had already been diagnosed with these conditions. For each patient, both coronal and axial cuts were selected, including both contrast and non-contrast studies, following protocols for the whole abdomen and urogram. In the preparation phase, the Dicom studies were carefully selected one diagnosis at a time to ensure precise targeting of the region of interest for each radiological finding. This step ensured that only relevant images were included in the dataset. Patient information and metadata were then excluded from the Dicom images to protect privacy. Subsequently, the Dicom images were converted to a lossless JPG format to facilitate easier handling and use in machine learning models. After conversion, each image was verified by a radiologist and a medical technologist to reconfirm the correctness of the data, ensuring high-quality and reliable dataset content.

The dataset comprises a total of 12,446 unique images. These images are categorized into four groups based on the radiological findings: cyst, normal, stone, and tumor. This distribution provides a diverse set of examples for each condition, which is beneficial for training robust machine learning models.

Dataset Description of Kidney Classes:

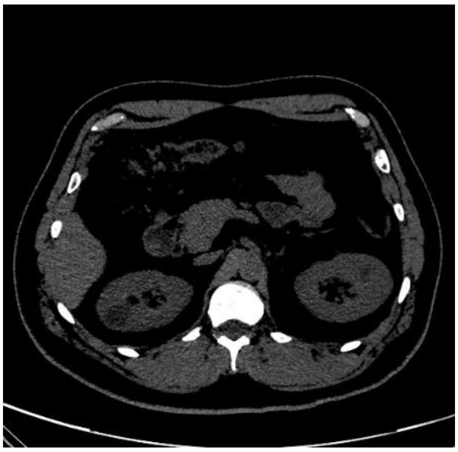
S.NO	CLASS	NO OF IMAGES
1	CYST	3709
2	NORMAL	5077
3	STONE	1377
4	TUMOR	2283

Upon analysis, a significant class imbalance was evident, with the Stone class having notably fewer images compared to others, potentially affecting the model's ability to accurately classify this class. To address this imbalance, data augmentation techniques were employed specifically for the Stone class. These techniques included random horizontal and vertical flips, rotations up to 15 degrees, and adjustments to color jitter (brightness, contrast, saturation, hue).

The augmentation process effectively increased the number of images in the Stone class from 1,377 to 2,200, ensuring a more balanced representation of this class in the dataset. Additionally, to create a balanced dataset for training and evaluation, the number of images in the Normal, Cyst, and Tumor classes were reduced to 2,200 each, aligning with the augmented size of the Stone class. Consequently, the final dataset comprised 8,800 images in total, with each class (Normal, Cyst, Stone, Tumor) containing **2,200 images**, providing a balanced foundation for training and evaluating the classification model.

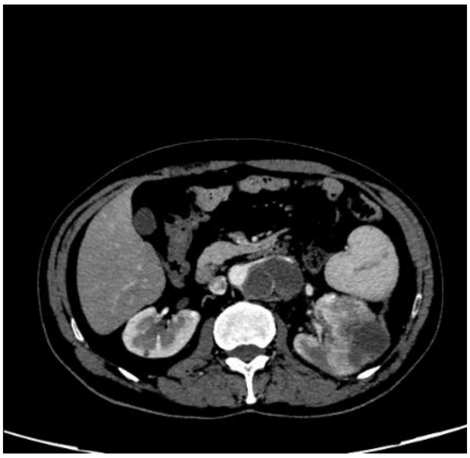
Sample Data

Normal



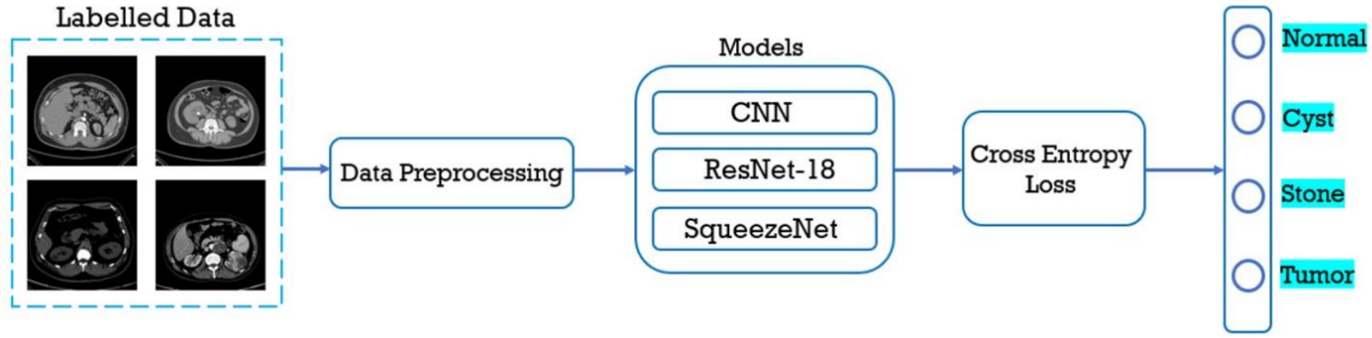
Cyst

Stone



Tumor

METHODOLOGY



1. Data Loading:
    - The first step involves downloading the dataset from Kaggle. This dataset contains CT scan images categorized into four classes: cyst, normal, stone, and tumor.

2. Data Balancing:
    - After loading the dataset, it is crucial to identify and address any class imbalances to ensure effective model training. The original distribution of images is as follows: cyst (3,709 images), normal (5,077 images), stone (1,377 images), and tumor (2,283 images).
    - To address the underrepresentation of the stone class, data augmentation techniques are applied specifically to this class. These techniques include random horizontal and vertical flips, rotations, and color jitter adjustments, which help to artificially increase the number of images in the stone class.
    - Following augmentation, the dataset is adjusted to contain 2,200 images per class. This results in a balanced dataset with a total of 8,800 images, ensuring that each class is equally represented.

3. Dataset Splitting:
    - The balanced dataset is then split into three subsets: training, validation, and test sets. This is done to evaluate the model's performance effectively.
    - The training set comprises 70% of the dataset, which is used to train the model.
    - The validation set comprises 15% of the dataset, which is used to tune the model's hyperparameters and to prevent overfitting.
    - The test set comprises the remaining 15% of the dataset, which is used to evaluate the final model performance.
    - It is essential to ensure that each subset maintains a balanced representation of the four classes to provide an accurate assessment of the model's performance across all categories.

4. Model Selection:
   CNN Model:
   - Utilize a basic Convolutional Neural Network (CNN) architecture suitable for image classification tasks, featuring alternating convolutional and pooling layers followed by fully connected layers.

   ResNet-18 Model:
   - Implement ResNet-18, a deeper neural network with residual connections to address vanishing gradient issues and improve training efficiency, known for high performance on complex image classification tasks.

   SqueezeNet Model:
   - Use SqueezeNet, an efficient neural network architecture designed to achieve high accuracy with fewer parameters, employing fire modules to minimize parameter count while maintaining performance.

5. Training the Models
   - Loss Function and Optimizer:
   - Use Cross Entropy Loss as the criterion for multi-class classification tasks, as it is effective in handling probability distributions across multiple classes.
   - Apply the Adam optimizer with a learning rate of 0.001, which combines the advantages of AdaGrad and RMSProp, making it efficient for training deep networks. Additionally, implement L2 regularization (weight decay) to prevent overfitting by penalizing large weight magnitudes.

6. Training Process:
   - Train each model using the training set, ensuring data is shuffled to improve generalization.
   - Validate the models using the validation set to monitor performance and adjust hyperparameters as needed, preventing overfitting and ensuring robustness.
   - Conduct multiple epochs of training, monitoring both training and validation loss to detect and address overfitting or underfitting, and continue until the model performance stabilizes.

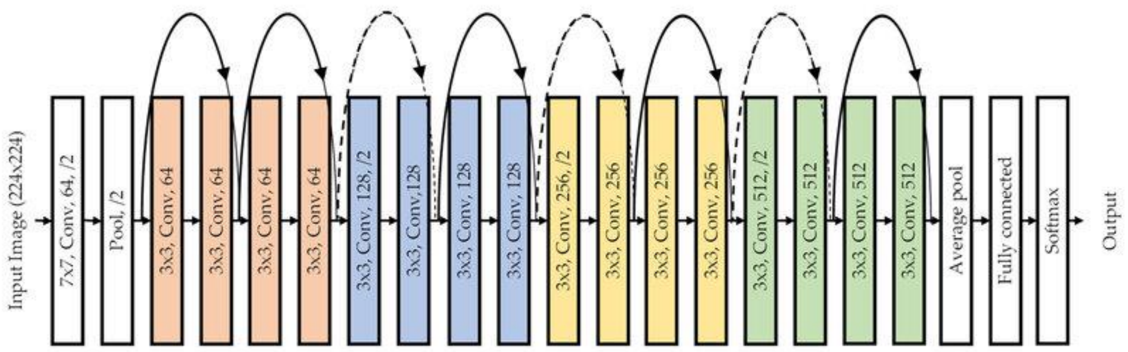
7. Model Evaluation
   - After training, evaluate the models on the test set to assess their performance on unseen data.
   - Use metrics including accuracy, precision, recall, and F1-score to provide a comprehensive evaluation of the classification results, highlighting the model's strengths and weaknesses in distinguishing between different classes.

8. Post-Processing and Analysis
   - Analyze the performance of each model based on evaluation metrics, reviewing confusion matrices and other diagnostic tools to understand the models' classification behavior.
   - Compare the results of all models to determine the best-performing one, considering overall accuracy, precision, recall, and F1-score.

Model Architectures

1. ResNet18

ResNet-18 is a deep convolutional neural network designed for image classification, comprising 18 layers. The architecture features convolutional layers, batch normalization, ReLU activation functions, and fully connected layers. A key innovation in ResNet-18 is the use of skip connections, or residual connections, which link the input of a layer directly to the output of a subsequent layer. This effectively addresses the vanishing gradient problem and facilitates the training of deeper networks. The network is organized into residual blocks, each containing two convolutional layers followed by batch normalization and ReLU activations. It begins with a convolutional layer that has 64 filters of size 7x7, followed by a max-pooling layer. Subsequent layers are divided into four stages, each with two residual blocks. Most residual blocks use identity shortcuts, but some employ 1x1 convolutional layers to match dimensions when needed.

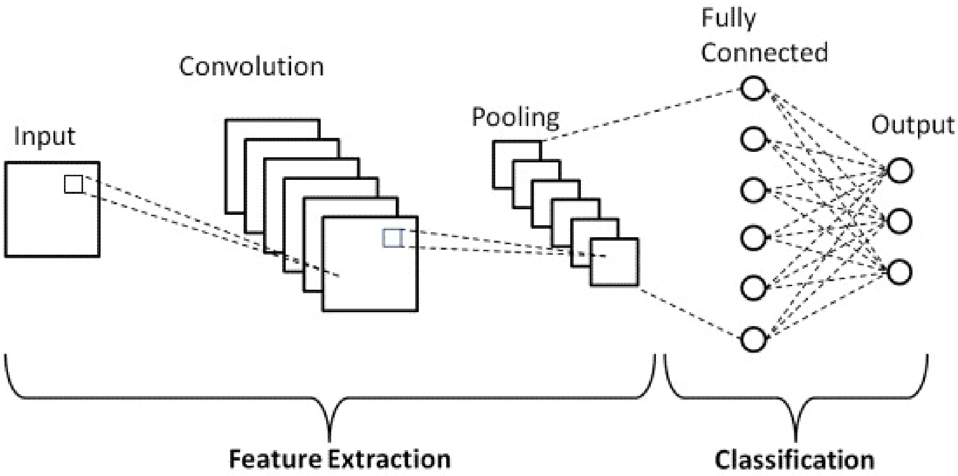


Skip connections in ResNet-18 enhance training efficiency by ensuring smooth gradient flow, allowing the model to learn complex and deep features. This capability helps the network capture intricate patterns in images, essential for accurate classification. Despite its relatively smaller size compared to other deep networks, ResNet-18 achieves strong performance due to its effective gradient handling and deep feature learning, making it a robust and efficient choice for image classification tasks.

2. Convolution Neural Networks (CNN):

Convolutional Neural Networks (CNNs) are pivotal in processing grid-like data such as images, facilitating the learning of spatial hierarchies through convolution, pooling, and fully connected layers. These layers sequentially transform input data to desired output, constituting the fundamental architecture of CNNs.

CNNs typically commence with convolutional layers that learn distinctive features, such as edges and textures, from input images. These features are subsequently refined through pooling layers, which reduce spatial dimensions while maintaining crucial information, thereby enhancing computational efficiency and providing spatial invariance. Following this, fully connected layers integrate the learned features to facilitate high-level reasoning and final classification. The output layer, tailored to the specific task, furnishes predictions often in the form of class probabilities.



In our CNN model architecture, an input image of size (224, 224, 3) undergoes a series of convolutional and pooling layers. Specifically, the initial convolutional layer (Conv1) transforms the input, leading to subsequent pooling (Pool1), convolutional (Conv2), and pooling (Pool2) layers. Further convolutional (Conv3) and pooling (Pool3) layers refine the learned features. A flattening layer then converts the output to a 1D vector, facilitating processing by fully connected layers (FC1, FC2). Additionally, dropout regularization is employed to mitigate overfitting, ensuring the robustness of the model. This comprehensive architecture effectively processes and classifies kidney disease images, leveraging convolution, pooling, and fully connected layers in conjunction with regularization techniques to enhance performance and generalization.

3. SqueezeNet:

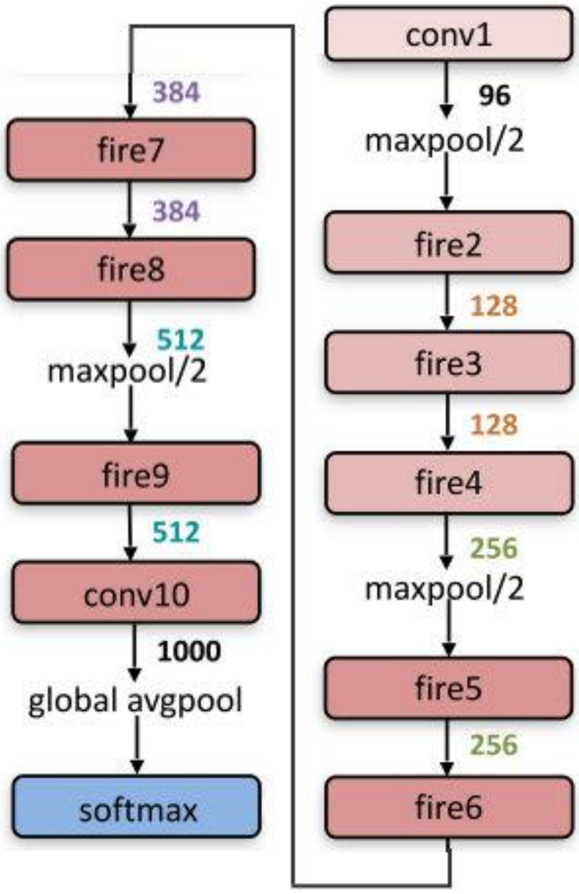
The main objective of this development was to identify CNN architectures that have few parameters while maintaining competitive accuracy. So, they came up with the below points for efficient implementation:

• Strategy 1: Replace 3x3 filters with 1x1 filters

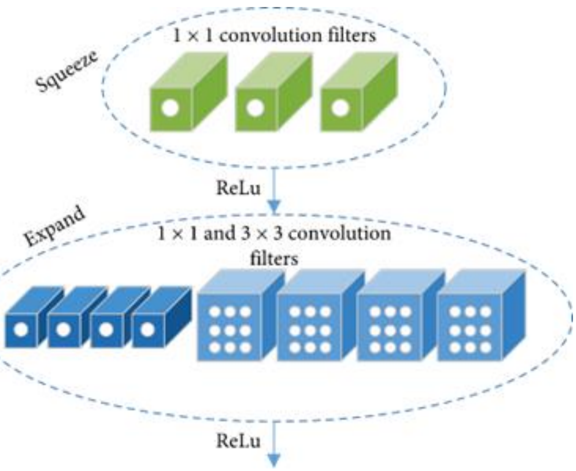
• Strategy 2: Decrease the number of input channels to 3x3 filters: We decrease the number of input channels to 3x3 filters using squeeze layers

• Strategy 3: Down sample late in the network so that convolution layers have large activation maps. Here the Intuition is that large activation maps (due to delayed down sampling) can lead to higher classification accuracy, with all else held equal.

Strategies 1 and 2 are about judiciously decreasing the quantity of parameters in a CNN while attempting to preserve accuracy. Strategy 3 is about maximizing accuracy on a limited budget of parameters.



Fire Module:



A squeeze convolution layer (which has only 1x1 filters), feeding into an expand layer that has a mix of 1x1 and 3x3 convolution filters.

There are three tunable dimensions (hyperparameters) in a Fire module s_{1x1}, e_{1x1}, e_{3x3}

- s_{1x1}: is the number of filters in the squeeze layer (all 1x1)
- e_{1x1}: is the number of 1x1 filters in the expand layer
- e_{3x3}: is the number of 3x3 filters in the expand layer
- $s_{1x1}<(e_{1x1}+e_{3x3})$ so, squeeze layer helps to limit the number of input channels to the 3x3 filters

As the name suggests the squeeze layer is to limit the number of inputs to the 3x3 filters, as the role of 1x1 filters is dimension reduction (limit number of channels)

Evaluation Metrics

1. **Accuracy:**

- Accuracy is the ratio of correctly predicted instances (both true positives and true negatives) to the total number of instances. The formula is:

  Accuracy = (TP + TN) / (TN + FP + TP + FN)

  Where:

  TP (True Positive) = Number of instances correctly predicted as positive

  TN (True Negative) = Number of instances correctly predicted as negative

  FP (False Positive) = Number of instances incorrectly predicted as positive

  FN (False Negative) = Number of instances incorrectly predicted as negative

- Accuracy provides an overall measure of the model's performance but can be misleading for imbalanced datasets.

2. **F1 score:**

- The F1 score is the harmonic mean of precision and recall, providing a balanced measure of a model's performance. The formula is:

  F1 score = (2 × Precision × Recall) / (Precision + Recall)

3. **Precision:**

- Precision is the ratio of true positives to the total number of instances predicted as positive. The formula is:

  Precision = TP / (TP + FP)

- Precision measures the model's ability to avoid false positive predictions.

4. **Recall:**

- Recall is the ratio of true positives to the total number of actual positive instances. The formula is:

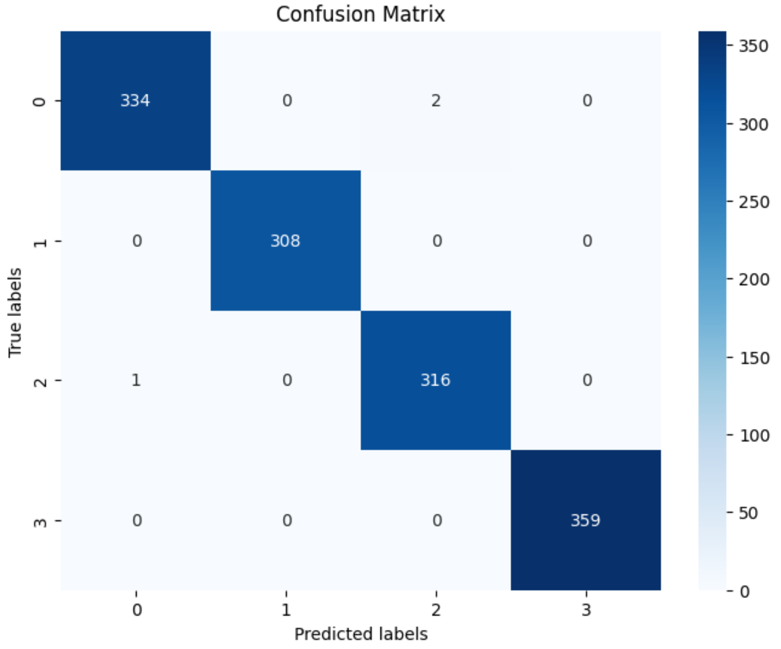
  Recall = TP / (TP + FN)

- Recall measures the model's ability to correctly identify all positive instances.

Experimental Results:

1. CNN

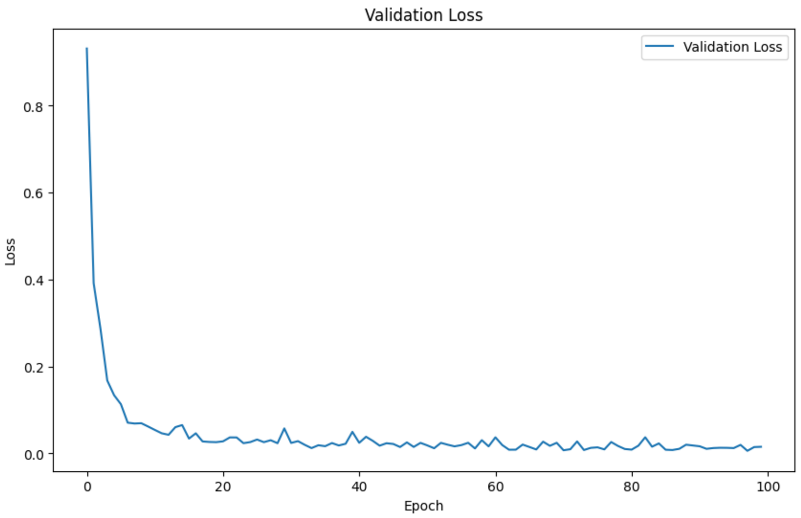
Confusion Matrix:

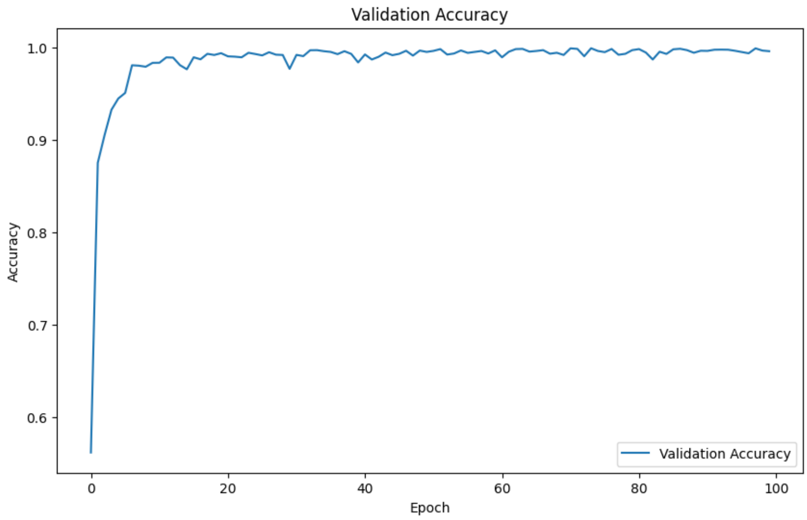


Results:

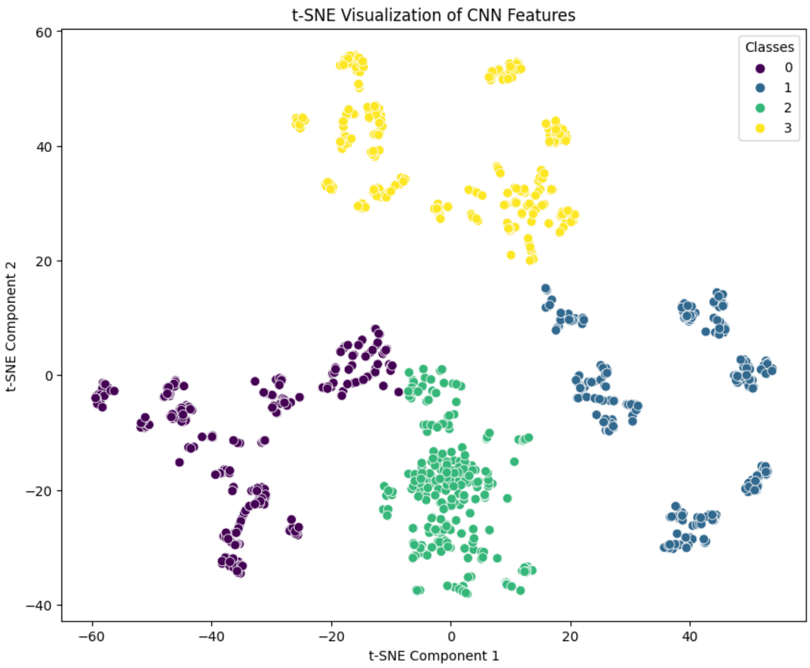
Accuracy	0.9977
Precision	0.9976
Recall	0.9977
F1-Score	0.9977

Validation Accuracy and Loss Curves:



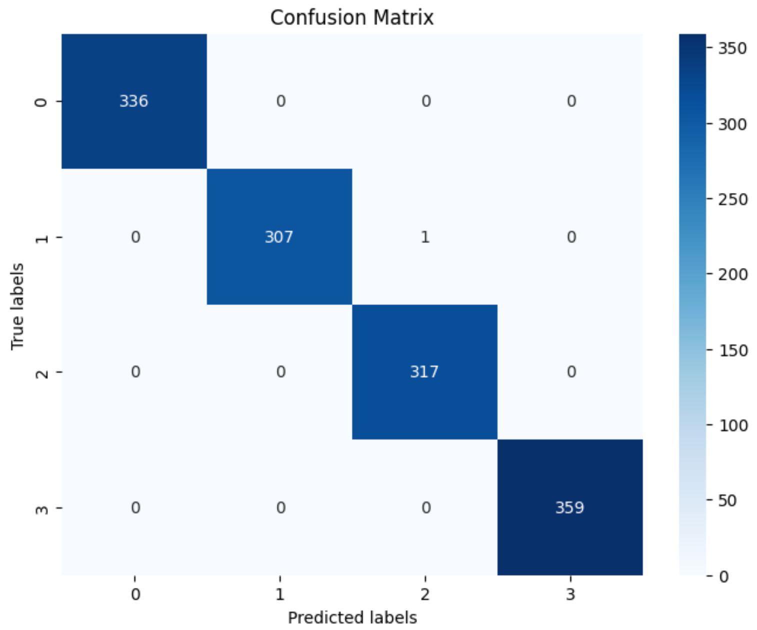

Validation Accuracy
1.0
0.9
0.8
0.7
0.6
Accuracy
Validation Accuracy
0
20
40
60
80
100
Epoch

t-SNE Plot:


t-SNE Visualization of CNN Features
60
40
20
0
−20
−40
t-SNE Component 2
Classes
0
1
2
3
−60
−40
−20
0
20
40
t-SNE Component 1

2. ResNet18

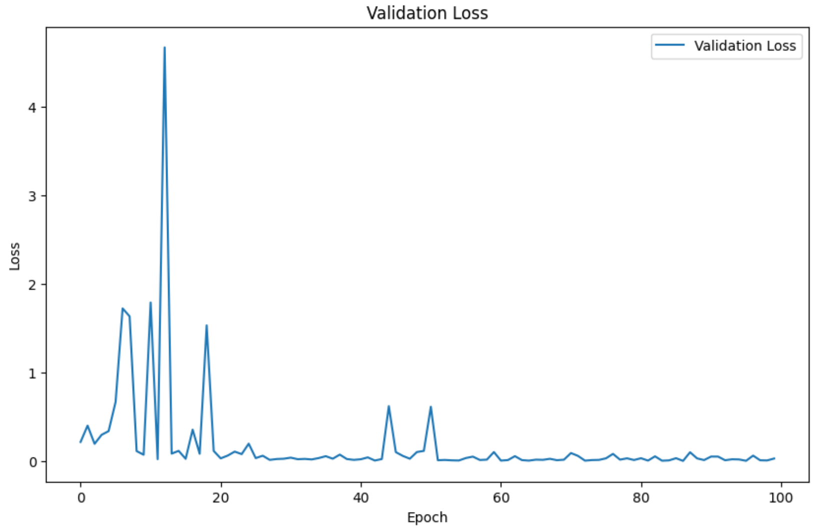
Confusion Matrix:

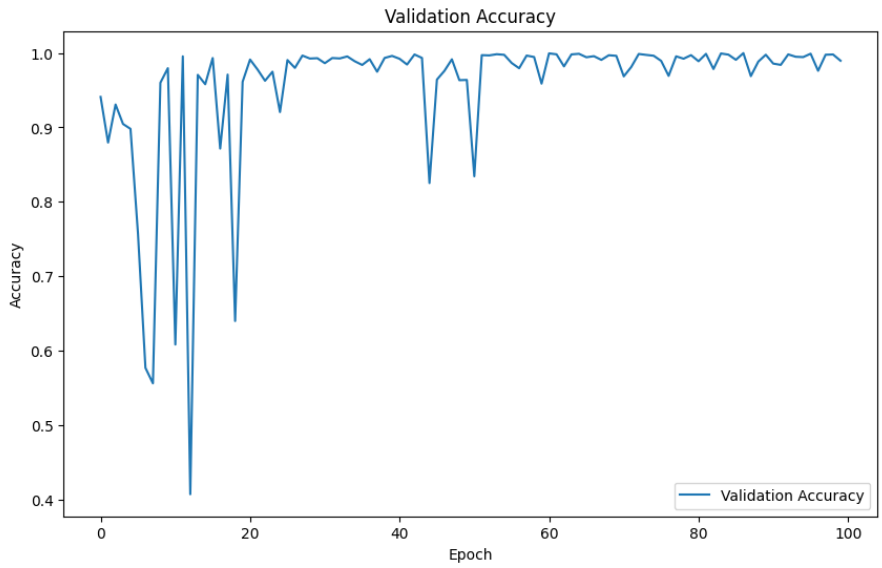


Results:

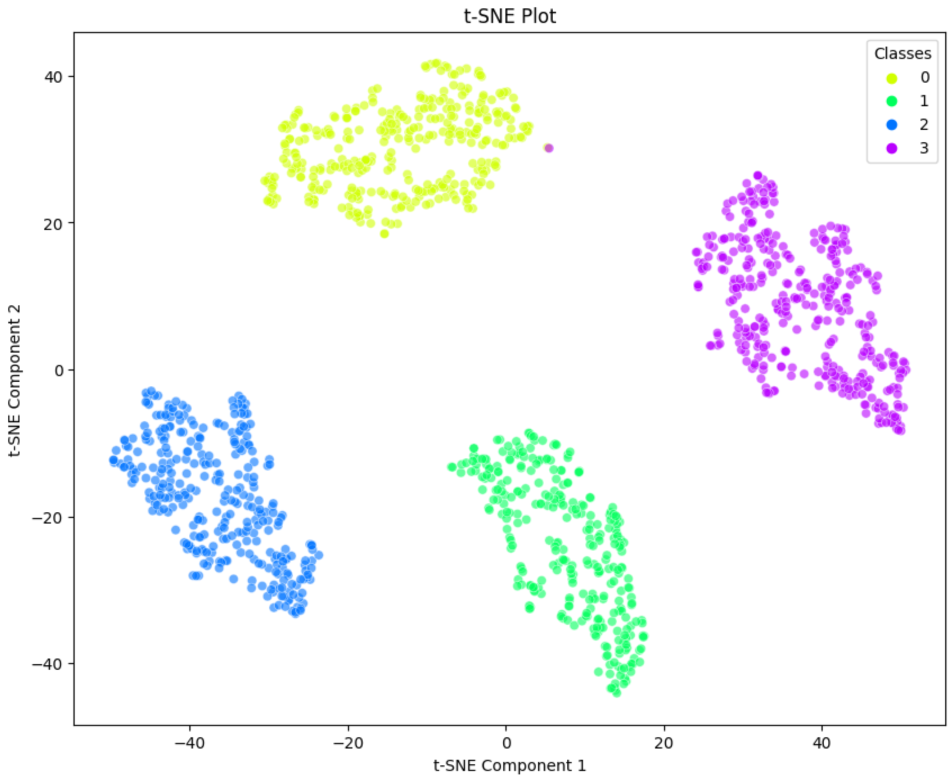
Accuracy	0.9992
Precision	0.9992
Recall	0.9991
F1-Score	0.9991

Validation Accuracy and Loss Curves:



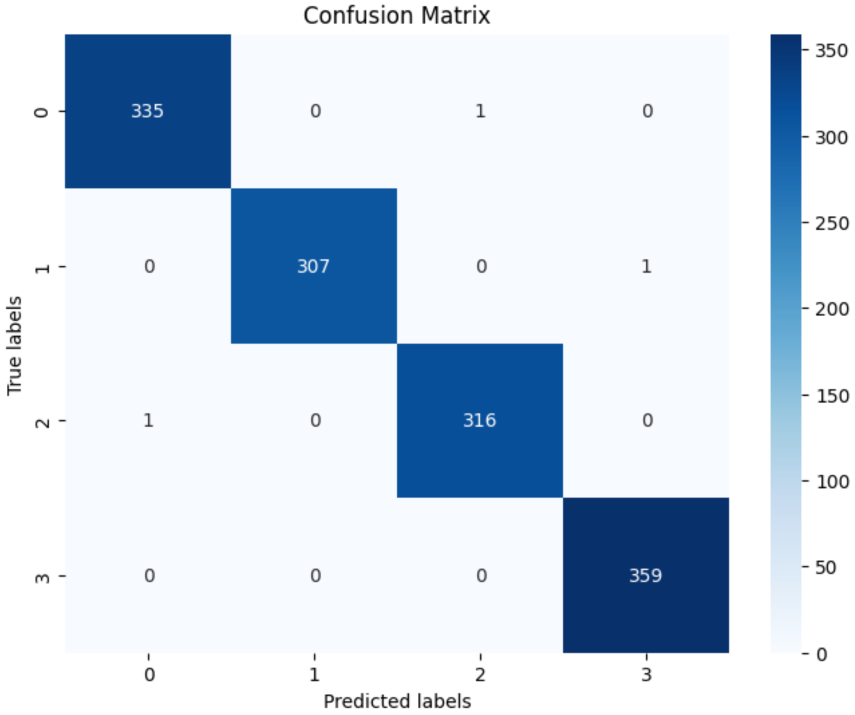

Validation Accuracy
Accuracy
1.0
0.9
0.8
0.7
0.6
0.5
0.4
0
20
40
60
80
100
Epoch
Validation Accuracy

t-SNE Plot:


t-SNE Plot
Classes
0
1
2
3
t-SNE Component 2
40
20
0
−20
−40
−40
−20
0
20
40
t-SNE Component 1

3. SqueezeNet:

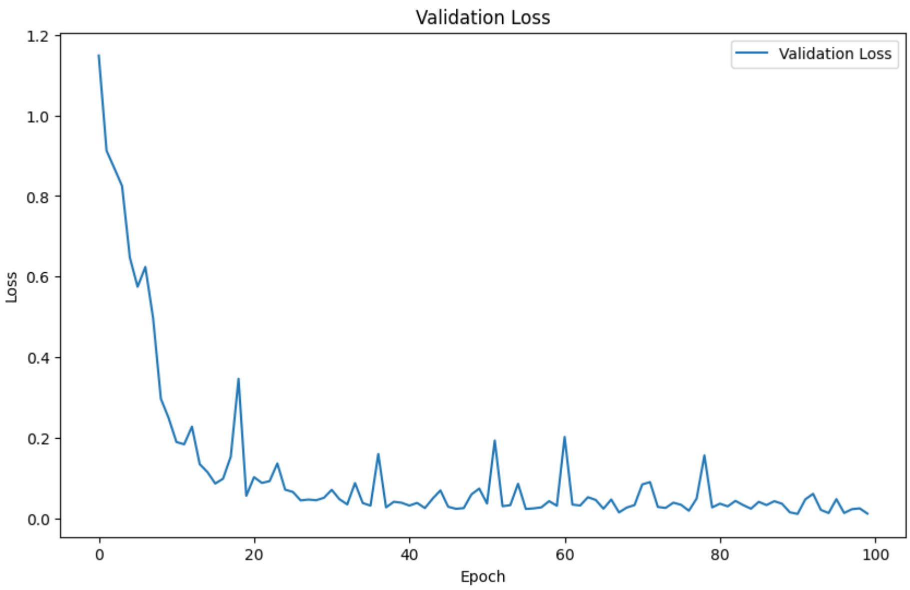
Confusion Matrix:

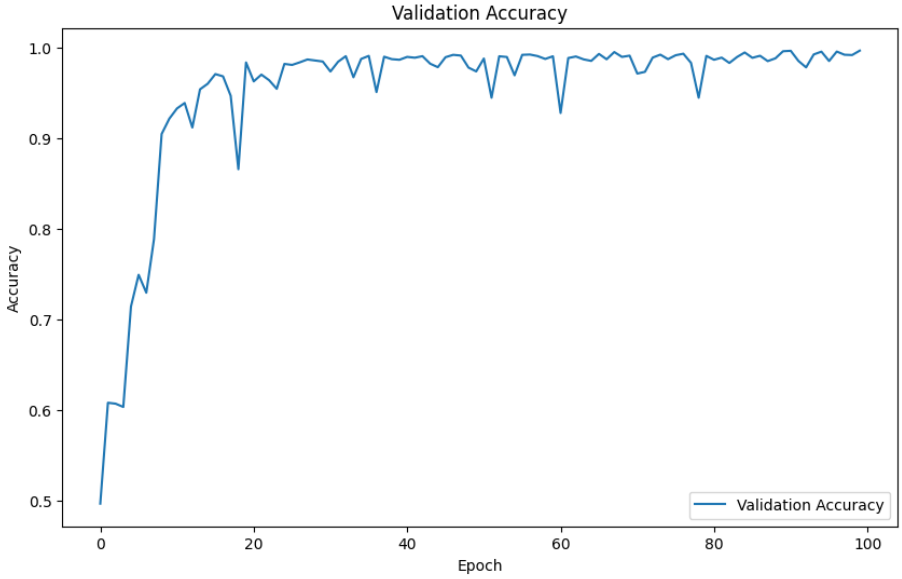


Results:

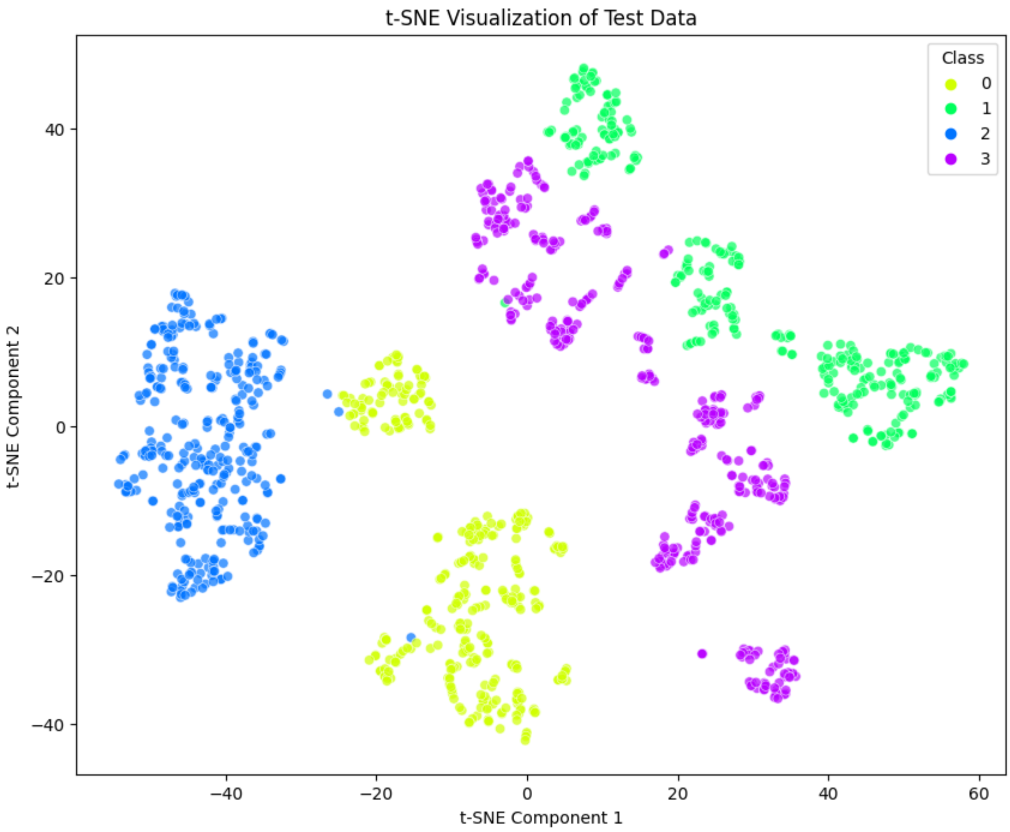
Accuracy	0.9977
Precision	0.9977
Recall	0.9976
F1-Score	0.9975

Validation Accuracy and Loss Curves:




Validation Accuracy
1.0
0.9
0.8
0.7
0.6
0.5
Accuracy
Validation Accuracy
0
20
40
60
80
100
Epoch

t-SNE Plot:


t-SNE Visualization of Test Data
Class
0
1
2
3
40
20
0
−20
−40
t-SNE Component 2
−40
−20
0
20
40
60
t-SNE Component 1

INFERENCE

Through the comprehensive evaluation of deep learning models for kidney disease classification, this project underscores the efficacy of advanced architectures in accurately diagnosing medical conditions from imaging data. The superior performance of ResNet-18, characterized by its exceptional accuracy and robustness, positions it as the primary choice for deployment in clinical settings where precise diagnosis is paramount. Additionally, the exploration of alternative models such as CNN and SqueezeNet highlights the importance of considering trade-offs between performance and resource efficiency, offering valuable insights for applications in resource-constrained environments.

In practical terms, the findings of this study offer significant implications for the development of automated diagnostic systems in healthcare. By leveraging the capabilities of deep learning architectures, healthcare professionals can enhance the efficiency and accuracy of disease diagnosis, leading to improved patient outcomes and streamlined clinical workflows. Moreover, the adaptability of these models to diverse computational environments ensures their applicability across a wide range of healthcare settings, from well-equipped hospitals to remote medical facilities, thus paving the way for accessible and equitable healthcare solutions.

CONCLUSION

The results of this comprehensive study underscore the remarkable efficacy of deep learning models in accurately classifying kidney diseases from medical images. Through meticulous evaluation and analysis, it is evident that these models exhibit significant promise in revolutionizing the diagnostic process for such critical medical conditions. Upon careful examination of the evaluation outcomes, ResNet-18 emerges as the foremost recommendation for deployment in real-world applications. Its unparalleled accuracy and overall performance make it an ideal choice for scenarios where precision and robustness are paramount. ResNet-18's advanced architectural design, notably incorporating residual connections, equips it with the capability to discern subtle nuances in medical images, thus enhancing diagnostic accuracy and reliability. Nevertheless, it is imperative to recognize the value of alternative models such as CNN and SqueezeNet. These models present compelling alternatives, particularly in environments constrained by limited computational resources. CNN, with its established effectiveness in image processing tasks, remains a strong contender, offering a balance between performance and resource efficiency. Similarly, SqueezeNet emerges as a noteworthy candidate, especially in contexts where mobility or embedded applications are prevalent. Its compact model size renders it well-suited for deployment in scenarios where space and computational constraints are prominent considerations. In summary, while ResNet-18 stands out as the primary recommendation for its exceptional performance, CNN and SqueezeNet present viable alternatives, each catering to specific requirements and constraints. This nuanced understanding of the capabilities and suitability of different deep learning architectures is pivotal in informing decision-making processes for the deployment of kidney disease classification systems in diverse clinical settings.