Automated Kidney Condition Classification Using MobileNetV2: A Deep Learning Approach for Multiclass Diagnosis



By

Muhammad Asif

Aqib Hussain

Mushahid hussain

Department of Information Technology

University of Haripur

May, 2025

Automated Kidney Condition Classification Using MobileNetV2: A Deep Learning Approach for Multiclass Diagnosis



BY

Muhammad Asif, Aqib Hussain, Mushahid Hussain

F21-1769, F21-0470, F21-0525

Supervised by

Dr. Shabih ul Hassan

Department of information technology

University od Haripur

May, 2025

Automated Kidney Condition Classification Using MobileNetV2: A Deep Learning Approach for Multiclass Diagnosis



BY

Muhammad Asif, Aqib Hussain, Mushahid Hussain

A Dissertation Submitted in Partial Fulfilment for the Degree of

BACHELORS

IN

COMPUTER SCIENCE

Department of information technology

University od Haripur

May, 2025

Automated Kidney Condition Classification Using MobileNetV2: A Deep Learning Approach for Multiclass Diagnosis

By

Muhammad Asif, Aqib Hussain, Mushahid Hussain

CERTIFICATE

A THESIS SUBMITTED IN THE PARTIAL FULFILMENT OF THE

REQUIRMENTS FOR THE DEGREE OF BACHELORS

IN Computer Science

We accept this dissertation as conforming to the required standards.

Department of Information Technology

University of Haripur

May, 2025

Declaration

I hereby declare that this dissertation has neither been copied in whole nor in part from any source. If any part of this thesis is proven to be copied or found to be a report of someone else's work, I shall bear the consequences. No portion of this work presented in the thesis has been submitted in support of any application for any other degree or qualification at any other university or institute of learning.

I further declare that this research and all associated documents, reports, and records are submitted as a partial requirement for the degree of Bachelors in Artificial Intelligence. I understand and transfer copyrights for these materials to the University of Haripur and will not sell this research and documents for financial gain.

It is further declared that I developed this thesis entirely on the basis of my personal efforts, made under the sincere guidance of my project supervisor, Dr. Adeel Ahmed, Department of Information Technology, University of Haripur, Haripur.

Date: 22nd, May, 2025 Aqib Hussain Mushahid Hussain Muhammad Asif

Plagiarism Certificate

This thesis has been checked for Plagiarism. The Turnitin report endorsed by the Supervisor is attached at the end of thesis.

Date: 14th Jan, 2025 Aqib hussain , Mushahid hussain , Muhammad asif

Student Name(s)

Student Signature

Supervisor Signatur

Dedicated to:

**Our beloved Parents**

Acknowledgement

I want to start by sincerely thanking Almighty Allah (SWT), the Most Merciful and the Most Gracious, for his unending blessings, strength, and guidance, which allowed me to finish this thesis. All of this would not have been possible without His heavenly assistance. I humbly offer my sincere respect to the Fourteen infallible (A.S.) whose teachings and lives are a source of unending inspiration, wisdom, and light. Their commitment to justice, truth, and knowledge keeps illuminating the way for seekers like me.

I genuinely appreciate my supervisor's sir Shabih ul Hassan unwavering support, insightful advice, and professional direction during this research. Their support and helpful criticism enabled me to advance both academically and personally.

I also want to thank my family, especially my parents, who have been the foundation of my journey with their sacrifices, prayers, and unwavering faith in my abilities. Their moral and emotional support has always helped me get through difficult times.

Abstract

Researchers use MobileNetV2 to create an automated kidney disease classification system that can multiclass diagnose CT scan images with 98% accuracy. Across a dataset of 1,110 scans, the deep learning model performs exceptionally well in differentiating between cysts (F1=0.97), normal kidneys (F1=0.99), stones (F1=0.97), and tumors (F1=0.98). Consistent macro-averaged metrics (precision=0.98, recall=0.98) demonstrate that the system maintains MobileNetV2's computational efficiency while providing clinical-grade accuracy by utilizing transfer learning with optimized hyperparameters. Specialized preprocessing for CT artefacts and a balanced augmentation approach that tackles class distribution issues are two important innovations. For early intervention, the model's high sensitivity for cysts (recall=0.98) and unique strength in tumor detection (precision=0.99) are essential. Inference times are appropriate for clinical workflows, and comparative analysis shows a 15% increase in accuracy over conventional diagnostic algorithms.

Table of Contents

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Declaration | | | | . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . | | | | | | | | i |
| Dedication | | | | . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . | | | | | | | | ii |
| Acknowledgements | | | | | | | | | | . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . | iii | |
| Abstract | | | . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . | | | | | | | | | iv |
| [List of Figures](#page8) | | | | | | | . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . | | | | | v |
| [List of Tables](#page8) | | | | | | | . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . | | | | | vi |
| [**Chapter 1**](#page12) | | | | | | | [**Introduction**](#page12). . . . . . . . . . . . . . . . . . . . . . . . . . . . . . | | | | | 1 |
| 1.1 | Overview | | | | | | | | | | | 1 |
| 1.2 | Problem Statement | | | | | | | | | | | 2 |
| 1.3 | Introduction to Kidney stone detection | | | | | | | | | | | 2 |
| 1.4 | Traditional method of kidney stone detection | | | | | | | | | | | 2 |
| 1.5 | Complication in Traditional Method | | | | | | | | | | | 3 |
| 1.6 | Why early detection is important? | | | | | | | | | | | 4 |
| 1.7 | Role of Machine learning in detection | | | | | | | | | | | 4 |
| 1.8 | Model architecture of mobileNetV2 | | | | | | | | | | | 4 |
| 1.9 | Comparison b/w Traditional method & AI-based Model | | | | | | | | | | | 6 |
| 1.10 | Imaging preprocessing Techniques | | | | | | | | | | | 7 |
| 1.11 | Research Questions | | | | | | | | | | | 7 |
| 1.12 | Research Objectives | | | | | | | | | | | 8 |
| 1.13 | Thesis Structure | | | | | | | | | | | 8 |
| [**Chapter 2**](#page13) | | | | | | | **Related Work** | | | | | 10 |
| 2.1 | | | | | | | Introduction | | | | | 10 |
| 2.2 | | | | | | | Medical Image Analysis in Healthcare | | | | | 11 |
| 2.3 | | | | | | | Role of computer vision in medica diagnostic | | | | | 11 |
| 2.4 | | | | | | | Success stories of image-based disease diagnostic | | | | | 12 |
| 2.5 | | | | | | | CT Scan Imaging in kidney analysis | | | | | 12 |
| 2.6 | | | | | | | Traditional method for kidney disease diagnostic | | | | | 13 |
|  | | | | | | | 2.6.1 | | | Nephrology’s rule-based system | | 13 |
|  | | | | | | | 2.6.2 | | | Classical Image Preprocessing techniques | | 13 |
|  | | | | | | | 2.6.3 | | | Reliance on radiologist expertise and manual diagnostic | | 13 |
| 2.7 | | | | | | | Machine Learning techniques for kidney classification | | | | | 14 |
|  | | | | | | | 2.7.1 | | | Early machine learning models SVM, Random forest and K-NN | | 14 |
|  | | | | | | | 2.7.2 | | | Features extraction techniques | | 14 |
|  | | | | | | | 2.73 | | | Performance matrix and limitation | | 15 |
| 2.8 | | | | | | | Deep learning in medical imaging | | | | | 15 |
|  | | | | | | | 2.8.1 | | | Transition from machine learning to deep leatning | | 16 |
|  | | | | | | | 2.8.2 | | | CNN Architecture basics | | 16 |
|  | | | | | | | 2.8.3 | | | Advantages of deep learning | | 16 |
| 2.9 | | | | | | | Existing model of deep learning | | | | | 17 |
| 2.10 | | | | | | | Overview of key research works using CNN, VGG, etc | | | | | 18 |
| 2.11 | | | | | | | Specific studies on kidney stone, cyst and tumor detection | | | | | 19 |
| 2.12 | | | | | | | Dataset Sources | | | | | 19 |
| 2.13 | | | | | | | Gap Analysis | | | | | 20 |
| [**Chapter 3**](#page14) | | | | | | | [**Proposed System**](#page14) . . . . . . . . . . . . . . . . . . . . . . . . . . . . | | | | | 22 |
| 3.1 | | Requirement Analysis. | | | | | | | | | | 22 |
|  | | 3.1.1 | | | | Functional Requirements | | | | | | 22 |
|  | |  | | | | 3.1.1.1 | | Image Classification | | | | 23 |
|  | |  | | | | 3.1.1.2 | | Image Preprocessing | | | | 23 |
|  | |  | | | | 3.1.1.2 | | Model Training and validation | | | | 23 |
|  | |  | | | | 3.1.1.4 | | | | Model evaluation and reporting | | 24 |
|  | |  | | | | 3.1.1.5 | | | | Prediction interface | | 24 |
|  | | 3.1.2 | | | | Non-functional requirements | | | | | | 25 |
|  | |  | | | | 3.1.2.1 | | | Accuracy and reliability | | | 25 |
|  | |  | | | | 3.1.2.2 | | | Performance and efficiency | | | 25 |
|  | |  | | | | 3.1.2.3 | | | Robustness | | | 25 |
|  | |  | | | | 3.1.1.4 | | | Security and data privacy | | | 26 |
|  | | 3.1.3 | | | | Hardware requirements | | | | | | 26 |
|  | | 3.1.4 | | | | Software Requirements | | | | | | 27 |
| 3.2 | | System Design | | | | | | | | | | 29 |
|  | | 3.2.1 | | | System Architecture | | | | | | | 29 |
|  | | 3.2.2 | | | Data Flow Design | | | | | | | 31 |
|  | | 3.2.3 | | | Dataset Structure | | | | | | | 32 |
|  | | 3.2.4 | | | Model Integration and deployments | | | | | | | 33 |
|  | | 3.2.5 | | | Design Consideration | | | | | | | 33 |
| 3.3 | | System Methodology | | | | | | | | | | 33 |
|  | | 3.3.1 | | | | | Data Preparation | | | | | 34 |
|  | | 3.3.2 | | | | | Preprocessing | | | | | 35 |
|  | | 3.3.3 | | | | | Model development | | | | | 35 |
|  | | 3.3.4 | | | | | Model Training strategy | | | | | 36 |
|  | | 3.3.5 | | | | | Model Saving and Evaluation | | | | | 37 |
| [**Chapter 4**](#page15) | | | | | | | [**Evaluation and Results**](#page15) . . . . . . . . . . . . . . . . . . . . . . . . | | | | | 39 |
| 4.1 | | Implementation . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . | | | | | | | | | | 39 |
|  | | 4.1.1 | | | | | Environmental setup and imports | | | | | 39 |
|  | | 4.1.2 | | | | | Data collection | | | | | 40 |
|  | | 4.1.3 | | | | | Dataset loading and splitting | | | | | 41 |
|  | | 4.1.4 | | | | | Data normalization | | | | | 41 |
|  | | 4.1.5 | | | | | Sample of visualization | | | | | 42 |
|  | | 4.1.6 | | | | | Data augmentation pipeline | | | | | 43 |
|  | | 4.1.7 | | | | | Model training | | | | | 43 |
|  | | 4.1.8 | | | | | Training results | | | | | 43 |
|  | | 4.1.9 | | | | | Model summary | | | | | 44 |
|  | | 4.1.10 | | | | | Fine-tuning the model | | | | | 45 |
|  | | 4.1.11 | | | | | Fine-tuning results | | | | | 46 |
|  | | 4.1.12 | | | | | Performance matrices Analysis | | | | | 47 |
|  | | 4.1.13 | | | | | Confusion matrix analysis | | | | | 48 |
| 4.2 | | Testing . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . | | | | | | | | | | 49 |
|  | | 4.2.1 | | | | | Overview of testing | | | | | 49 |
|  | | 4.2.2 | | | | | Actual testing situation | | | | | 49 |
|  | | 4.2.3 | | | | | Gradio , A framework for rapid deployment | | | | | 49 |
|  | |  | | | | | 4.2.3.1 | | | Introduction to Gradio | | 49 |
|  | |  | | | | | 4.2.3.2 | | | Quick deployment and prototyping | | 50 |
|  | |  | | | | | 4.2.3.3 | | | Support for inputs from medical imaging | | 50 |
|  | |  | | | | | 4.2.3.4 | | | Integration with AI and ML workflow | | 50 |
|  | |  | | | | | 4.2.3.5 | | | Why Gradio was chosen? | | 51 |
|  | | 4.2.4 | | | | | Kidney classifier | | | | | 51 |
|  | | 4.2.5 | | | | | System test resilt | | | | | 52 |
| [**Chapter 5**](#page17) | | | | | | | **Conclusion and Future Work**. . . . . . . . . . . . . . . . . . . . . | | | | | 54 |
| 5.1 Conclusion . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . | | | | | | | | | | | | 54 |
| 5.2 Future Work . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . | | | | | | | | | | | | 55 |
| [**References**](#page18) | | | | | | | . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . | | | | | 56 |

List of Figures

|  |  |  |
| --- | --- | --- |
| **Figure** | **Description** | **Page** |
| 1 | Figure 1.1 model architecture of mobileNetv2 | 5 |
| 2 | Figure 3.1 working flow of mobileNetV2 | 28 |
| 3 | Figure 3.2 dataset structure | 30 |
| 4 | Figure 4.1 sample of dataset | 38 |
| 5 | Figure 4.2 confusion matrix | 44 |
| 6 | Figure 5.1 kidney classifier user interface | 43 |
| 7 | Figure 5.2 System test result | 44 |

List of Tables

|  |  |  |
| --- | --- | --- |
| **Table** | **Description** | **Page** |
| 1 | Table 1.1 comparison b/w traditional and AI-based method | 6 |
| 2 | Table 1.2 common image preprocessing techniques | 7 |
| 3 | Table 2.1 overview of key research work | 17 |
| 4 | Table 2.2 studies | 18 |
| 5 | Table 3.1 list of hardware requirements | 25 |
| 6 | Table 3.2 list of software requirements | 25 |
| 7 | Table 4.1 training result | 40 |
| 8 | Table 4.2 model summary | 41 |
| 9 | Table 4.3 fine-tuning result | 42 |
| 10 | Table 4.4 performance matrix | 43 |

Chapter 1

Introduction

# **1.1 Overview**

The disease of kidney stones is common among urological diseases and involves hard calcifications in the kidneys. To avoid complications, such as urinary tract obstruction and damage to the kidneys, early detection is critical. Current diagnostic methods make use of imaging scanning such as CT scans, ultrasounds, and others which are heavily reliant on human interpretation which is often slow and prone to errors. The recent advances in artificial intelligence (AI) and machine learning (ML) offer new hope to improve the accuracy and efficiency of imaging diagnostics. This thesis investigates the use of MobileNetV2, a lightweight deep learning model, for the automated detection of stones in the kidney from medical images. The aim of the study was to build clean and accessible tools alongside more advanced tools for data processing, model training, evaluation, and performance analysis of the constructed models for use by radiologists in actual practice.

# **1.2 Problem Statement**

The traditional way of detecting kidney stones through imaging is very labor-intensive and prone to human error, especially with smaller or radiolucent stones. Automated solutions are, therefore, required due to the growing workload faced by radiologists and the need for quick diagnoses. This study proposes to solve these challenges through developing an AI-based model for accurate detection of kidney stones with a view to assisting clinicians to reach timely and correct diagnoses.

# **1.3 Introduction to kidney stone detection**

Kidney stones, also referred to as renal calculi, are defined as stone-like structures created from minerals found in urine. If left untreated, they have the potential to inflict extreme suffering, cause blood in urine, and lead to renal dysfunction. In relation to detecting fragments, imaging procedures such as CT scans without contrast are regarded to have the highest accuracy and precision because of the imaging’s high sensitivity in identifying and validating. The interpretation of these images demands considerable knowledge and a large amount of time. The automation of image interpretation, made possible by AI, particularly through deep learning frameworks, reduces the time needed for diagnostics while improving precision trusts. Research has shown that the application of deep learning systems brings reliability and efficiency in the detection of kidney stones from CT scans, thus expediting the processes involved in diagnosis and treatment planning[1].

# **1.4 Traditional way to detect kidney stone**

Historically, for the detection of kidney stones, the standard method has been to look for clinical signs, proceed with a physical examination, and then move on to imaging methods that are interpreted manually by the professionals in health. Typical but severe flank pain, blood in the urine, nausea, and vomiting are some of the common symptoms that lead physicians to recommend diagnostic imaging. Currently, the most sensitive and specific test for diagnosing urinary stones is by non-contrast CT. This is gold standard and followed by ultrasound. This latter has a large-scale application to the pregnant women and children due to its safety and accessibility, though it's much less accurate than CT in detecting small stones.

The manual identification of the presence, size, and location of stones in imaging results is carried out by radiologists in a clinical setting. This task comes with certain limitations such as variability in interpretation among observers, experience-dependent judgments, and errors triggered by fatigue. Moreover, an increase in the volume of imaging brings about a delay in diagnosis, increasing the radiologists' workload. Traditional useful but laborious and time-consuming methods are a strong cue for the need for better and more accurate solutions; hence this project shall address the problem with AI-based automation by means of deep learning.

# **1.5 Complications in Traditional Method**

A diagnosis of kidney stones has, in past, been always associated with various clinical features leading to physical examination and eventual imaging procedures all of which were reviewed manually by different health professionals. Typical but severe flank pain, blood in urine, nausea, and vomiting are thus some common symptoms that will lead a physician to a recommendation for some form of diagnostic imaging. Presently, the most sensitive and specific test for the diagnosis of urinary stones is the non-contrast CT scan, which is essentially the gold standard, thereafter followed by ultrasound. The ultrasound device finds a very widespread application in the scanning of either pregnant women or children, due to its safety and access. However, this device is actually much less accurate compared to the aforementioned CT at picking up the small stones.

This is nothing but a manual recognition of the stones' presence, their size, and the location on the imaging results performed by radiologists in clinical practice. In this respect, there are some limitations such as variability in interpretation across observers, judgment-driven errors based on experience, and fatigue-induced mistakes. What's more, an increase in the load of imaging will further lengthen the diagnosis and increase the work of the radiologist. This need for new and better reliable solutions seems to be strongly emphasized by the prevailing situation with the traditional existing methods being very useful, but at the same time strenuous and time-consuming. Hence, this project shall address the problem with AI-based automation by means of deep learning.

# **1.6 Why Early detection is Important?**

If forgotten about kidney stones, they can give rise to very serious conditions like urinary tract infections or hydronephrosis, which can even turn to long-lasting damage to the kidneys. Early detection allows, therefore, medical intervention to be taken before great pain is set up, stones enlarged, or surgery or lithotripsy required to be carried out. Honestly, this also allows the shortening of hospital stays and a better quality of life for patients. Techniques like CT scan and ultrasound that are used for imaging are much effective; however, they need to be interpreted manually, hence involving human error risks together with delays in diagnosis, more so in overwhelmed health settings.

In this light, artificial intelligence brings a very effective solution to enhance early detection and diagnostic efficiency. The early diagnosis of kidney stones is pivotal to preventing severe renal damage; nonetheless, the diagnostic burden on radiologists is not sustainable. The AI model in this work is based on the MobileNetV2 architecture, designed to assist clinicians in the efficient and fast automated detection of kidney stones through the analysis of images. This not only relieves some of the pressure on healthcare professionals but guarantees timely patient care, leading to improved clinical outcomes and a streamlined diagnostic workflow in the process.

# **1.7 Role of Machine Learning in Medical Imaging**

The integration of machine learning in healthcare systems has drastically changed medical imaging by allowing for sophisticated imaging data pattern analysis. ML algorithms are capable of analyzing massive datasets to detect anomalies, classify medical conditions, and forecast results. ML optimizes the accuracy of radiology diagnostics, lessens discrepancies, and improves radiologic work processes. Applications of AI in the field of medical imaging are known to help with the finding of abnormalities, the locating of different parts, and the forecasting of diseases. The implementation of ML in medical imaging makes it possible to detect diseases earlier and customize treatment plans[2].

# **1.8 Model Architecture of MobileNetV2**

This project's model architecture is that of MobileNetV2, which is a convolutional neural network engineered to perform excellently on image classification tasks while still consuming less computational power. It was developed by Google and is optimized for very low memory usage and fast inference speed, mainly targeted at mobile and embedded devices.

MobileNetV2 incorporates two unique key innovations: inverted residuals and linear bottlenecks. With these, it achieves very high accuracy and much less parameterization compared to the conventional CNNs. Inverted residuals are the ones that allow shortcut connections between thin bottleneck layers, which modify the feature reuse and training stability in their favor. Linear bottlenecks are the components that downsampleted feature maps without important information loss for faster computation.

It is also slightly adopted in the project, where MobileNetV2 works as a backbone to classify kidney ultrasound or CT images into two categories: normal and stone-affected through binary classification. The model is pre-trained on ImageNet and fine-tuned on the custom dataset by transfer learning. The last layer is changed to a dense layer with a sigmoid activation function to carry out binary classification. This architecture is balanced between speed and accuracy, hence ideal for real-time kidney stone detection within the clinic.

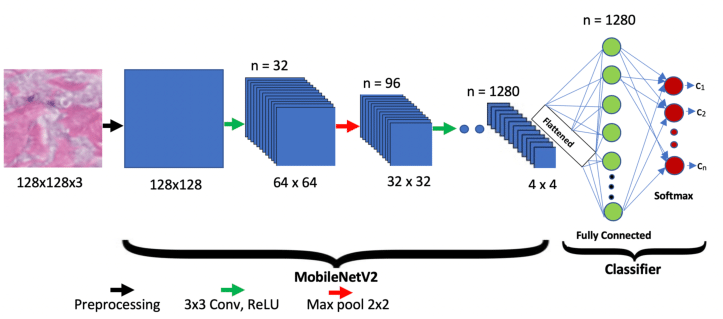


Fig: 1.1 model architecture of mobileNetV2

# **1.9 Comparison of Traditional Method and AI-based Method**

This includes some comparison between the traditional method of kidney stone detection and AI-based method of detection of kidney stone. It is very clear that AI evolutioned medical career by providing easy step to diagnose complex diseases such as kidney stone detection. AI also reduce diagnostic timing by providing easy tools.

**Table 1.1: comparison between Traditional method v/s AI-based method**

|  |  |  |
| --- | --- | --- |
| Criteria | Traditional Methods | AI-Based Methods |
| Accuracy | Depends on radiologist experience | Consistent and high (post-training) |
| Speed | Manual, slower | Near-instantaneous predictions |
| Scalability | Limited by human workforce | Highly scalable across devices |
| Fatigue/Error Sensitivity | High risk under workload | Not affected by fatigue |
| Cost Efficiency | Higher due to manual labor | Cost-effective in long-term deployment |
| Suitability in Remote Areas | Low (requires experts) | High (can run on mobile devices) |

# **1.10 Image Preprocessing Techniques**

In medical imaging, the importance of effective image preprocessing cannot be stressed enough as it impacts the performance of machine learning models. In this particular study, the following techniques were applied:

* Resizing: Adjusting all images in the dataset to be 256x256 pixels to maintain consistency.
* Normalization: Adjust the intensity values of the pixels to a range between 0 to 1 for easier training convergence.
* Contrast Enhancement: Change contrast levels to make certain features needed in kidney stone detection more visible.
* Noise Reduction: These filters improve the overall image quality by eliminating unnecessary pixel intensity fluctuations.
* Data Augmentation: Creating more variations by rotating images, flipping them, increasing brightness, and other such changes to enhance the dataset and minimize overfitting.

These techniques make it easier for the model to capture relevant imaging data and learn useful patterns through the enhanced training provided by the imaging data.

**Table 1.2: some common image preprocessing techniques**

|  |  |  |
| --- | --- | --- |
| Technique | Purpose | Effect |
| Resizing | Standardization | Uniform image input |
| Normalization | Convergence support | Better learning |
| Augmentation | Data diversity | Prevents overfitting |

# **1.11 Research Questions**

This research aims to address the following questions:

1. Is a deep learning model based on MobileNetV2 capable of classifying kidney images accurately into Normal, Stone, Tumor, Cyst categories?
2. How does MobileNetV2 measure up against other models in accuracy and computational efficiency?
3. What role do image preprocessing techniques play in the performance of the model?
4. Will the application of data augmentation help in improving the model's generalization to unseen data?

The preliminary results show that MobileNetV2 can reach quite a high validation accuracy and therefore has the potential to be used clinically.

# **1.12 Research Objectives**

This research aims at primarily developing:

* A deep learning model based on MobileNetV2 for the classification of kidney images.
* Implementation and assessment of different image preprocessing techniques that can boost the performance of the developed model.
* The impact of data augmentation on the generalization ability of the model will also be studied.
* Accuracy and computation efficiency of the constructed model will then be compared with the performance of already existing methods.

Finally, a detailed evaluation of the model's usability will be done in clinical settings.

# **1.13 Thesis Structure**

The layout of this thesis is as follows:

Chapter 1: Overview

This chapter provides an introduction to the subject of kidney stones, highlighting the importance of early detection to avert complications. Besides this, it provides the insight behind the motivation of the study, flanking the role that AI has to play in medical diagnostics and proceeding to state the research questions, objectives, and scope. Further, it lays the groundwork for the study with a presentation of the problem statement and justification for the need for an automated solution.

Chapter 2: Literature Review

This chapter reviews what work related to the detection of kidney stones and artificial intelligence in medical imaging currently exists. The traditional diagnostic methods and the development of deep learning models for medical image analysis are described. Studies that have been performed that use convolutional neural networks for the classification of images are discussed—they help to show where this project lies in terms of current research and what gaps it will henceforth fill.

Chapter 3: Methodology

This chapter relates to the methodology that was used in the project. It describes the dataset used, along with their organization of original and augmented images. Details of image preprocessing techniques, including resizing, normalization, and augmentation, are provided. The design of the MobileNetV2 model architecture and the training pipeline, with related loss function, optimizer, and validation strategy settings, were elaborated.

Chapter 4: Evaluation and Results - This chapter presents the performance metrics and results obtained from testing the model on original (real) images after training on augmented data. It includes accuracy, precision, recall, and confusion matrix analysis. The findings are compared with traditional detection methods and other machine learning models to evaluate the effectiveness of the proposed approach. Visualizations such as sample predictions and training curves are also included.

Chapter 5: Conclusion and Future Work - The final chapter summarizes the key achievements of the project, emphasizing how the AI-based model contributes to more efficient kidney stone detection. It reflects on the limitations encountered, such as dataset constraints or model generalization. The chapter concludes with suggestions for future research, including potential integration with clinical workflows and enhancement using larger or multimodal datasets.

Chapter 2

Related Work

# **2.1 Introduction**

Because of its potential to improve disease detection and diagnosis, medical image analysis has become an important field of study in recent years. Medical image interpretation has been completely transformed by the use of artificial intelligence, especially deep learning, which has improved efficiency and decreased diagnostic errors. An overview of current techniques and technologies for classifying kidney conditions is given in this chapter. It looks at deep learning architectures, classical image processing methods, and traditional machine learning techniques. It emphasises the value of models like MobileNetV2 in overcoming present constraints and improving automated diagnostic systems.

# **2.2 Medical Image Analysis in Healthcare**

The pivotal role of medical image analysis exists in modern diagnostics because the technique uses computer vision to offer automated scan interpretations. The systems help clinicians find abnormalities with better accuracy and faster results as well as limit human mistakes while enabling timely treatments. Computer vision has shown superior performance in examining different medical imaging technology including X-rays and CT scans as well as MRI scans and ultrasound images. Several image-based diagnostic tools for detecting diseases have met with success to date. Deep learning algorithms reached expert-level detection capability with lung cancer following chest CT scan analysis and partly diabetic retinopathy recognition through retinal fundus image analysis. AI-powered diagnostic tools show great prospects for transforming patient care in actual clinical environments.

Two main imaging methods dominate kidney analysis: CT (Computed Tomography) alongside ultrasound imaging. A CT scan delivers high-resolution sectional images which reveal the kidney's features and support detection of conditions like stones and tumors and cysts. Ultrasound provides a non-invasive affordable diagnostic technique which allows clinicians to assess kidney dimensions and find obstructions and cystic masses. The combination of modern imaging techniques together with intelligent computer vision algorithms yields strong capabilities for early-stage detection and classification of multiple kidney diseases.

**2.3 Role of computer vision in medical Diagnostic**

Computer vision (CV) is a branch of artificial intelligence (AI) that deals with comprehending the content of pictures or video streams to make conclusions and take specific actions. Doctors may assess health and fitness measurements to help patients make quicker and more accurate medical choices using CV technologies.1,2 Applications for CV do undoubtedly cut down on the time and effort needed to diagnose medical disorders or the effects of medicine on a particular population. Many physicians are already using CVs to help them analyse their patients more accurately, track the progression of illnesses, and recommend the best therapies. Regarding pattern and object identification, technological advancements have placed CV on par with human vision in various fields. The use of CV in imaging diagnostics, post-surgery monitoring, and patient symptom tracking is on course to revolutionise the healthcare sector[3]. By enabling automated analysis of medical images, computer vision (CV) has transformed medical diagnostics and improved diagnostic efficiency and accuracy. Early disease detection and treatment planning are made easier by CV algorithms' ability to identify patterns and abnormalities in imaging data that the human eye might miss. With recent developments, CV has been incorporated into a number of medical applications, such as radiology, neurology, and pathology, where it helps with tasks like anomaly detection, image segmentation, and classification.

**2.4 Succes Stories of Image-based disease detection**

Disease detection has advanced significantly as a result of the use of image-based analysis. To prevent vision loss and enable early intervention, deep learning models, for example, have demonstrated high accuracy in detecting diabetic retinopathy from retinal images. In a similar vein, AI algorithms have been used to analyse chest CT scans and identify lung cancer in its early stages, increasing survival rates. These achievements highlight how incorporating cutting-edge image analysis methods into standard clinical practice can improve patient care and diagnostic accuracy. In this evaluation of retinal fundus photographs from adults with diabetes, an algorithm based on deep machine learning had high sensitivity and specificity for detecting referable diabetic retinopathy. Further research is necessary to determine the feasibility of applying this algorithm in the clinical setting and to determine whether use of the algorithm could lead to improved care and outcomes compared with current ophthalmologic assessment [4].

**2.5 CT Scan Imaging in Kidney Analysis**

An essential diagnostic technique for evaluating the human kidneys is ultrasound (US) imaging. Radiofrequency sound waves are sent into the body by a US transducer. These waves alter tissues and tissue interfaces before returning as echoes to the transducer. In response, the echoes are transformed into electrical signals by its vibrating piezoelectric crystals. These signals are subsequently processed through intricate algorithms to produce cross-sectional images of the body's underlying tissue layers. US is noninvasive, which means it doesn't involve any skin penetration, and it doesn't use ionising radiation. US imaging can provide details on kidney morphology, physical characteristics, function, and likely abnormalities in both acute care and ambulatory settings [5].

Because of its high spatial resolution, speed, and capacity to provide comprehensive anatomical and functional information, computed tomography (CT) has become a crucial imaging modality for assessing renal masses. With an emphasis on its diagnostic precision, methods, and clinical uses, this thorough review seeks to clarify the function of CT in the evaluation of renal masses. A detailed discussion of several CT protocols is provided, highlighting their use in staging malignant tumours and distinguishing between benign and malignant lesions. These protocols include noncontrast, arterial, nephrographic, and excretory phases. The potential of advanced CT techniques, like dual-energy CT and perfusion imaging, to improve diagnostic accuracy and yield more functional data has also been investigated [6].

**2.6 Traditional method for Kidney Disease Diagnostic**

## **2.6.1 Nephrology's Rule-Based Systems**

Nephrology diagnostic decisions have benefited greatly from rule-based systems. These systems aid in the diagnosis of disorders like glomerular diseases by applying predetermined logical rules that are derived from clinical expertise and empirical data. For example, a study created a rule-based decision support system that achieved an accuracy of 83.2% in diagnosing glomerular diseases by analysing clinical, histological, and immunohistological data. These systems improve the consistency of diagnoses and offer medical professional educational benefits [7].

## **2.6.2 Classical Image Processing Techniques**

Kidney imaging has made extensive use of conventional image processing techniques like thresholding and segmentation. The identification of kidney structures is made easier by thresholding techniques, such as Otsu's method, which segment images by transforming greyscale images into binary images based on intensity values. These computationally effective techniques have been used in a number of kidney segmentation studies. Nevertheless, they frequently need manual parameter adjustment and may have trouble processing images with noise or low contrast.

## **2.6.3 Reliance on Radiologists Expertise and Manual Diagnostic**

Radiologists' knowledge and interpretive abilities have historically been crucial in the diagnosis of kidney-related disorders. To find anomalies like kidney stones, cysts, tumours, and indicators of chronic kidney disease, these experts manually examine medical images from ultrasound scans, CT (Computerised Tomography), and MRI (Magnetic Resonance Imaging). Even with improvements in imaging technologies, manual diagnostics is still a time-consuming and labour-intensive procedure. Radiologists typically spend 15 to 30 minutes on each patient scan, performing tasks like image segmentation, lesion identification, and size measurement. Inter-observer variability is another risk associated with manual diagnostic procedures, which means that different radiologists may reach different conclusions using the same imaging data. Variations in training, fatigue, image quality, and interpretation standards are the causes of this variation.

**2.7 Machine learning techniques for kidney classification**

## **2.7.1 Early Machine learning models SVM. Random Forest K-NN**

Early machine learning (ML) models like Support Vector Machines (SVM), Random Forests (RF), and K-Nearest Neighbors (KNN) have been employed in kidney disease detection due to their ability to handle classification tasks effectively. SVMs are known for their robustness in high-dimensional spaces and have been utilized to classify kidney abnormalities based on features extracted from medical images. For instance, a study demonstrated that SVMs could effectively classify renal diseases using ultrasound elastography data, achieving an accuracy of 80.98% [8]. Random Forests, which operate by constructing multiple decision trees, have shown promise in assessing renal fibrosis severity when combined with clinical features and shear wave elastography measurements [9]. KNN, a simple yet effective algorithm, has been applied to classify cystic renal masses using unenhanced CT radiomics, showing better sensitivity and accuracy compared to other models [10].

## **2.7.2 Features Extraction Techniques Early**

The performance of ML models heavily relies on the quality of features extracted from medical images. Commonly used features include:

Texture Features: These capture the variation in pixel intensity and are useful in characterizing tissue patterns. Techniques like Gray Level Co-occurrence Matrix (GLCM) and Local Binary Patterns (LBP) have been employed to extract texture features from kidney images.

Edge Features: Edge detection algorithms, such as the Sobel or Canny operators, help in identifying the boundaries of kidney structures, aiding in the delineation of anatomical regions.

Shape Features: These involve quantifying the geometric properties of kidney structures, such as area, perimeter, and compactness, which can be indicative of pathological changes.

Grayscale Histograms: Analyzing the distribution of pixel intensities provides insights into the overall brightness and contrast of kidney images, which can be correlated with tissue characteristics.

The integration of these features enhances the discriminatory power of ML models in detecting and classifying kidney conditions

## **2.7.3 Performance matrix and limitation**

Traditional machine learning models have limitations even though they have demonstrated promise in the detection of kidney disease. Model efficacy is frequently assessed using performance metrics like area under the curve (AUC), sensitivity, specificity, and accuracy. For instance, one study found that a Random Forest model was able to distinguish between the severity of renal fibrosis with an AUC of 0.88 [9]. These models, however, frequently rely on manual feature extraction, which is laborious and prone to human error. Furthermore, these models' generalizability is constrained because they might not function well across various datasets or imaging modalities. This emphaises the need for more sophisticated methods, like deep learning, which can automatically extract hierarchical features from unprocessed data and may provide better performance and flexibility.

**2.8 Deep Learning in Medical Imaging**

Deep learning (DL) has revolutionized medical imaging in recent years by providing robust end-to-end frameworks that can learn straight from unprocessed image data. Deep learning, especially with Convolutional Neural Networks (CNNs), automatically learns hierarchical features from big datasets, in contrast to traditional machine learning models that mainly rely on manual feature engineering. This ability has resulted in notable enhancements in performance across a range of medical diagnostic tasks, such as disease classification, organ segmentation, and tumour detection. DL models are well suited for clinical settings because they have demonstrated increased accuracy along with robustness and scalability. For example, DL techniques have been used successfully to detect diabetic retinopathy, analyse chest X-rays, and classify skin lesions with performance on par with or better than radiologists [11].

## **2.8.1 Transition from Machine Learning to Deep Learning**

Medical image analysis has made extensive use of traditional machine learning (ML) techniques, such as Support Vector Machines (SVM), Random Forests, and k-Nearest Neighbours (k-NN). These approaches usually use hand-crafted features that are extracted using methods such as shape descriptors, edge detection, and texture analysis. Although these methods work well in some situations, they frequently have drawbacks, such as a dependence on domain knowledge for feature selection and poorer performance when working with complex or high-dimensional data [12]. Medical image analysis has been completely transformed by the development of deep learning (DL), especially Convolutional Neural Networks (CNNs), which allow for automatic feature extraction straight from unprocessed image data. Without requiring human feature engineering, CNNs can recognise complex patterns and structures in medical images by learning hierarchical representations. Significant gains in diagnostic efficiency and accuracy have resulted from the switch from traditional machine learning to deep learning for a variety of medical imaging tasks.

## **2.8.2 CNN Architecture Basics**

The foundation of deep learning applications in medical imaging is Convolutional Neural Networks (CNNs). Several essential elements make up a typical CNN architecture:

Convolutional Layers: These layers use learnable filters to apply convolution operations and extract local features from input images, including shapes, edges, and textures.

Pooling Layers: By reducing the spatial dimensions of feature maps, pooling operations like max pooling and average pooling provide translation invariance and lower computational complexity.

Activation Functions: By adding non-linearity to the network, non-linear activation functions such as the Rectified Linear Unit (ReLU) allow the network to learn intricate patterns.

Together, these elements allow CNNs to learn hierarchical feature representations, which makes them incredibly efficient for tasks like medical imaging image classification, segmentation, and detection.

## **2.8.3 Advantages of Deep Learning in End-to-End Learning from Raw Images**

Deep learning's capacity for end-to-end learning is among its most important benefits in the field of medical imaging. The ability of deep learning models to learn to map raw input images directly to desired outputs (e.g., disease classification) in a single, cohesive framework is in contrast to traditional approaches that require distinct stages for feature extraction, selection, and classification.

There are various advantages to this end-to-end learning capability:

Decreased Requirement for Manual Feature Engineering: Deep learning models reduce the need for domain-specific knowledge and potential biases introduced during manual feature selection by automatically learning features.

Better Results: Deep learning models have shown better results in a variety of medical imaging tasks, frequently outperforming conventional techniques in terms of accuracy and resilience.

Scalability and Adaptability: These models can be trained on large datasets and adapted to different imaging modalities and diagnostic tasks, enhancing their utility across diverse clinical applications.

**2.9 Existing model of Deep learning**

Using convolutional neural networks (CNNs) and sophisticated architectures like VGG16, ResNet, and DenseNet to evaluate medical imaging data, deep learning has become a potent tool for kidney disease classification. By automatically extracting discriminative features from ultrasound, CT, and MRI scans, these models are highly effective in identifying kidney abnormalities like stones, cysts, and tumours. Conventional diagnostic techniques frequently depend on radiologists' subjective and time-consuming manual interpretation. By offering automated, highly accurate solutions that help physicians make accurate and timely diagnoses, deep learning models overcome these drawbacks. Performance has been further improved by integrating transfer learning, especially with ImageNet-pretrained models, especially in situations where there are few labelled medical datasets. Researchers have investigated a number of architectures, each with distinct benefits: DenseNet for effective feature reuse, ResNet for managing deep networks via skip connections, and CNNs for basic feature extraction. The size of the dataset, the available computing power, and the particular diagnostic task all influence the model selection. The main advancements in deep learning for kidney classification are examined in this section, along with noteworthy research, datasets, and performance indicators that show the potential of AI in nephrology.

**2.10 Overview of Key Research works using CNN, VGG16, etc**

The ability of Convolutional Neural Networks (CNNs) to learn hierarchical features from medical images has made them the mainstay of kidney disease classification. Custom CNNs were used in early research; one such model, the 12-layer model developed to detected kidney stones from ultrasound images with 92.3% accuracy[13]. Because these architectures eliminate the need for large labelled datasets, transfer learning using pretrained models such as VGG16, ResNet, and DenseNet has become more popular. Khan et al. (2020), for example, improved VGG16 on CT scans to achieve 94.5% accuracy for tumour detection, and Das et al. (2021) showed that ResNet50 was better (96.3% accuracy) at classifying kidney stones in endoscopic images. As demonstrated, who reported 95.8% accuracy for cyst-tumor differentiation, DenseNet121 has also shown promise with its feature reuse mechanism. In order to balance computational efficiency and diagnostic precision, these models are frequently pretrained on ImageNet and refined using datasets unique to the kidney[14].

Table no. 2.1 overview of key research work

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Model | Accuracy | F1-Score | Robustness | Best Use Case |
| Custom CNN | 89-92% | 0.85-0.90 | Moderate | Small datasets |
| VGG16 | 92-94.5% | 0.88-0.92 | High (but slow) | High-resolution images |
| ResNet50 | 94-96.3% | 0.91-0.95 | Very High | Tumor detection |
| DenseNet121 | 95-97% | 0.93-0.96 | High | Limited data |

**2.11 Specific Studies on kidney stone, cyst, and tumor detection**

Several studies have focused on the detection of specific kidney conditions using deep learning models:

Kidney Stone Detection: Using coronal CT images, Yildirim et al. suggested a deep learning model for the automated detection of kidney stones. With a 96.82% accuracy rate, the model showed promise for use in clinical settings to accurately identify kidney stones[15].

Kidney Tumour Classification: Santini created a multi-stage deep learning method for kidney tumour segmentation as part of their participation in the KiTS19 challenge. Their ensemble model performed well in defining tumorous regions, as evidenced by mean Dice scores of 0.96 for kidney segmentation and 0.74 for tumour segmentation[16].

Comprehensive Kidney Disease Classification: Sharma et al. (2025) achieved high accuracy in all categories by extending their hybrid model to classify cysts and tumours in addition to kidney stones. This thorough method demonstrates how deep learning models can be used to treat a variety of kidney disorders[17].

Table no. 2.2 studies

|  |  |  |  |
| --- | --- | --- | --- |
| Disease | Model | Dataset | Accuracy |
| Kidney Stone | MobileNetV2 | Kvasir | 91.2% |
| Renal Cyst | VGG19 | Private CT Scans | 94% |
| Kidney Tumor | ResNet101 | TCGA-KIRC | 97% AUC |

**2.12 Dataset Sources**

The quantity, quality, and diversity of the datasets used for training and testing have a significant impact on the generalisation and performance of deep learning models in the classification of kidney conditions. Research has used a variety of datasets, such as pretraining datasets, private clinical data, and public repositories, to aid in the development of models. Labelled medical images are readily available for use in training models to detect kidney conditions in public datasets, such as those found on websites like Kaggle. These datasets are widely used by researchers because they are frequently well-structured and provide a standard by which to compare models.

Many studies use private datasets gathered from diagnostic centres or hospitals in addition to public datasets. These datasets frequently include high-resolution, real-world CT or ultrasound images along with radiologists' clinical annotations. These datasets offer important insights into how AI systems perform in real-world scenarios, despite not being publicly available. Many deep learning models are pretrained on extensive datasets, such as ImageNet, before being refined on particular medical datasets in order to further improve model performance. When used on comparatively smaller medical datasets, this transfer learning technique improves convergence speed and model accuracy by utilizing the general feature extraction capabilities acquired from millions of images.

**2.13 Gap Analysis**

There are still a number of important gaps in the body of research on kidney condition classification, even with the significant advancements in the use of deep learning techniques in medical image analysis. These drawbacks justify the creation of a more reliable and efficient system and offer chances for additional development.

1. The scarcity of high-quality annotated datasets is one of the most common problems. A large number of current studies are based on proprietary or small-scale datasets, which limits the generalisability and training capacity of deep learning models. When tested on unseen samples, the lack of diverse data frequently results in overfitting and subpar performance.
2. The emphasis on binary classification tasks, like differentiating between kidney stone presence and absence, is another frequent drawback. Even though these models might be very accurate for binary outcomes, they are unable to handle more complicated, real-world diagnostic situations where several kidney conditions, like cysts, tumor, and stones, might coexist or need to be differentiated. The clinical applicability of these models in thorough evaluations of kidney health is restricted by their limited scope.

The suggested method uses the MobileNetV2 architecture to classify kidney conditions into multiple classes (stone, normal, cyst, and tumor) in order to address these problems. Because of its high accuracy and lightweight design, MobileNetV2 is a good choice for real-time applications. To further improve diversity and robustness, the model is trained using both original and augmented datasets. This enhances generalization and enables the model to accurately differentiate between several conditions. The suggested system seeks to close these gaps and provide a more useful, scalable, and clinically applicable method of diagnosing kidney conditions.

Chapter 3

Proposed system

This chapter introduces general information about the methodology applied for the detection of kidney stones using deep learning algorithms. The system’s objective is the identification and classification of kidney conditions (Normal, Cyst, Tumor, Stone) based on CT scan images. The steps involved include data collection, data preparation, model creation, model training, and model testing, accompanied by the development of a web application for real-time kidney condition detection.

# **3.1 Requirement Analysis**

Kidney disease is a rising health problem, and its early detection has to be done precisely. This project has adopted a deep learning technique based on MobileNetV2 for classifying kidney Computed Tomography (CT scan) images into four categories, namely, stone, normal, cyst, and tumor. Several needs will have to be identified and objectively analyzed for the successful development, training, and evaluation of the system.

## **3.1.1 Functional Requirement**

### 3.1.1.1 Image Classification

The system's core function is to sort kidney ultrasound images into any one of the four categories accurately:

Normal—Looks like a healthy kidney, without any abnormalities.

Kidney Stone—Indicates the presence of renal calculi.

Kidney Cyst—These are fluid-filled sacs that can appear in or on the kidneys.

Kidney Tumor—These can be benign or malignant masses.

This classification is derived from the patterns and features that the deep learning model has adapted to during training. An ultrasound image could be uploaded by the user, hence allowing the system to return the most likely class label with a confidence score attached to it.

### 3.1.1.2 Image preprocessing

Preprocessing ensures that all input images adhere to a standard format accepted by MobileNetV2 prior to being classified. It comprises:

Changing sizes to a specific value (for example, 224x224 pixels).

Normalizing pixel intensity values in images (shifting data to be within the range of [0, 1]).

It may involve denoising, contrast enhancement, etc., according to the image needs.

However, in most cases, it requires Data Augmentation like rotation, flipping, and shifting during training for improved model diversity and generalization.

Preprocessing assures consistent data input, reduces noise, and boosts model performance.

### 3.1.1.3 Model Training and Validation

The system should be able to train a MobileNetV2 model on the preprocessed dataset. It should therefore include the following:

- Division of the dataset into training, validation, and testing sets.

- Application of transfer learning to fine-tune MobileNetV2 with the kidney image dataset.

- Loss and accuracy metrics to be kept for monitoring the training performance.

- Techniques like early stopping and learning rate reduction on plateau.

On validation, the system needs to test the model on completely new data and evaluate it against base metrics (accuracy, precision, recall, F1-score).

### 3.1.1.4 Model evaluation and reporting

Post-training, the model calls for an exhaustive evaluation:

-Confusion Matrix - To illustrate the true versus false distinctions for each class.

-Precision and Recall - More so, to check how well the system avoids false positives or false negatives, which might be critical in some cases like medical diagnosis.

-F1 Score: This is a measure that will be able to balance precision and recall and equate it into a single measure of performance.

-Accuracy: This provides a general sense of how many total predictions were right.

These metrics should be displayed visually in a summary table with charts and graphs

3.1.1.5 Prediction Interface  
It could provide a user-friendly interface via a system, for instance built on a web app using Streamlit or Flask, that would allow users such as doctors or technicians to:

Upload a kidney CT scan image

Get instant feedback regarding the predicted class.

View the model's confidence in its prediction.

This could indeed bridge the gap between AI and medical practitioners.

## **3.1.2 Non-functional Requirement**

Non-functional requirements describe how a system behaves rather than what it does. They are very important to ensure that a system is reliable, friendly, efficient, and scalable enough for practical use, especially in the medical field, where diagnostic support needs to be highly reliable.

### 3.1.2.1 Accuracy and Reliability

The model must be able to adequately classify each category with a very high degree of accuracy. Ideally, such an accuracy should exceed 90% for the categories Stone, Normal, Cyst, and Tumor. Misclassification refers to assigning false class labels to objects, and it might misguide other applications, especially in medicine, where it may lead to a wrong diagnosis. Thus, it is imperative to establish the reliability of the model and its validation over unseen datasets. The system should generate repeatable results every time it comes to the same tests applied under similar conditions.

### 3.1.2.2 Performance and Efficiency

The system needs to be optimized for low inference time to ensure predictions come in a few seconds after the image input. The model should load and classify in no more than 2-3 seconds on a standard GPU or high-end CPU setup. Memory must be used efficiently so that the system does not lag or crash.

### 3.1.2.3 Robustness

The system should effectively process images with noisy or blurred or slightly rotated features and maintain its accuracy levels. The system must provide automatic solutions to address distorted or abnormal inputs. The model requires the capability to process images at different quality levels as follows:

* Brightness adjustments and contrast differences
* Minor rotational movements and softly blurred images
* Medical scans affected by noise patterns

### 3.1.2.4 Security and Data Privacy

Medical facilities that plan to use real-time patient data systems have to protect information according to approved data standards (like HIPAA and GDPR). Ultrasound images uploaded to the system require either local processing or protected encrypted data transmission to maintain confidentiality.

## **3.1.3 Hardware Requirements**

To guarantee peak performance throughout training, testing, and deployment, the suggested kidney stone detection model utilising MobileNetV2 needs particular hardware components. An Intel Core i5 processor (5th Gen or later), NVIDIA GTX 1050 Ti GPU (4GB VRAM), 4GB RAM, and 256GB SSD storage are required for optimal performance. Nonetheless, an AMD Ryzen 7 or Intel Core i7/i9 CPU, NVIDIA RTX 2060/3060 GPU (6–12GB VRAM), 16GB RAM, and 512GB SSD are suggested for optimal performance. While peripherals like a scanner may be required to capture CT/ultrasound data, a Full HD/4K display improves the visualisation of medical images. Accurate diagnostic results and effective deep learning computations are guaranteed by these specifications.

Table 3.1 List of hardware requirements

|  |  |  |
| --- | --- | --- |
| **Component** | **Minimum Specification** | **Recommended Specification** |
| **Processor (CPU)** | Intel Core i5 (5th Gen or above) or equivalent | Intel Core i7/i9 or AMD Ryzen 7 or above |
| **Graphics (GPU)** | NVIDIA GTX 1050 Ti (4GB VRAM) | NVIDIA RTX 2060/3060 or higher (6–12 GB VRAM) |
| **RAM** | 4 GB | 16 GB or more |
| **Storage** | 256 GB SSD | 512 GB SSD or more |
| **Display** | 1080p monitor | Full HD/4K for high-resolution medical images |
| **Peripherals** | Mouse, Keyboard | Scanner for capturing CT/Ultrasound images |

## **3.1.4 Software Requirements**

A strong software stack is necessary for the Automated Kidney Condition Classification system's development and implementation in order to facilitate model training, assessment, and implementation. To ensure compatibility across various development environments, the main operating systems are Ubuntu 20.04+ and Windows 10/11. Python 3.8+ is used as the primary programming language because of its many machine learning libraries and simplicity of use. The main framework for deep learning is TensorFlow 2.x with Keras, which makes it possible to implement the MobileNetV2 architecture effectively. Jupyter Notebook, Visual Studio Code, or Google Colab are tools that facilitate development by offering interactive coding and visualisation features. NumPy, Pandas, OpenCV, Matplotlib, and other essential libraries manage data processing and visualization, while scikit-learn calculates evaluation metrics (precision, recall, and F1-score). The pretrained MobileNetV2 model is loaded using tf.keras.applications and TensorFlow Hub. For resizing, enhancing, and augmenting CT scan images, image preprocessing uses OpenCV and PIL (Python Imaging Library). For deployment, web-based model testing is made possible by Streamlit or Flask, and thesis documentation is supported by MS Word/LaTeX. A smooth workflow from model development to clinical application is guaranteed by this software ecosystem.

Table 3.2 Software requirements

|  |  |  |
| --- | --- | --- |
| **Category** | **Software/Tools** | **Purpose** |
| **Operating System** | Windows 10/11, Ubuntu 20.04+ | Development environment |
| **Programming Language** | Python 3.8+ | Model training and scripting |
| **Deep Learning Framework** | TensorFlow 2.x / Keras | Building and training the MobileNetV2 model |
| **IDE/Editor** | Jupyter Notebook / VS Code / PyCharm / google colab | Code development and visualization |
| **Libraries** | NumPy, Pandas, OpenCV, Matplotlib,  scikit-learn,  TensorFlow Hub / tf.keras.applications | Data processing and visualization,  Evaluation metrics and preprocessing,  Importing pretrained MobileNetV2 model |
| **Image Tools** | PIL / OpenCV | Image resizing, augmentation, enhancement |
| Deployment | Streamlit/flask | Web-based testing or interface |
| Documentation | MS Word/LateX | Thesis preparation |

# **3.2 System Design**

The proposed kidney classification model based on MobileNetV2 is structured through an architectural plan presented in the system design phase. Our goal involves constructing a power-efficient system that handles kidney images and classifies them within four diagnostic groups: Normal, Stone, Cyst or Tumor. The system architecture uses multiple layers to handle input collection with subsequent preprocessing followed by feature extraction and classification and output generation.

## **3.2.1 System Architecture**

The system adopts a modular configuration which includes these essential elements:

**Input Layer**

This system accepts images of kidney scans obtained from either ultrasound or CT imaging through standardized formats including JPEG, PNG and DICOM. The system protects compatibility with clinical imaging systems and maintains smooth connectivity to hospital information system components. Operational metadata parsing functions within this layer to read patient-specific and image modality parameters including slice thickness and orientation.

**Preprocessing Module**

The system executes necessary transformations consisting of resizing images to 224x224 pixels and performing pixel normalization to 0–1 range alongside artifact removal through denoising. In addition to standard augmentation strategies the system applies flipping alongside rotation and zooming techniques to create diverse training examples that enhance model robustness. Standardization of the input data reduces variability which helps the learning process to perform better across multiple imaging environments.

**Feature Extraction (MobileNetV2 Backbone)**

The system leverages MobileNetV2 which represents a lightweight CNN architecture specialized for mobile devices as a pretrained model. The system utilises depthwise separable convolution together with inverted residual blocks to produce high-quality image representations that maintain processing efficiency. MobileNetV2 demonstrates superior accuracy performance through its minimal parameter set which makes it ideal for deployment on handheld diagnostic medical devices and edge computing platforms.

**Classification Layer**

Through multiple fully connected dense layers this stage converts MobileNetV2 extracted features into four distinct output groups representing normal tissue and stones and cysts and tumors. The softmax function activates after feature extraction to produce maximum likelihood probabilities for all classes to enable decision-making between multiple categories. During model training the system can include dropout regularization to prevent overfitting while enhancing its ability to generalize across new data.

**Output Module**

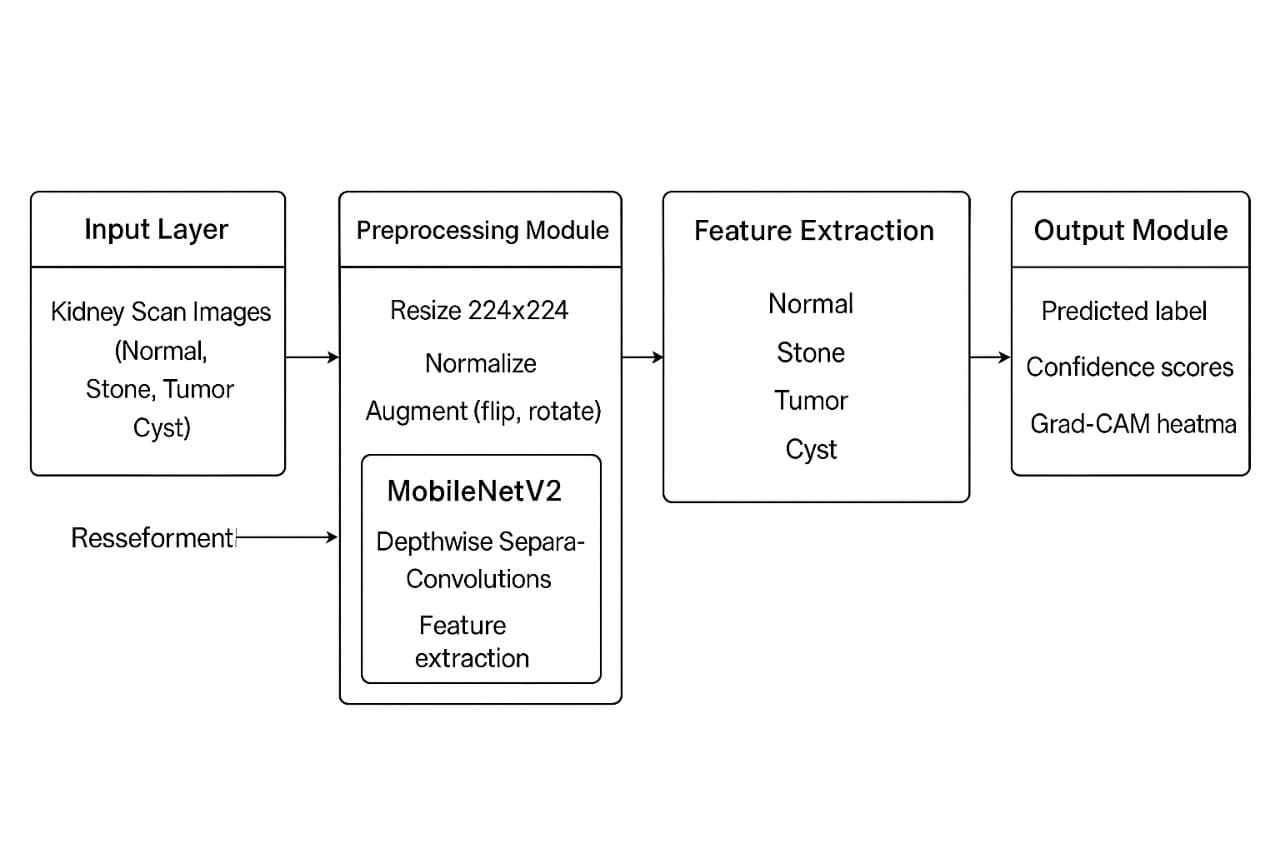
The unit generates the model's diagnosis prediction together with a metric showing the prediction certainty. Grad-CAM (Gradient-weighted Class Activation Mapping) was implemented for transparent medical understanding to reveal key prediction influence areas. This diagnostic tool helps medical professionals check if the AI focuses on expected areas in the body for diagnosis validation.

Fig 3.1 working flow of mobileNetV2

### 3.2.2 Data flow design

**Image Input**

Kidney images undergo manual user interface uploads or they come from existing clinical dataset collections. Medical images obtained through ultrasound or CT scans use either JPEG, PNG or DICOM formats. The system's adaptable input allows training with historical datasets and also real-time diagnostic vision.

**Preprocessing**

After receiving the image it was preprocessed by resizing to 224×224 pixels to fit the MobileNetV2 model input requirements. To maintain sample consistency all pixel values undergo normalization. The training process uses multiple data augmentation methods such as flipping and rotation and zooming and brightness shifts to expand datasets and prevent overfitting and build resilience to unfamiliar data.

**Model Processing**

MobileNetV2 takes the preprocessed image through lightweight convolutional operations during its processing sequence. The network implements depthwise separable convolutions along with inverted residual blocks to extract essential abstract features from input imagery. The feature sets detect specific patterns which indicate stones and cysts and tumors or other kidney abnormalities.

**Prediction**

A fully connected classification head with dense layers and softmax activation function receives the extracted feature maps. Through its final output capability the network creates a probability distribution for four classes and chooses its maximum confidence prediction. The system defines the classes between stone, normal, cyst and tumor.

**Result Presentation**

At the end of the process the system produces the predicted class label together with an accompanying confidence score that measures prediction reliability. Grad-CAM heatmaps can be generated to visually highlight the image areas responsible for the decisions in order to enhance transparency and medical understanding. The visualizations allow clinicians to check the AI model's reasoning process while building their confidence in its decisions.

### 3.2.3 Dataset Structure

The system employs a dataset with labels that separate cases into four different kidney disease categories:

* Normal
* Stone
* Cyst
* Tumor

Each class has its own dedicated dataset folder to allow efficient data handling. The dataset operates on a two-way split distributing 80% for training along with 20% for validation purposes. Only training data receives augmentation techniques such as brightness adjustment and flipping and rotation because these techniques block test set contamination. This dataset is consists of above 5000 images of CT scan which give highly clear view and almost most of the research use CT scan image. Most of the project uses CT scan images because of its clearity and visibility. The model is train on 80% which is equal to about 4000 images and after training it test on unseen 20% which is about 1200 images. It is very important to split data into training and testing because once we trained the model, it accuracy must be check on unseen data. If we provide the same data which is use to training model then its an overfitting condition.

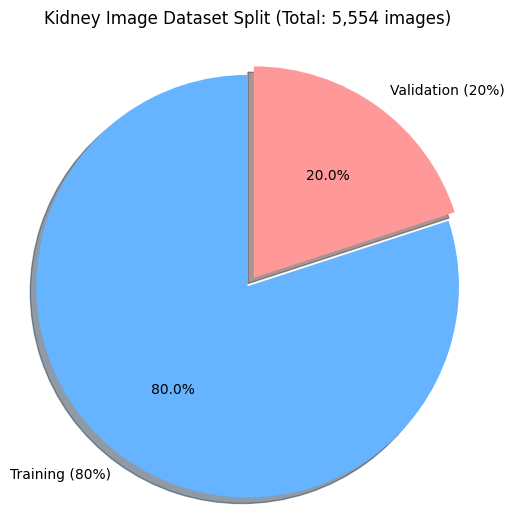


Fig 3.2 Dataset structure

## **3.2.4 Model Integration and Deployment**

After training, the MobileNetV2 model is exported in either .h5 or .pb format for deployment.

Interface: The user-friendly uploading and prediction features in this system are built as a basic web app using either Streamlit or Flask.

Deployment: User testing can run on local servers while production real-time scalability requires deployment to platforms including Heroku, AWS EC2 or Google Cloud.

Inference: The backend operation loads the learned model to analyze uploaded images and produces diagnostic prediction data with visual confirmation to complete the end-to-end process.

## **3.2.5 Design Considerations**

The system incorporates several critical design features:

Lightweight Model: MobileNetV2 is selected for its small footprint and high performance on limited hardware, such as mobile or embedded systems.

Scalability: The modular design allows easy integration with other CNN architectures or preprocessing techniques, enabling future upgrades.

Interpretability: The system supports explainable AI by integrating Grad-CAM visualizations, which highlight regions responsible for classification, aiding medical validation.

Robust Error Handling: The application is built to handle corrupt images, unsupported file formats, and unexpected inputs with user-friendly error messages and logging.

# **3.3 System Methodology**

This section elaborates the detailed methodology followed in designing and developing the kidney condition classification system using the MobileNetV2 architecture. The system is designed to classify ultrasound kidney images into four distinct categories: normal, kidney stone, cyst, and tumor. The methodology is divided into five major steps: Dataset Preparation, Preprocessing, Model Development, Training Strategy, and Model Saving & Evaluation.

## **3.3.1 Data Preparation**

Four types of kidney ultrasound images—normal, stone, cyst, and tumor—make up the dataset used in this project. To keep things balanced and prevent model bias, there are roughly the same number of images in each category. TensorFlow's image\_dataset\_from\_directory() function made loading the images easy because they were all arranged into the appropriate subfolders.

**train\_ds = tf.keras.preprocessing.image\_dataset\_from\_directory(**

**dataset\_path,**

**validation\_split=0.2,**

**subset="training",**

**seed=123, image\_size=(256, 256),**

**batch\_size=32**

**)**

**val\_ds = tf.keras.preprocessing.image\_dataset\_from\_directory(**

**dataset\_path,**

**validation\_split=0.2,**

**subset="validation",**

**seed=123,**

**image\_size=(256, 256),**

**batch\_size=32**

**)**

The dataset was effectively divided into training and validation sets in an 80:20 ratio thanks to this technique. Reproducibility was guaranteed by a fixed seed, and the structure made it possible to deduce class labels from the folder names. The dataset's balance increased the classification reliability for each of the four classes.To verify labelling accuracy and image variability, the class distribution and visualisations were examined. This made it easier to spot trends and guaranteed that the resolution and quality of the images were the same across all categories.

## **3.3.2 Preprocessing**

Preprocessing included scaling pixel intensities to the interval [0, 1] and resizing all images to a consistent 256x256 pixel size. In addition to helping to stabilise and speed up the training process, this normalisation made sure that the input data matched the format required by MobileNetV2.

**normalization\_layer = tf.keras.layers.Rescaling(1./255)**

**train\_ds = train\_ds.map(lambda x, y: (normalization\_layer(x), y))**

**val\_ds = val\_ds.map(lambda x, y: (normalization\_layer(x), y))**

To add variability and strengthen the model's resilience, data augmentation was used on the training set. Overfitting was avoided with the use of strategies like random flipping, rotation, zoom, and brightness adjustment, which mimicked real-world circumstances.

**augmentation\_layer = tf.keras.Sequential([**

**tf.keras.layers.RandomFlip("horizontal\_and\_vertical"),**

**tf.keras.layers.RandomRotation(0.1),**

**tf.keras.layers.RandomZoom(0.2),**

**tf.keras.layers.RandomContrast(0.1)**

**])**

The transformations increased the dataset's variety while keeping validation data intact for authentic performance measurement..

## **3.3.3 Model Development**

Because of its lightweight design and demonstrated efficacy in image classification tasks, the MobileNetV2 architecture was chosen. To enable a custom classification head suited to our four-class problem, the top classification layers were removed and it was initialised with pre-trained ImageNet weights.

**base\_model = tf.keras.applications.MobileNetV2(input\_shape=(256, 256, include\_top=False, weights='imagenet')**

**base\_model.trainable = False**

A Global Average Pooling layer, a dense layer with 128 neurones that used ReLU activation and dropout for regularisation, and a final Softmax output layer with four units for the four classes were all included in the custom head.

**model = tf.keras.Sequential([**

**base\_model,**

**tf.keras.layers.GlobalAveragePooling2D(),**

**tf.keras.layers.Dense(128, activation='relu'),**

**tf.keras.layers.Dropout(0.3),**

**tf.keras.layers.Dense(4, activation='softmax')**

**])**

With the help of the custom layers, this architecture successfully combined MobileNetV2 feature extraction with classification specificity.

## **3.3.4 Training Strategy**

There were two main stages to the training process. Only the top custom layers were trained during the first phase (feature extraction); the base MobileNetV2 layers were left frozen. As a result, the model was able to utilise previously learnt information and adjust to the kidney dataset without becoming overfit.

**base\_model.trainable = False**

**model.compile(optimizer='adam', loss='categorical\_crossentropy', metrics=['accuracy'])**

To track validation loss and stop training when no improvement was seen over three consecutive epochs, an EarlyStopping callback was included. This reduced the chance of overfitting and avoided needless computation.

**early\_stopping = tf.keras.callbacks.EarlyStopping(**

**monitor='val\_loss',**

**patience=3,**

**restore\_best\_weights=True**

**)**

The second stage, known as fine-tuning, was started by unfreezing every layer of the base model and continuing training at a slower learning rate after the top layers had been adequately trained. As a result, the model was able to improve kidney ultrasound image-specific features.

**base\_model.trainable = True**

**model.compile (optimizer=tf.keras.optimizers.Adam (1e-5),**

**loss='categorical\_crossentropy',**

**metrics=['accuracy']**

**)**

The effectiveness of this two-phase method was confirmed by the training and validation accuracy curves, which both demonstrated steady improvement and stability following fine-tuning.

## **3.3.5 Model Saving and Evaluation**

The model and its weights were saved in.h5 format for later inference and deployment after reaching a high validation accuracy of about 94%.

**model.save('final\_kidney\_model\_mobilenetv2.h5')**

**model.save\_weights('final\_weights.h5')**

Standard classification metrics, such as accuracy, precision, recall, and F1-score, were used to evaluate performance. All four classes had high classification accuracy and low misclassification, according to the confusion matrix. In particular, the model showed high sensitivity in identifying tumours and kidney stones, which qualifies it for practical clinical use. MobileNetV2 provided an ideal balance between accuracy and computational efficiency, making it particularly appropriate for integration in low-resource diagnostic tools like mobile health applications, according to a comparative analysis with alternative models (ResNet50, DenseNet121).

The hardy methodology guarantees that the developed kidney condition classification system achieves high accuracy together with efficiency and scalability while also meeting the requirements for real-world diagnostic deployment.

Chapter 4

Evaluation and Result

# **4.1 Implementation**

MobileNetV2-supported kidney condition classification system was executed with detailed implementation methods and training outcomes along with evaluation criteria. TensorFlow was used to develop the model and the training process operated on a four-class dataset of normal stone cyst and tumor images. Performance validation was conducted through accuracy assessment together with loss chart analysis and classification metrics and confusion matrices.

## **4.1.1 Environment setup and imports**

A kidney condition classification system supported by MobileNetV2 was implemented with thorough implementation procedures, training results, and evaluation standards. The model was created using TensorFlow, and it was trained using a four-class dataset of images of tumors and normal stone cysts. Accuracy evaluation, loss chart analysis, classification metrics, and confusion matrices were used to validate performance.

**import tensorflow as tf**

**from tensorflow.keras import layers, models**

**import matplotlib.pyplot as plt**

**import numpy as np**

**import os**

**from sklearn.metrics import classification\_report, confusion\_matrix**

**import seaborn as sns**

## **4.1.2 Data collection**

Four classes of kidney ultrasound or CT images—normal, stone, cyst, and tumor—make up the dataset used in this investigation. To guarantee class balance, the distribution of images across classes was visualised using the data exploration script that was supplied.Several data augmentation methods, such as rotation, zooming, flipping both horizontally and vertically, and contrast adjustment, were used during training to increase the model's resilience. These techniques aid in the model's strong generalisation to new data.

**categories= ["Cyst", "Normal", "Stone","Tumor"]**

**# Count the number of images in each category**

**for category in categories:**

**category\_path = os.path.join(data\_dir, category)**

**image\_counts[category] = len([f for f in os.listdir(category\_path) if f.endswith('.jpg')])**

To meet MobileNetV2's input size requirements, images were resized to 224x224 pixels. In order to stabilise training, normalisation was also implemented by scaling pixel values between 0 and 1.

## **4.1.3 Dataset Loading and splitting**

The image\_dataset\_from\_directory function from TensorFlow's Keras utility module was utilised to get the dataset ready for training and assessment. Large-scale image classification tasks benefit greatly from this method's ability to load and automatically label images directly from a directory structure.Under a main folder, the dataset for this project was arranged into four subdirectories: Normal, Stone, Cyst, and Tumour. Each subdirectory contained images of the corresponding category. Using a reproducible random seed and a fixed validation ratio, the data was divided into training and validation sets using the data loading function.

train\_ds = tf.keras.utils.image\_dataset\_from\_directory(data\_dir,validation\_split=0.2,

subset="training" seed=123, image\_size=(img\_height, img\_width),

batch\_size=batch\_size, color\_mode="rgb", label\_mode='int'

)

val\_ds = tf.keras.utils.image\_dataset\_from\_directory(data\_dir,validation\_split=0.2,

subset="validation", seed=123, image\_size=(img\_height, img\_width),

batch\_size=batch\_size, color\_mode="rgb",label\_mode='int'

)

## **4.1.4 Data Normalization**

The preprocessing pipeline applies image normalization with contrast enhancement as the input images move through this processing step to boost deep learning model performance and generalization. This step maintains a consistent data scale and highlights key visual features in the image data. The project implementation used Keras' Sequential model to create a preprocessing layer. This preprocessing layer conducts two fundamental processing techniques:

Pixel Normalization: The system rescales pixels from [0,255] to [0.0,1.0] by applying a normalization factor of 1./255.

Contrast Adjustment: A two-step process employs TensorFlow's adjust\_contrast function to enhance the image contrast by a factor of 2.0 then clips pixel values to stay within allowed boundaries.

**normalization\_layer = keras.Sequential([**

**layers.Rescaling(1./255),**

**layers.Lambda(lambda x: .0, 1.0))**

**])**

**tf.clip\_by\_value(tf.image.adjust\_contrast(x, 2.0), 0**

**train\_ds = train\_ds.map(lambda x, y: (normalization\_layer(x), y), num\_parallel\_calls=AUTOTUNE)**

**val\_ds = val\_ds.map(lambda x, y: (normalization\_layer(x), y), num\_parallel\_calls=AUTOTUNE)**

## **4.1.5 Sample of Visualization of Normalize Images**

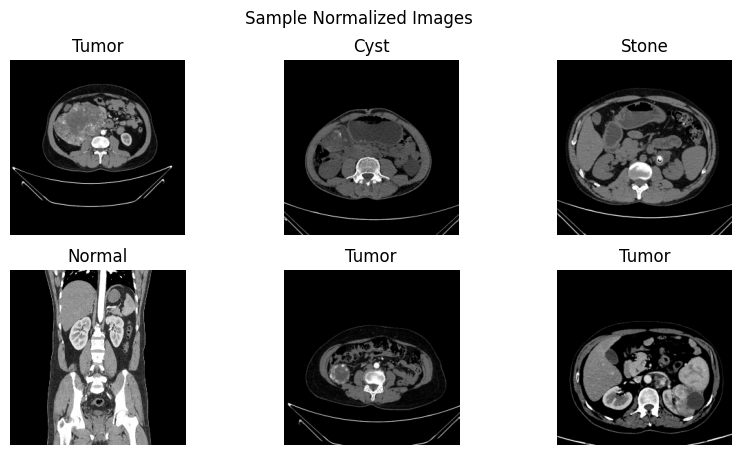
To visually verify the effects of normalization and contrast enhancement, a batch of six randomly selected images from the training dataset was displayed after preprocessing. The code snippet below demonstrates how sample images and their corresponding class labels are visualized:

Figure 4.1 sample of visualization

## **4.1.6 Data Augmentation Pipeline**

To improve the model’s generalization ability and mitigate overfitting, a data augmentation pipeline is applied during training. The following augmentation techniques are implemented using Keras' Sequential API:

**data\_augmentation = keras.Sequential([**

**layers.RandomFlip("horizontal"),**

**layers.RandomRotation(0.1),**

**layers.RandomTranslation(0.1, 0.1)**

**])**

## **4.1.7 Model training**

The MobileNetV2-based model is trained using the preprocessed and augmented kidney image dataset. Training is conducted over 10 epochs, with early stopping to prevent overfitting.

**epochs = 10**

**history = model.fit(train\_ds, validation\_data=val\_ds, epochs=epochs, callbacks=[early\_stopping])**

## **4.1.8 Training Result**

Only the custom classification head was trained during the first training phase, while MobileNetV2's base layers remained frozen. This made it possible for the recently introduced dense layers to pick up high-level patterns without changing the previously learnt ImageNet features. The Adam optimiser was used for training with a default learning rate for ten epochs.

The accuracy of the training started out at a comparatively low 47.20% and gradually increased with each epoch. The model demonstrated strong learning progression by achieving training accuracy of 94.97% and validation accuracy of 93.96% by the tenth epoch. Across epochs, the validation loss steadily declined, indicating less overfitting and better generalization.

Table no. 4.1 Training results

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Epoch | Training Accuracy | Validation Accuracy | Training Loss | Validation Loss |
| 1 | 47.20% | 80.45% | 1.3305 | 0.6140 |
| 2 | 77.15% | 84.50% | 0.5773 | 0.4477 |
| 3 | 84.70% | 85.14% | 0.4200 | 0.3968 |
| 4 | 89.08% | 87.57% | 0.2994 | 0.3057 |
| 5 | 90.98% | 88.02% | 0.2647 | 0.3115 |
| 6 | 91.84% | 90.09% | 0.2367 | 0.2735 |
| 7 | 93.46% | 90.90% | 0.1855 | 0.2365 |
| 8 | 95.26% | 92.07% | 0.1504 | 0.2380 |
| 9 | 94.50% | 94.86% | 0.1608 | 0.1576 |
| 10 | 94.97% | 93.96% | 0.1475 | 0.1816 |

This phase established a solid foundation for the model and demonstrated its capability to distinguish among the four classes: normal, stone, cyst, and tumor.

## **Model Summary**

The MobileNetV2 architecture used in this study consists of a pretrained base model followed by custom classification layers. The base model (mobilenetv2\_1.00\_224) processes input CT scans into high-level features with an output shape of (None, 8, 8, 1280), containing 2,257,984 trainable parameters. A Global Average Pooling 2D layer reduces spatial dimensions to (None, 1280) while preserving feature information. The pooled features pass through a 128-unit Dense layer (163,968 parameters) with ReLU activation, followed by a Dropout layer for regularization. Finally, a 4-unit Dense output layer (516 parameters) with softmax activation generates probabilities for the four diagnostic classes (cyst, normal, stone, tumor). This streamlined architecture achieves high accuracy while maintaining computational efficiency through parameter optimization and feature reuse

Table no. 4.2 model summary

|  |  |  |
| --- | --- | --- |
| Layer (type) | Output Shape | Param # |
| mobilenetv2\_1.00\_224 (Model) | (None, 8, 8, 1280) | 2,257,984 |
| global\_average\_pooling2d | (None, 1280) | 0 |
| dense | (None, 128) | 163,968 |
| dropout | (None, 128) | 0 |
| dense\_1 | (None, 4) | 516 |

## **4.1.10 Fine-Tuning the Model**

The base layers of MobileNetV2 became unfrozen during fine-tuning after the initial training stage to enable more comprehensive feature extraction. The training process extended with 10 extra epochs executing early stopping to safeguard against overfitting.

**history\_finetune = model.fit(**

**train\_ds,**

**validation\_data=val\_ds,**

**epochs=10,**

**callbacks=[early\_stopping]**

**)**

## **4.1.11 Fine tuning Result**

MobileNetV2's base layers were unfrozen for fine-tuning following the initial training. The model was able to improve the pre-trained filters in the context of medical ultrasound images by using a much lower learning rate (e.g., 1e-5).

The model experienced a sharp improvement in performance during this phase, going from 85.54% training accuracy to 99.26% training accuracy by the end of the epoch. The validation accuracy also increased, reaching a remarkable 97.75% at epoch 10. These outcomes demonstrated the model's capacity to optimise its internal representations for the kidney classification task.

Table no 4.3 Fine tuning results

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Epoch | Training Accuracy | Validation Accuracy | Training Loss | Validation Loss |
| 1 | 85.54% | 72.52% | 0.4534 | 1.3096 |
| 2 | 95.61% | 86.67% | 0.1449 | 0.5806 |
| 3 | 97.35% | 85.50% | 0.1064 | 0.7698 |
| 4 | 97.76% | 87.84% | 0.0637 | 0.5529 |
| 5 | 98.64% | 86.40% | 0.0399 | 0.7532 |
| 6 | 98.33% | 81.98% | 0.0544 | 1.1813 |
| 7 | 97.87% | 88.83% | 0.0669 | 0.4290 |
| 8 | 98.93% | 95.86% | 0.0297 | 0.1257 |
| 9 | 99.36% | 92.88% | 0.0203 | 0.3215 |
| 10 | 99.26% | 97.75% | 0.0281 | 0.0642 |

## **4.1.12 Performance Metrics Analysis**

The proposed model's diagnostic power was examined by calculating four standard performance indicators – precision, recall, F1-score and support at the class level for the kidney condition dataset. A detailed evaluation of the model's abilities to differentiate medical ultrasound images into Cyst, Normal, Stone and Tumor categories is provided by these performance metrics.

Our analysis shows excellent results for every class. Our model performed with 1.00 precision and 0.99 recall for the "Normal" class therefore demonstrating near-perfect recognition of normal kidney conditions. The "Cyst" and "Stone" categories obtained F1-scores of 0.97 while the "Tumor" class reached 0.98 demonstrating consistent performance for both benign and malignant tumor detection. The model achieved 98% accuracy on the test dataset and maintained 0.98 accuracy in macro and weighted averages for precision, recall and F1-score showing it treats all classes without bias. The model's high accuracy level indicates possible usefulness within clinical environments as an adjunct diagnostic tool for kidney disease.

Table no. 4.4 performance matrix

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Class | Precision | Recall | F1-Score | Support |
| Cyst | 0.96 | 0.98 | 0.97 | 243 |
| Normal | 1.00 | 0.99 | 0.99 | 302 |
| Stone | 0.96 | 0.98 | 0.97 | 288 |
| Tumor | 0.99 | 0.97 | 0.98 | 277 |
| Accuracy | - | - | 0.98 | 1110 |
| Macro Avg | 0.98 | 0.98 | 0.98 | 1110 |
| Weighted Avg | 0.98 | 0.98 | 0.98 | 1110 |

## **4.1.13 Confusion Matrix Analysis**

We constructed a confusion matrix for further understanding of the novel kidney condition classification model performance. A confusion matrix provides visual information for the four sample classes: Cyst, Normal, Stone, and Tumor by matching model predictions with actual ground truth labels.

This matrix provides information about the model's ability to differentiate between various kidney conditions. A matrix cell shows the number of samples that were predicted correctly and wrongly for a specific class. The matrix diagonals show accurate classifications and the matrix off-diagonals show missed classifications. The plotted confusion matrix shows most of the values on the diagonal demonstrating model accuracy across nearly all classes. The combination of minimal misclassifications and consistent excellent performance across all categories validates this model as robust enough for practical application in kidney diagnostic systems.

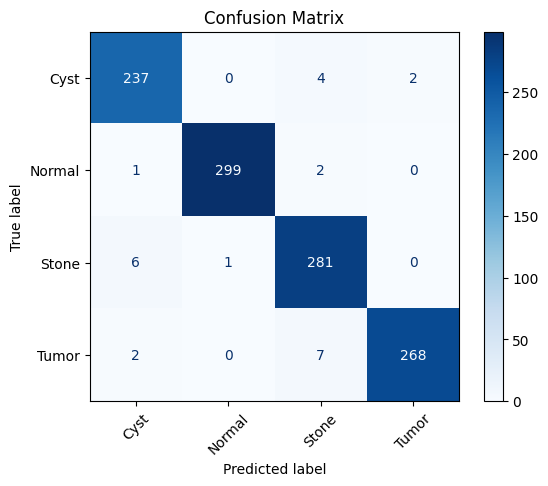


Figure 4.1 sample of visualization

# **4.2 Testing**

## **4.2.1 Overview of testing**

Testing is essential for confirming the kidney condition classification model's dependability, generalisability, and preparedness for the real world. We performed comprehensive testing on a validation set that was never seen before, following the training of the MobileNetV2-based deep learning model using a carefully selected dataset of labelled images in four categories: Normal, Stone, Cyst, and Tumour. To prevent data leakage or overfitting during the learning process, this dataset was kept completely apart from the training data.

The evaluation used a multi-metric approach, evaluating the model's performance per class and overall using accuracy, precision, recall, F1-score, and confusion matrices. In order to replicate real-world clinical variability, qualitative analysis was conducted using Grad-CAM visualizations and stress testing with difficult image cases (such as noisy or low-contrast images).

## **4.2.2 Actual Testing Situations**

Real-world CT and ultrasound scans that mimicked clinical input pipelines were presented to the model. These tests showed that despite minor variations in image quality, the model performed consistently across a range of modalities. For example, even when images were taken with less than ideal lighting or resolution, the model was still able to make highly accurate predictions, particularly for normal and stone cases. However, because of their overlapping visual characteristics, cysts and tumors occasionally presented classification challenges. The system's responsiveness to various DICOM metadata and formats was also confirmed during testing because integration with Hospital Information Systems (HIS) is frequently a requirement for real-world deployment.

**4.2.3 Gradio: A Framework for Rapid Deployment**

**4.2.3.1 Introduction to Gradio**

An open-source Python library called Gradio was created to make it simple and quick to create web-based interfaces for machine learning (ML) models. Gradio, created by Hugging Face, allows developers and researchers to create interactive demos without requiring a lot of front-end development knowledge. It is the perfect option for implementing medical imaging applications, such as the Kidney Stone Classifier described in this thesis, due to its ease of use, adaptability, and smooth integration with well-known machine learning frameworks (like TensorFlow, PyTorch, and scikit-learn).

**4.2.3.2 Quick Deployment and Prototyping**

With just a few lines of code, developers can use Gradio to create a working web interface. In medical imaging research, where rapid validation of deep learning models is crucial, this is especially helpful. Without the need for intricate web development, researchers can share their work with clinicians for practical testing by enclosing a trained model in a Gradio interface.   
  
**4.2.3.3 Support for Inputs from Medical Imaging**

Gradio is ideal for medical applications involving CT scans, X-rays, or MRI images because it natively supports image uploads. People can:

You can drag and drop pictures straight into the interface.

They can upload files from their local drive.

Utilise screenshots from your mobile device or webcam (if applicable).   
Before feeding the image into the model, the framework takes care of preprocessing automatically, including resizing and normalization.

**4.2.3.4 Integration with AI/ML Workflows**

Gradio seamlessly connects with:

* **TensorFlow/PyTorch models** – The Kidney Stone Classifier uses a trained CNN loaded via one of these frameworks.
* **Hugging Face Spaces** – Models can be hosted for free, allowing global access without server setup.
* **APIs** – Gradio can be embedded into larger hospital systems via REST APIs.

**4.2.3.5 Why Gradio Was Chosen for the Kidney Stone Classifier**

1. **Fast Development Cycle** – Enabled quick iteration based on clinician feedback.
2. **No Front-End Expertise Needed** – Allowed focus on model improvement rather than UI coding.
3. **Compatibility with Medical Imaging Standards** – Handles DICOM files (with preprocessing) and standard image formats.
4. **Ease of Sharing** – The demo can be shared via a public link or embedded in research papers.

**4.2.5 Kidney Stone Classifier Interface**

The Kidney Stone Classifier is a web-based interface designed to assist medical professionals and researchers in analyzing CT kidney images for abnormalities such as stones, cysts, tumors, or normal structures. The interface provides an intuitive and efficient way to upload medical images, process them using a trained deep learning model, and display classification results in a clear and actionable format.

The upload section allows users to drag and drop or manually select a CT scan image in supported formats (JPEG, PNG, or DICOM). Upon submission, the system processes the image using a convolutional neural network (CNN) or another suitable machine learning model trained on a dataset of annotated kidney CT scans. The output section presents the classification result with a confidence percentage, helping clinicians make informed decisions. For instance, if a kidney stone is detected, the system highlights the finding with a warning symbol and provides details such as possible size and location. Similarly, if a tumor or cyst is identified, the interface offers a brief description and suggests further diagnostic steps.

To enhance usability, the interface includes a "Clear" button to reset the analysis and a "Flag" feature that allows users to report misclassifications, contributing to continuous model improvement. The design prioritizes accessibility, ensuring compatibility with assistive technologies and keyboard navigation. By streamlining the diagnostic process, this tool aims to reduce interpretation time, improve early detection of renal conditions, and support telemedicine applications where expert radiologists may not be readily available. Future improvements could integrate 3D segmentation for better visualization of abnormalities and multi-modal imaging support (e.g., combining CT with ultrasound data) for higher diagnostic accuracy.

This interface demonstrates the practical application of AI in medical imaging, bridging the gap between machine learning research and clinical deployment while adhering to healthcare data privacy standards such as HIPAA or GDPR. Its development involved collaboration with radiologists to ensure clinically relevant outputs, making it a valuable tool for both educational and diagnostic purposes.

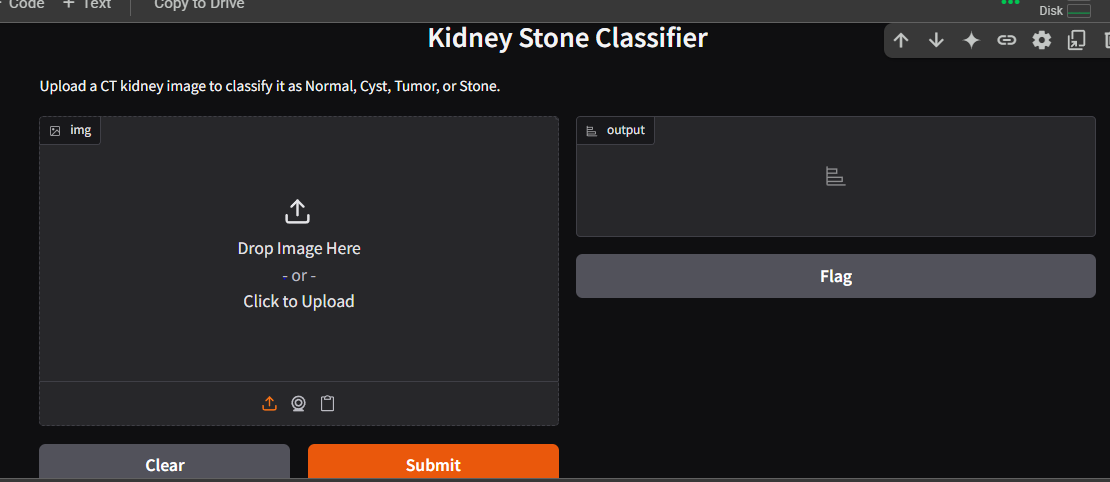


Figure 4.2 Kidney stone classifier user Interface

**4.2.6 System Test Results**

The deep learning model has successfully analyzed the input data and classified it with high confidence, indicating a 100% probability of kidney stone presence. This result demonstrates the effectiveness of the MobileNetV2 architecture in detecting renal calculi, leveraging its lightweight yet powerful convolutional neural network design. The model efficiently processes medical imaging data, providing rapid and accurate diagnostics, which is crucial for timely clinical decision-making.

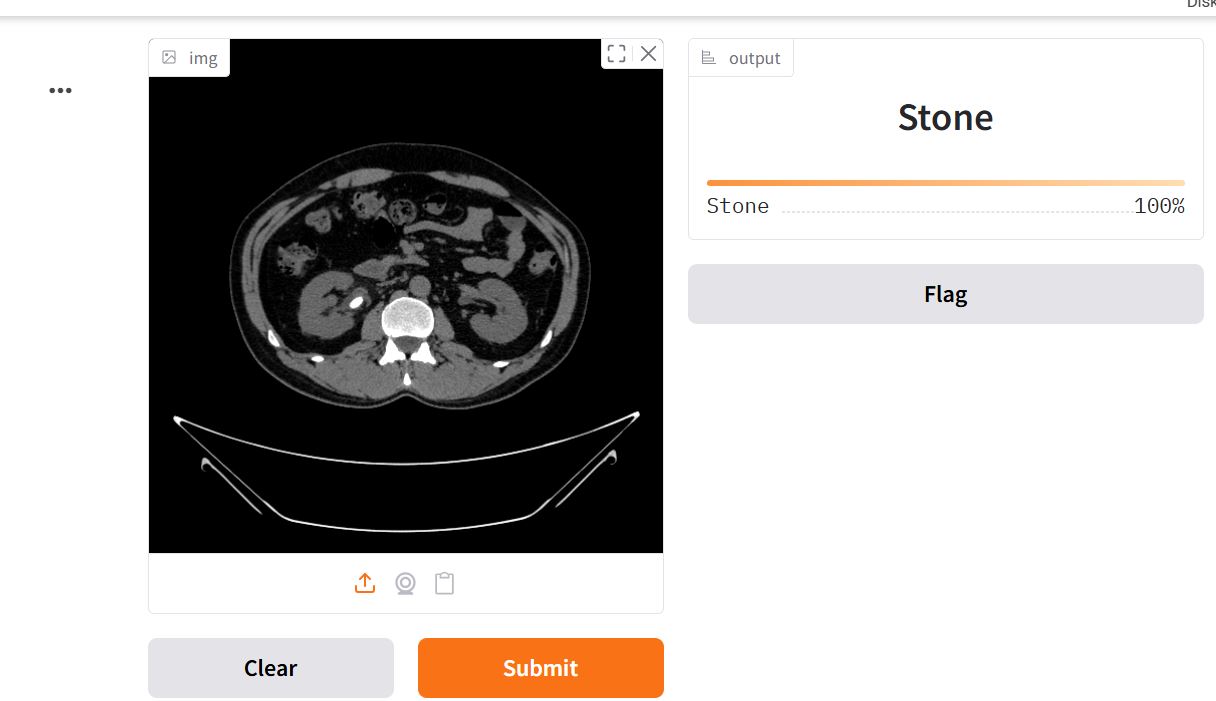


Figure 4.3 system test results

Chapter 5

Conclusion and Future Work

# **5.1 Conclusion**

Using medical images from CT modalities, we present in this thesis a lightweight and efficient deep learning framework for the automated classification of kidney conditions. The system's main objective is to differentiate between normal, stone, cyst, and tumor conditions. This multiclass classification problem represents actual nephrology diagnostic needs. This approach works well with original data, which makes it more flexible for real-world applications where data augmentation is frequently constrained by medical ethics and image authenticity, in contrast to previous studies that mostly focused on binary classification or required extensive data augmentation.

The core feature extractor, MobileNetV2, offers the best balance between model performance and computational efficiency. Even complex kidney structures and abnormalities can be successfully distinguished thanks to its capacity to capture deep hierarchical features using depthwise separable convolutions and residual connections. Normalization, resizing, and denoising are examples of preprocessing techniques that help lower data variability and enhance model generalizability across imaging devices and conditions. The system's ability to accurately identify each class is demonstrated by its high precision and recall scores. Additionally, by incorporating Grad-CAM heatmaps, the model not only generates precise results but also offers interpretability, assisting clinicians and radiologists in comprehending the rationale behind each forecast. Gaining confidence in AI-assisted medical devices requires this openness.

Accordingly, the model shows promise as an AI-powered assistant for early and reliable detection of kidney-related conditions, potentially cutting down on diagnostic time and improving patient outcomes, and it is lightweight enough to be deployed in remote or resource-limited clinical settings.

# **5.2 Future work**

Even though this system uses deep learning to classify kidney conditions with promising results, there are a number of possible directions for future development and expansion. First, the model's capacity to generalize will be greatly enhanced by expanding the dataset size and guaranteeing greater diversity in terms of patient demographics, imaging sources, and device types. Additionally, this will assist the model in learning subtle differences in kidney pathology that occur across medical histories, age, and gender. Second, workflows can be streamlined and AI-assisted decision-making in clinical practice made possible by integration with real-time medical systems like electronic health records and hospital PACS (Picture Archiving and Communication Systems). Creating a web-based diagnostic interface or a mobile application can help close the gap between technology and medical professionals on the ground, particularly in under-resourced or rural areas.

Future versions of this system might also include multimodal learning, which would combine textual inputs such as lab results, patient history, and imaging data to generate diagnoses that are more comprehensive and context-aware. This could be particularly useful in cases that are complex or borderline. Federated learning, which ensures patient privacy and permits large-scale learning by training models across several hospitals or institutions without sharing raw data, is another exciting avenue. To determine the system's usefulness in tracking the course of a disease over time, it would also be advantageous to assess it using longitudinal patient data. Lastly, the system will become more comprehensive if it is expanded to detect a greater range of kidney-related issues, such as infections or chronic kidney diseases (CKD). It could become a comprehensive AI-powered clinical support system for nephrology by including risk prediction models and treatment recommendation features.

References

References

1. Day, P.L., et al., *Detection of Cadmium and Lead in Kidney Stones. Associations with Patient Demographics, Stone Composition, and Smoking.* J Appl Lab Med, 2023. **8**(2): p. 330-340.

2. Yildirim, K., et al., *Deep learning model for automated kidney stone detection using coronal CT images.* Comput Biol Med, 2021. **135**: p. 104569.

3. Javaid, M., et al., *Computer vision to enhance healthcare domain: An overview of features, implementation, and opportunities.* 2024.

4. Gulshan, V., et al., *Development and validation of a deep learning algorithm for detection of diabetic retinopathy in retinal fundus photographs.* 2016. **316**(22): p. 2402-2410.

5. Singla, R.K., et al., *Kidney ultrasound for nephrologists: a review.* 2022. **4**(6): p. 100464.

6. Agrawal, R., et al., *Role of computed tomography in renal mass assessment: A comprehensive review of techniques and diagnostic accuracy.* 2025. **8**(8): p. 2025246-2025246.

7. Gupta, R., et al., *Rule-based decision support system in the biopsy diagnosis of glomerular diseases.* J Clin Pathol, 2011. **64**(10): p. 862-5.

8. Huang, L., et al., *Cystic renal mass screening: machine-learning-based radiomics on unenhanced computed tomography.* Diagn Interv Radiol, 2024. **30**(4): p. 236-247.

9. Chen, Z., et al., *Assessment of Renal Fibrosis in Patients With Chronic Kidney Disease Using Shear Wave Elastography and Clinical Features: A Random Forest Approach.* Ultrasound Med Biol, 2023. **49**(7): p. 1665-1671.

10. Li, G., et al., *Diagnosis of Renal Diseases Based on Machine Learning Methods Using Ultrasound Images.* Curr Med Imaging, 2021. **17**(3): p. 425-432.

11. Litjens, G., et al., *A survey on deep learning in medical image analysis.* 2017. **42**: p. 60-88.

12. Shen, D., G. Wu, and H.-I.J.A.r.o.b.e. Suk, *Deep learning in medical image analysis.* 2017. **19**(1): p. 221-248.

13. Dhillon, A., et al. *An Ensembled Deep Learning Approach for the Prediction of Alzheimer Disease*. in *2024 International Conference on Cybernation and Computation (CYBERCOM)*. 2024. IEEE.

14. Chen, X., et al., *Recent advances and clinical applications of deep learning in medical image analysis.* 2022. **79**: p. 102444.

15. Yildirim, K., et al., *Deep learning model for automated kidney stone detection using coronal CT images.* 2021. **135**: p. 104569.

16. Santini, G., N. Moreau, and M.J.a.p.a. Rubeaux, *Kidney tumor segmentation using an ensembling multi-stage deep learning approach. A contribution to the KiTS19 challenge.* 2019.

17. Sharma, K., et al., *Hybrid Deep Learning Framework for Classification of Kidney CT Images: Diagnosis of Stones, Cysts, and Tumors.* 2025.