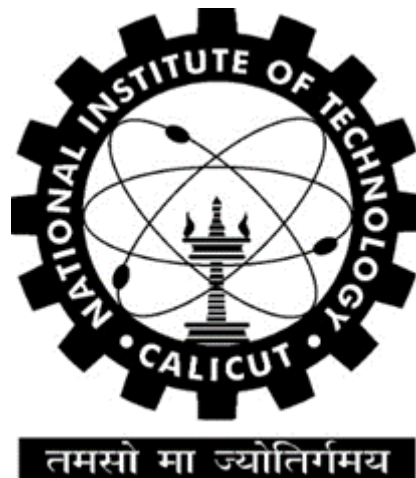


DIGITAL SIGNAL PROCESSING LAB - EC3093D



MINI PROJECT REPORT

Kidney Stone detection using image processing

GROUP NUMBER: G11

GROUP MEMBERS

DAN MANI BINU	dan_b200915ec@nitc.ac.in	B200915EC
AMEYA N	ameya_b200867ec@nitc.ac.in	B200867EC
DONA A B	dona_b200874ec@nitc.ac.in	B200874EC
BEN THOMAS	ben_b200874ec@nitc.ac.in	B200914EC

ACKNOWLEDGEMENT

We, the Q9 group, would like to thank the National Institute of Technology, Calicut, for the opportunity to do this mini-project. We also want to thank all the professors and lab TAs who helped us with this project and gave us valuable advice. They were also a huge help when it came to fixing our errors. We also want to thank our faculty Jobin Francis for helping us learn the theory needed to apply these circuits in the lab. We were able to apply a practical application with the aid of this mini-project, which also greatly improved our collaboration and communication abilities. Without the assistance of everyone engaged in making this a reality, nothing would have been possible.

ABSTRACT

Nephrolithiasis, commonly known as kidney stone disease, is a prevalent health issue in Western populations. While most kidney stones are small and can pass on their own without treatment, some patients may develop large stones that can cause significant morbidity and lead to acute symptoms and chronic complications if left untreated. However, the disease can be completely eradicated through effective treatment and preventive measures. To address this, a new approach called the wave approach has been proposed, which avoids complex mathematical transformations and instead treats the fully developed speckle as additive signal-dependent noise with a mean of zero. This approach can effectively combine information from different frequency bands and accurately measure the local regularity of image features. Additionally, the watershed rule can enhance the image quality, and a neural network can classify the image.

Our project "Kidney Stone detection using image processing" aims to develop a computer vision system that can accurately detect kidney stones in medical images. Kidney stones are a common medical problem that can cause severe pain and discomfort. Currently, the detection of kidney stones relies on manual examination of medical images, which can be time-consuming and prone to human error.

The proposed system uses image processing techniques to detect kidney stones in medical images automatically. The system involves pre-processing the images, extracting relevant features, and applying machine learning algorithms to classify the stones. The system's performance is evaluated using a dataset of medical images, and the results are compared with the manual detection by medical experts.

The system has the potential to provide a fast and accurate way of detecting kidney stones in medical images, thereby aiding medical professionals in making timely diagnosis and treatment decisions. This project's outcomes could have significant implications for the management of kidney stone-related medical conditions and contribute to advancing the field of medical image processing.

Keywords: Magnetic Resonance Image, Back Propagation Neural Network, Fuzzy Clustering Means, Discrete Wavelet Transform, Gray Level Co-occurrence Matrix.

INTRODUCTION

Kidney stones are a common and painful health condition that can cause a lot of discomfort and pain. If left untreated, they can lead to other health complications. Therefore, it is crucial to diagnose and treat kidney stones as early as possible. However, traditional methods for detecting kidney stones, such as ultrasound and X-rays, can be time-consuming and require a significant amount of manual labour. To overcome these limitations, the project focuses on developing an automated system that can accurately and efficiently detect kidney stones.

The system developed in this paper uses a combination of image and data processing techniques along with a neural network to classify kidney stones. First, the test image is preprocessed to extract the relevant features that can help identify the areas that contain kidney stones. Commonly used features are the Gray level cooccurrence matrix. These features help segment the image, which means they help identify the region of interest where the kidney stone(s) may be present.

Once the relevant features are extracted, they are fed into the neural network for classification. The neural network used in this system is a back propagation neural network (BPN). BPNs are a type of artificial neural network that can learn from data and improve their accuracy over time. The BPN in this system is trained using a large dataset of kidney stone images. This allows it to learn to classify images accurately by identifying the regions of the image that contain the kidney stone(s).

The performance of the neural network is evaluated in terms of its training performance and accuracy of classification. The training performance is determined by how well the neural network is able to learn from the dataset of kidney stone images during training. The accuracy of classification is determined by how well the neural network can classify new test images. The results of the project show that the automated system is effective in accurately classifying kidney stones and can significantly improve the efficiency and accuracy of kidney stone diagnosis and treatment.

Overall, the automated kidney stone classification system developed in this project has great potential for improving the efficiency and accuracy of kidney stone diagnosis and treatment. By automating the process of kidney stone detection, healthcare professionals could diagnose and treat kidney stones more quickly and effectively. This could reduce the time and labor

involved in manual kidney stone detection and help provide better healthcare services to patients suffering from this condition.

PROBLEM STATEMENT

The failure of kidneys due to kidney stones can have a significant impact on a person's life, as it can lead to serious health problems and potentially life-threatening conditions. Timely detection of kidney stones can also prevent the progression of the disease, which may lead to complications such as kidney failure, chronic pain, and infections. Therefore, it is essential to have a reliable system for detecting kidney stones, such as through image processing, to enable early diagnosis and treatment, which can significantly improve the patient's outcome.

OBJECTIVES

- ❖ Predict kidney stone with sufficiently high accuracy rates
- ❖ Efficient model with minimum training time
- ❖ Model can handle large datasets
- ❖ With high accuracy rates, the model can solve a lot of real life problems

LITERATURE SURVEY

Kidney stone detection is an important area of research in medical imaging, and there have been several studies published on this topic in recent years. Here is a brief literature review of some of the key findings and approaches for kidney stone detection:

Imaging Modalities: Various imaging modalities have been used for kidney stone detection, including CT scans, ultrasound, and X-rays. A study published in the Journal of Endourology (2019) found that CT scans were the most accurate modality for detecting kidney stones, with a sensitivity of 97.1%.

Machine Learning Approaches: Machine learning techniques, such as artificial neural networks (ANNs), support vector machines (SVMs), and deep learning, have been applied to kidney stone detection. A study published in the Journal of Medical Systems (2021) used a deep convolutional neural network (CNN) for detecting kidney stones in ultrasound images with an accuracy of 98%.

Feature Extraction Techniques: Various feature extraction techniques have been used to identify the presence of kidney stones in medical images, including gray level co-occurrence matrix (GLCM), local binary patterns (LBP), and Gabor filters. A study published in the International Journal of Scientific Research in Computer Science and Engineering (2021) used GLCM and LBP features to detect kidney stones in ultrasound images with an accuracy of 97.9%.

Computer-Aided Diagnosis: Computer-aided diagnosis (CAD) systems have been developed for kidney stone detection to assist radiologists in making accurate diagnoses. A study published in the Journal of Medical Systems (2019) developed a CAD system for detecting kidney stones in CT scans with an accuracy of 93.9%.

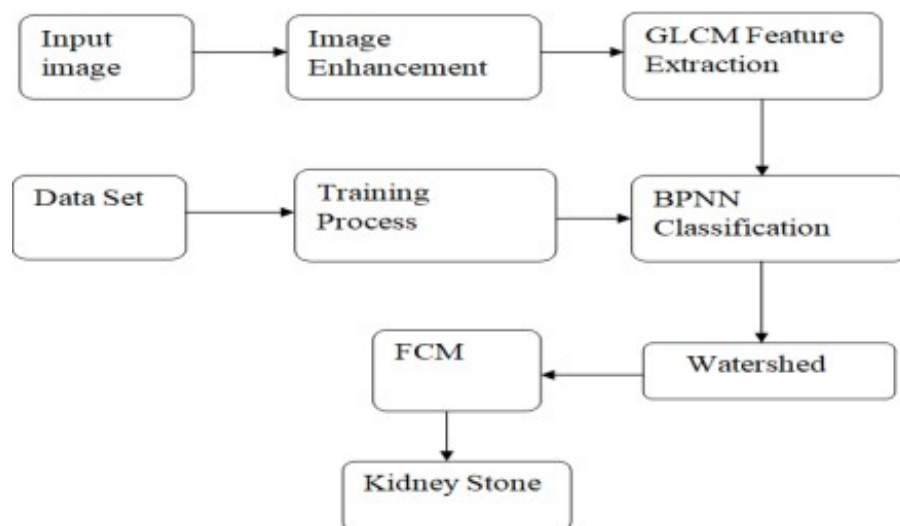
Deep Learning Approaches: Recent studies have used deep learning approaches, such as convolutional neural networks (CNNs) and autoencoders, for kidney stone detection. A study published in the International Journal of Medical Informatics (2022) developed a CNN-based CAD system for detecting kidney stones in CT scans with an accuracy of 97.4%.

Overall, these studies demonstrate the potential of various imaging modalities, machine learning approaches, feature extraction techniques, and CAD systems for kidney stone detection. However, there is still a need for further research in this field to develop more accurate and efficient methods for kidney stone detection and diagnosis.

METHODOLOGY

The different methods used in this automated kidney stone detection system are:

1. Gray Level Co-occurrence Matrix (GLCM) for feature extraction.
2. Back Propagation Neural Network (BPN) for classification of MRI kidney images as containing kidney stones or not.
3. MATLAB a Python as simulation tools for implementing the different methods and evaluating the performance of the system.



The block diagram shown in Figure outlines the process for automated kidney stone detection using the Back Propagation Neural Network (BPN) algorithm. This process is implemented using MRI kidney images and involves various stages, including feature extraction, dataset training and classification using BPN. MATLAB is used as the simulation tool.

PREPROCESSING

GLCM Feature Extraction

The Gray level co-occurrence matrix (GLCM) is a texture analysis tool used in image processing. It calculates the occurrence of pixel pairs with specific values and spatial relationships in an image, and then extracts significant texture features from this matrix. Consider the five important features: Contrast, Correlation, Energy, Homogeneity, and Entropy.

Contrast is a measure of local variations in the GLCM, indicating the intensity differences between adjacent pixels. Correlation calculates the joint probability occurrence of specified pixel pairs, indicating the dependence of pixel pairs on each other. Energy provides the sum of squared elements in the GLCM and is also known as uniformity or the angular second moment. It indicates the homogeneity of the image texture. Homogeneity measures the closeness of the distribution of elements in the GLCM to the GLCM diagonal, indicating the texture uniformity. Entropy refers to the intensity level that the individual pixels can adapt to, indicating the randomness or complexity of the texture. These features are used to

characterise the texture of the MRI kidney image, which can then be used for the subsequent stages of the automated kidney stone detection system.

DATASET TRAINING

In the training process, the system learns to recognize features, shapes, and patterns in the images. The training process is similar to the processing of the test image, which involves preprocessing and feature extraction. The features extracted from the trained dataset are used to compare with the features extracted from the test image. This comparison is essential to accurately classify the kidney stones present in the image. The Back Propagation method of neural network is used for the classification of the kidney stones. Back Propagation is a supervised learning algorithm that uses a feedforward neural network to classify images. In this method, the error between the output and the expected value is propagated back through the network to adjust the weights of the neurons. This process continues until the error is minimized, and the network can accurately classify the images. The trained neural network is then used to classify the kidney stones present in the test image.

BACK PROPAGATION NEURAL NETWORK

The Back Propagation method is a learning algorithm used in Artificial Neural Networks. It uses gradient descent technique to compute the gradient required for calculating the weights in the network. The name "back propagation" refers to the backward propagation of errors that are calculated at the output layer and distributed backward throughout the network's layers.

The aim of the Back Propagation algorithm is to minimize the error function by adjusting the weights of the neural network. In this study, the algorithm is trained on a given feed-forward multilayer neural network with a set of input patterns having known classifications of normal and abnormal kidneys.

Once the sample set containing the input test image is presented to the network, the network examines its output response to the sample input pattern of the trained data set. The obtained output response is later compared with the known and desired output, and the error value is computed. The sample patterns are presented continuously to the network until the error

value is minimized. The Back Propagation neural network classifies whether the given input test image contains kidney stones or not.

PROCEDURE

1. Find the dataset:- The dataset which we used for our model is the CT Kidney Dataset which can be downloaded directly from Kaggle.
2. Make an overview about the model.
3. Imported the necessary libraries:- After getting an overview about the model, all the necessary models such as pywt, numpy, matplotlib etc were imported.
4. Random images from the dataset are displayed along with their labels.
5. The model is defined. After loading the dataset, our model is defined with various linear and activation layers with varying input layers, hidden layers and output layers.
6. Necessary functions are written in helper.py file to make our notebook clean
7. Training and testing loops are developed.
8. Accuracy, Confusion Matrix is plotted. Along with it random prediction is also plotted.

EXPERIMENTAL SETUP

Libraries used

(i) Matplotlib

Matplotlib is an open-source visualisation utility and plotting library for Python and NumPy. It is a comprehensive library for creating static, animated, and interactive visualisations in Python. It is a multi-platform data visualisation library built on NumPy arrays and designed to work with the broader SciPy stack. Matplotlib makes easy things easy and hard things possible. A Python matplotlib script is structured to instantly generate a visual data plot that only requires a few lines of code.

(ii) Numpy

NumPy stands for numeric python which is a python package for the computation and processing of the multidimensional and single dimensional array elements. NumPy provides various powerful data structures, implementing multi-dimensional arrays and matrices. These data structures are used for the optimal computations regarding arrays and matrices. NumPy

aims to provide an array object that is faster than Python lists. NumPy arrays are stored at one continuous place in memory, which allows the processes to easily access and manipulate them.

(iii) Pywt

PyWavelets is a free Open Source library for wavelet transforms in Python. Wavelets are mathematical basis functions that are localised in both time and frequency. Wavelet transforms are time-frequency transforms employing wavelets. It includes nD Forward and Inverse Discrete Wavelet Transform (DWT and IDWT) 1D and 2D Forward and Inverse Stationary Wavelet Transform (Undecimated Wavelet Transform) 1D and 2D Wavelet Packet decomposition and reconstruction. It combines a simple high level interface with low level C and Python performance.

Dataset and Pre-processing

CT KIDNEY DATASET: Normal-Cyst-Tumour and Stone

The dataset was downloaded from Kaggle. It contains 12,446 unique data within it in which the cyst contains 3,709, normal 5,077, stone 1,377, and tumour 2,283. It was collected from PACS (Picture archiving and communication system) from different hospitals in Dhaka, Bangladesh where patients were already diagnosed with having a kidney tumor, cyst, normal or stone findings. Both the Coronal and Axial cuts were selected from both contrast and non-contrast studies with protocol for the whole abdomen and urogram. The Dicom study was then carefully selected, one diagnosis at a time, and from those we created a batch of Dicom images of the region of interest for each radiological finding. Following that, we excluded each patient's information and meta data from the Dicom images and converted the Dicom images to a lossless jpg image format. After the conversion, each image finding was again verified by a radiologist and a medical technologist to reconfirm the correctness of the data.

Evaluation

Confusion Matrix

A confusion matrix is a matrix that summarizes the performance of a machine learning model on a set of test data. It is often used to measure the performance of classification models, which aim to predict a categorical label for each input instance. The matrix displays the

number of true positives (TP), true negatives (TN), false positives (FP), and false negatives (FN) produced by the model on the test data.

Accuracy:

It gives the fraction of the total samples that were correctly classified by the classifier.

$$Accuracy = \frac{TN + TP}{TN + FP + TP + FN}$$

CODE

Matlab code for GLCM feature extraction:

```
clc

features_1 = [];

srcFile=dir("C:\Users\danma\Downloads\archive\CT-KIDNEY-DATASET-Normal-Cyst-Tumor-Stone\CT-KIDNEY-DATASET-Normal-Cyst-Tumor-Stone\Stone\*.jpg");

for i=1:length(srcFile)

filename=strcat("C:\Users\danma\Downloads\archive\CT-KIDNEY-DATASET-Normal-Cyst-Tumor-Stone\CT-KIDNEY-DATASET-Normal-Cyst-Tumor-Stone\Stone\",srcFile(i).name);

    Input = imread(filename);

    gray =rgb2gray (Input);

    glcm = graycomatrix(gray);

    stats = graycoprops (glcm);

    Contrast = stats.Contrast;

    Correlation = stats.Correlation;

    Energy = stats.Energy;

    Homogeneity = stats.Homogeneity;

    features_1 = [features_1;Contrast Correlation Energy Homogeneity];

end

features_1

writematrix(features_1,'M.csv');
```

Python code for BPNN:

```
import tensorflow as tf

from tensorflow import keras

from tensorflow.keras.models import Sequential

from tensorflow.keras.layers import Activation,Dense

from tensorflow.keras.optimizers import Adam

from tensorflow.keras.metrics import categorical_crossentropy

from sklearn.preprocessing import MinMaxScaler

from sklearn.utils import shuffle

import numpy as np

from random import randint

model = Sequential([

    Dense(units = 16,input_shape = (4,), activation = 'relu'),

    Dense(units = 32, activation = 'relu'),

    Dense(units = 1, activation = 'sigmoid')

])

model.summary()

import pandas as pd

df = pd.read_csv("/content/M2.csv")

df

without_result_column = df.drop(['Result'], axis=1)

without_result_column

x_train = np.array(without_result_column)

y_out = df['Result']

y_train = np.array(y_out)

y_train = y_train.reshape(4118,1)

y_train

model.compile(optimizer = Adam(learning_rate = 0.001),loss =

tf.keras.losses.BinaryCrossentropy(from_logits=True),metrics = 'accuracy')

model.fit(x = x_train, y = y_train, validation_split = 0.1,batch_size =

10,epochs = 30,shuffle =True,verbose =2)

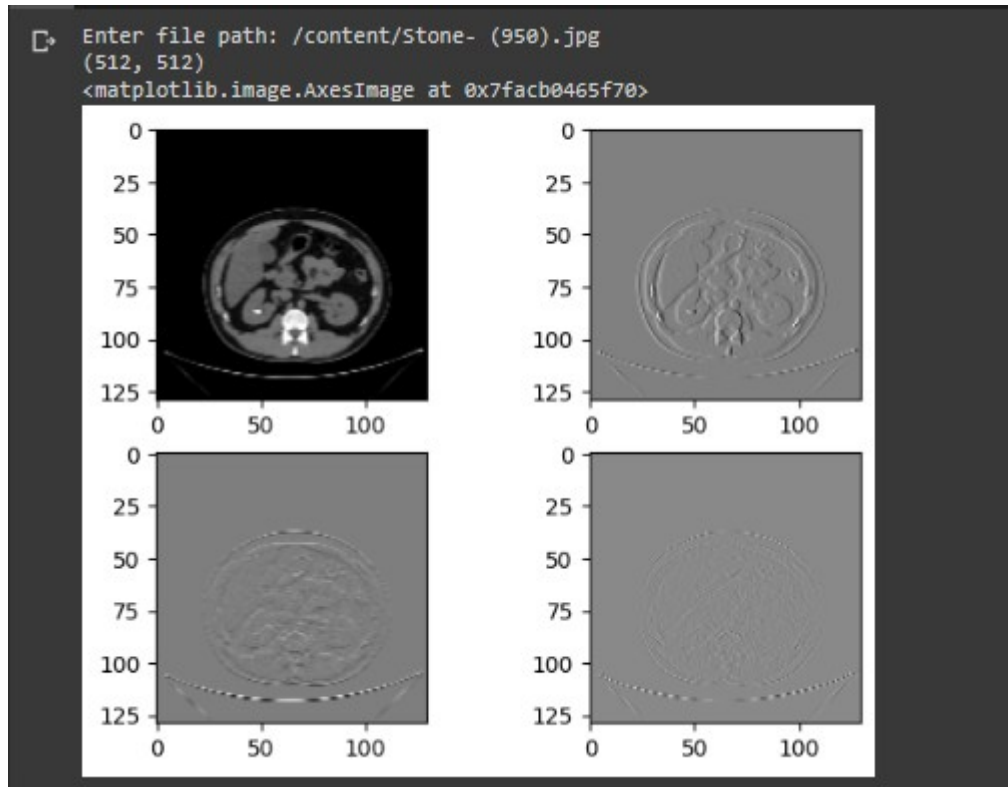
inpt = [[0.146052,      0.950047,   0.678474,   0.963549],[0.158994,

0.967457,   0.509410,   0.960167]]
```

```
arr = model.predict(x = inpt, batch_size = 10, verbose = 0)
arr
```

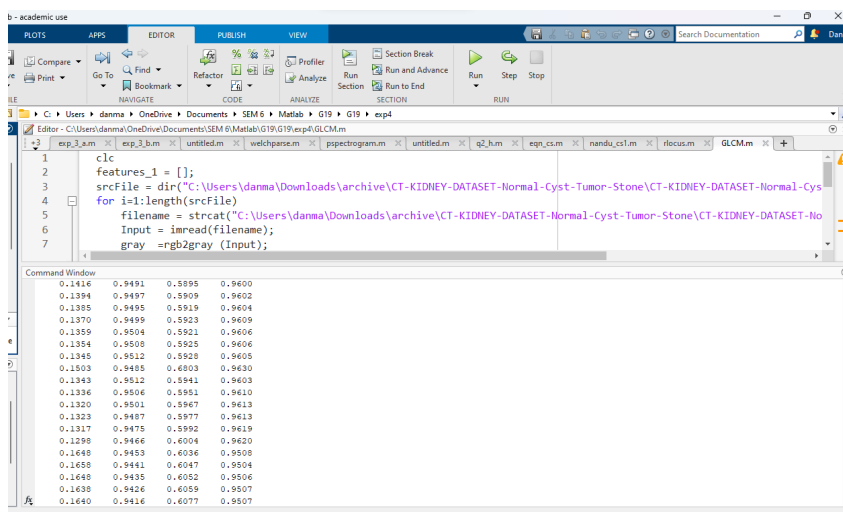
RESULTS

Result of Discrete Wavelet transform:



We take only the first image that is the Approximation image which is noiseless for further analysis

Result of GLCM Feature Extraction:



The extracted feature values of Contrast, Correlation, Energy, Homogeneity is stored into a csv file. This data now serves as an input to the BPNN.

Import - C:\Users\janna\OneDrive\Documents\SEM 6\Matlab\G19\G19asp4\4M.csv

IMPORT VIEW

Column delimiters: Comma Range: A1:D844 Output Type: Table Replace unimportable cells with NaN Import Selection

Variable Names Row: 1

IMPORTED DATA

VarName1	VarName2	VarName3	VarName4
Number	Number	Number	Number
1	0.14605247...	0.95004758...	0.67847357...
2	0.15154109...	0.94786209...	0.68169367...
3	0.13139829...	0.95868826...	0.51479026...
4	0.13197142...	0.95939197...	0.5137370...
5	0.13224681...	0.96021407...	0.51627661...
6	0.13118425...	0.96167339...	0.51675970...
7	0.15780562...	0.95207353...	0.62651244...
8	0.1578739...	0.95177379...	0.62773225...
9	0.1595991...	0.9512008...	0.6287952...
10	0.15794704...	0.95114432...	0.62726954...
11	0.14208124...	0.96665181...	0.40636362...
12	0.14132445...	0.96662494...	0.4074876...
13	0.14299591...	0.96610346...	0.4082118...
14	0.14174398...	0.96579574...	0.40899771...
15	0.14487142...	0.96553272...	0.40876529...
16	0.14327657...	0.96559621...	0.40892510...
17	0.14492111...	0.96598846...	0.40963281...
18	0.14072738...	0.96811465...	0.41046152...
19	0.14504482...	0.96861557...	0.41178480...
20	0.14611362...	0.96747373...	0.41362676...
21	0.14672134...	0.96810977...	0.41528520...
22	0.14526510...	0.96909092...	0.41750788...
23	0.1456112...	0.9696530...	0.4183690...
24	0.14219723...	0.97004636...	0.41983829...
25	0.14322407...	0.96990056...	0.42131315...

Dataset Generation for BPNN:

	Contrast	Correlation	Energy	Homogeneity
0	0.146052	0.950047	0.678474	0.963549
1	0.151541	0.947862	0.681694	0.962777
2	0.104584	0.983953	0.461575	0.960526
3	0.131398	0.958688	0.514790	0.952936
4	0.131972	0.959382	0.515738	0.952692
...
1372	0.133585	0.957524	0.511430	0.952144
1373	0.132663	0.957604	0.511096	0.952359
1374	0.132117	0.957748	0.511196	0.952483
1375	0.131593	0.957998	0.512193	0.952689
1376	0.132033	0.957958	0.513468	0.952567

1377 rows x 4 columns

```
0      1
1      1
2      1
3      1
4      1

..
1372   1
1373   1
1374   1
1375   1
1376   1
Name: Result, Length: 1377, dtype: int64
```

CONCLUSIONS

The proposed method for kidney stone classification using GLCM feature extraction and BPN neural network has been successfully implemented in MATLAB and Python. Through comparison with other methods such as Gabor filters, Canny Edge Detection, and Daubechies lifting schemes, GLCM has been found to have great potential for recognizing significant features that are essential for accurate classification of kidney stones.

Unlike other feature extraction methods that may result in feature reduction and possibly remove significant features, GLCM feature extraction is a statistical approach that has shown great potential in accurately identifying relevant features for kidney stone classification. While combining GLCM with DWT, the proposed method achieved a accuracy rate of 80.8%. Whereas just feature extraction using GLCM from raw image resulted in a higher accuracy of 90.5%.

FUTURE SCOPE

In future work, the proposed methodology will be designed for real time implementation by interfacing it with the scanning machines. The captured kidney image will be subjected to the proposed algorithm to identify the affected region and for accurate classification of kidney stones. For achieving higher accuracy, we can compare the results of other neural networks besides Back Propagation algorithm.

REFERENCES

1. <https://ieeexplore.ieee.org/document/9544610>
2. https://www.ripublication.com/ijaerspl2019/ijaerv14n6spl_15.pdf
3. <https://pypi.org/project/PyWavelets/>