

DETECTION OF KIDNEY STONES USING NEURAL NETWORK

Submitted in partial fulfillment of the requirements for the award
of Bachelor of Technology degree in Biomedical Engineering

By

CATHERINE MELITA PRADHIKSHA .P (39240011)

MONICA .S (39240026)

TAMIZHLARRASI .R (39240040)



DEPARTMENT OF BIOMEDICAL ENGINEERING

SCHOOL OF BIO AND CHEMICAL

SATHYABAMA

**INSTITUTE OF SCIENCE AND TECHNOLOGY
(DEEMED TO BE UNIVERSITY)**

Accredited with Grade "A" by NAAC

JEPPIAAR NAGAR, RAJIV GANDHI SALAI, CHENNAI - 600 119

APRIL - 2022



SATHYABAMA

INSTITUTE OF SCIENCE AND TECHNOLOGY

(DEEMED TO BE UNIVERSITY)

Accredited "A" Grade by NAAC | 12B Status by UGC | Approved by AICTE

www.sathyabama.ac.in

DEPARTMENT OF BIOMEDICAL ENGINEERING

BONAFIDE CERTIFICATE

This is to certify that this Inter disciplinary project Report is the bonafide work of Catherine Melita Pradhiksha.P (39240011) ,Monica.S (39240026),

Tamizhlarrasi .R (39240040) who carried out the inter disciplinary project entitled "Detection Of Kidney Stones Using Neural Network" under supervision from January 2022 to April 2022

Internal Guide
DR.GRACE KANMANI

Head of the Department
Dr.T.SUDHAKAR M.Sc.Ph.D

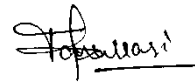
Submitted for Viva voce Examination held on 30.04.2022

Internal Examiner

External Examiner

DECLARATION

We Catherine Melita Pradhiksha .P , Monica .S ,Tamizhlarrasi .R hereby declare that the Inter disciplinary project Report entitled “Kidney Stone Detection Using Neural Network done by us under the guidance of Dr. Grace Kanmani (Internal) is submitted in partial fulfillment of the requirements for the award of Bachelor of Technology degree in Biomedical Engineering



DATE:30.04.2022

SIGNATURE OF THE CANDIDATES

PLACE: CHENNAI

ACKNOWLEDGEMENT

We are pleased to acknowledge our sincere thanks to Board of Management of SATHYABAMA for their kind encouragement in doing this Interdisciplinary Project and for completing it successfully. We are grateful to them.

We convey our thanks to School of Bio and Chemical Engineering and Dr.T.Sudhakar, Head of the Department, Dept. of Biomedical Engineering for providing us necessary support and details at the right time during the progressive reviews.

We would like to express our sincere and deep sense of gratitude to our Interdisciplinary Project Guide Dr. Grace Kanmani for her valuable guidance, suggestions and constant encouragement paved way for the successful completion of our Interdisciplinary Project work.

We wish to express our thanks to all Teaching and Non-teaching staff members of the Department of Biomedical Engineering who were helpful in many ways for the completion of the Interdisciplinary Project.

ABSTRACT:

Back Propagation Network with image and data processing techniques was employed to implement an automated kidney stone classification. The conventional method for medical resonance kidney images classification and stone detection is by human inspection. Operator-assisted classification methods are impractical for large amounts of data and are also non-reproducible. Medical Resonance images contain a noise caused by operator performance which can lead to serious inaccuracies classification. Artificial intelligent techniques through neural networks and feature extraction are shown great potential in this field. Hence, in this paper the Back Propagation Network was applied for the purposes. Decision making was performed in two stages: feature extraction using the principal component analysis and the classification using Back Propagation Network (BPN). This inter disciplinary project presents a segmentation method, Fuzzy C-Mean (FCM) clustering algorithm, for segmenting computed tomography images to detect the lung cancer in its early stages. The performance of the BPN classifier was evaluated in terms of training performance and classification accuracies. Back Propagation Network gives fast and accurate classification than other neural networks and it is a promising tool for classification of the Cancers.

CONTENTS

CHAPTER No.	TITLE	PAGE No.
	ABSTRACT	i
	LIST OF FIGURES	iii
	LIST OF ABBREVIATION	iv
01	INTRODUCTION	1
02	LITERATURE SURVEY	3
03	AIM AND SCOPE	5
04	EXPERIMENTAL METHODS & ALGORITHMS	6
	4.1.DOMAIN OVERVIEW	6
	<i>4.1.1.PYTHON</i>	6
	4.2.DEVELOPMENTAL ENVIRONMENT	7
	<i>4.2.1.BPNN</i>	7
	<i>4.2.2.TYPES</i>	8
	4.3.SYSTEM ANALYSIS	9
	4.4.SOFTWARE REQUIRED	9
	4.5.ALGORITHM	9
	<i>4.5.1.NEURAL NETWORK</i>	9
	<i>4.5.2.ARCHITECTURE DIAGRAM</i>	10
	<i>4.5.3.LAYERS</i>	11
	<i>4.5.4.WORKING</i>	11
05	RESULT,DISCUSSION AND PERFORMANCE	14
	5.1.IMPLEMENTATION	14
	<i>5.1.1.LIST OF MODULES</i>	14
	5.2.PROPOSED METTHOD	14
	5.3.APPLICATION	16
06	SUMMARY AND CONCLUSION	20
	REFERENCE	21

LIST OF FIGURES

FIGURE No.	FIGURE NAME	PAGE No.
4.1	ARCHITECTURE DIAGRAM	10
4.2	REPRESENTATION OF TRAINING SET	11
4.3	RADIAL BASIS FUNCTION	13
5.1	TYPES OF LAYERS USED	14
5.2	OUTPUT IMAGE 1	16
5.3	OUTPUT IMAGE 2	16
5.4	OUTPUT IMAGE 3	17
5.5	OUTPUT IMAGE 4	17
5.6	OUTPUT IMAGE 5	18
5.7	OUTPUT IMAGE 6	18
5.8	OUTPUT IMAGE 7	19

LIST OF ABBREVIATION

GLCM	- GRAY LEVEL CO – OCCURANCE MATRIX
DWT	- DISCRETE WAVELET TRANSFORM
NN	- NEURAL NETWORK
OCT	- OPTICAL COHERENCE TOMOGRAPHY
OMAG	- OPTICAL MICRO ANGIOGRAPHY
BPNN	- BACK PROPOGATION NEYRAL NETWORK
CNN	- CONVOLUTIONAL NEURAL NETWORK
GUI	- GRAPHICAL USER INTERFACE
MIME	- MULTIPURPOSE INTERNET MAIL EXTENTIONS
HTTP	- HYPERTEXT TRANSFER PROTOCOL
PYPL	- PYTHON PACKAGE INDEX
REPL	- READ EVAL PRINT LOOP
IDLE	- INTEGRATED DEVELOPMENT &LEARNING ENVIRONMENT
IDE	- INTEGRATED DEVELOPMENT ENVIRONMENT
K-NN	- K-NEAREST NEIGHBOUR
RSF	- RADIAL BASIS FUNCTION
GRNN	- GENERAL REGRESSION NEURAL NETWORK
DTREG	- DEPENDABLE THEATRICAL RELIGIOUS EXITABLE GODLY

CHAPTER- 01

INTRODUCTION:

Optical coherence tomography (OCT) is an established modality for non-invasive assessment of diseases. As a result, OCT image analysis is becoming increasingly important. Texture analysis has been actively investigated for tissue characterization. The potential of texture analysis has been demonstrated in numerous biomedical applications, including OCT imaging of skin, bladder, eye, atherosclerotic plaque, esophagus, and breast. In general, texture analysis techniques can be classified into three groups: statistical technologies, spectral technologies, and structural technology. The choice of optimal method for texture analysis may vary depending on the specific biomedical applications.

OCT and its functional extension (Doppler OCT and optical micro angiography, OMAG) have been used for imaging kidney microanatomy and microcirculation. OCT can resolve renal corpuscles and urineferous tubules. The morphological changes in these structures are associated with ischemia-reperfusion injury. Automatic algorithm for image analysis algorithm has been developed previously for quantifying spatially-resolved tubular diameter as a potential biomarker for indicating viability of the transplant kidney. In these previous studies, the tubular lumens were segmented out from cross-sectional OCT images of kidney microstructure based on an empirically-determined intensity threshold. However, as OCT intensity is attenuated with depth, segmentation of deeper tubular lumens based on one fixed threshold becomes challenging.

Ultrasound imaging is also one of the available imaging techniques used for diagnosis of kidney abnormalities, which may be like change in shape and position and swelling of limb; there are also other Kidney abnormalities such as formation of stones, cysts, blockage of urine, congenital anomalies, and cancerous cells. During surgical processes it is vital to recognize the true and precise location of kidney stone. The detection of kidney stones using ultrasound imaging is a highly challenging task as they are of low contrast and contain speckle noise. This challenge is overcome by

employing suitable image processing techniques. The ultrasound image is first preprocessed to get rid of speckle noise using the image restoration process. The restored image is smoothened using Gabor filter and the subsequent image is enhanced by histogram equalization. The preprocessed image is achieved with level set segmentation to detect the stone region. Segmentation process is employed twice for getting better results; first to segment kidney portion and then to segment the stone portion, respectively. In this work, the level set segmentation uses two terms, namely, momentum and resilient propagation to detect the stone portion. After segmentation, the extracted region of the kidney stone is given to Symlets, Biorthogonal (bio3.7, bio3.9, and bio4.4), and Daubechies lifting scheme wavelet subbands to extract energy levels. These energy levels provide evidence about presence of stone, by comparing them with that of the normal energy levels. They are trained by CNN to classify and its type of stone with a higher accuracy.

CHAPTER – 02

LITERATURE SURVEY

S.No	Title	Author Name	Year of Publish	Technology / Algorithm used	Drawbacks
1	Characterizing volumes of kidney segments in Streptozotocin induced diabetic rat model utilizing 4D contrast-enhanced ultrasound	Kennita A	2016	Machine Learning/ Contrast-enhanced ultrasound (CEUS)	Low resolution
2	High-Framerate Dynamic Contrast-Enhanced Ultrasound Imaging of Rat Kidney Perfusion	Emily Chang	2018	Image processing/ Filtering	Poor detection
3	Analysis and Identification of Renal Calculi In Computed Tomography Images	Tanmay Shah	2019	Image processing/ Segmentation	Time taking process
4	An image preprocessing method for kidney stone segmentation in CT scan images	Nilar Thein	2018	Image processing/ pre-processing algorithm	time taken, cost
5	RENAL STONE DETECTION AND ANALYSIS BY CONTOUR BASED ALGORITHM	Shahina M K	2019	Image processing/ Contour Based Algorithm	low clarity
6	Kidney Stone Detection in computed Tomography images	T.Akkasaligar	2017	Image processing/ Region of Growing Algorithm	Low Complexity

7	Characterizing the Acoustic Output of an Ultrasonic Propulsion Device for Urinary Stones	Bryan W. Cunitz, Barbrina Dunmire, and Michael R. Bailey	2017	Ultrasonic Propulsion Device	The individual pulses within the burst were dithered ± 2 mm laterally to reduce the local peak pressure.
8	Microwave Dielectric Spectroscopy of Renal Calculi: A Large Scale Study on Dielectric Properties from 500 MHz to 18 GHz	Tuba Yilmaz, Banu Saçlı, Gökhan Cansız, Sulayman Joof, Cemanur Aydınalp, Mehmet Çayören and İbrahim Akduman	2019	dielectric spectroscopy	Spectroscopy frequency are hazardous
9	Review on Lithotripsy and Cavitation in Urinary Stone Therapy	Morteza Ghorbani, Ozlem Oral, Sinan Ekici, Devrim Gozuacik and Ali Kosar	2016	Ultrasound	Low Quality
10	Virtual and Augmented Reality Systems for Renal Interventions: A Systematic Review	Felicitas J. Detmer, Julian Hettig, Daniel Schindele, Martin Schostak, and Christian Hansen	2017	VR & AR	Time Taking Process
11	Automatic detection of calcium phosphate deposit plugs at the terminal ends of kidney tubules	Joncamp	2017	Image processing/ Deep Learning	time-consuming process
12	An Automated Approach for Kidney Segmentation in Three-Dimensional Ultrasound Images	Mahdi Marsousi	2016	Image processing/ Image Colour Analysis	poor sensitivity

CHAPTER – 03

AIM :

The aim of the inter disciplinary project is to detect kidney stone from the person who is affected in it. .In this we used back propagation neural network technique to identify the stone. We are using image processing to extract the glcm and dwt . BPNN is used to train the image. In this we are divide into two segmentation part normal /abnormal if person kidney is abnormal it will pop out where it is affected in kidney .

SCOPE:

The main scope of the inter disciplinary project is to predict the kidney stone from the person who is affected and with help of convolutional neural network technique. In this we used discrete wavelet transform to identify the kidney stone.

CHAPTER – 04

EXPERIMENTAL METHODS AND ALGORITHMS

4.1 DOMAIN OVERVIEW

4.1.1 PYTHON:

Python is an interpreted high-level programming language for programming. Python offers multiple options for developing GUI (Graphical User Interface). Out of all the GUI methods, tkinter is the most commonly used method. It is a standard Python interface to the Tk GUI toolkit shipped with Python. Python with tkinter outputs the fastest and easiest way to create the GUI applications. Creating a GUI using tkinter is an easy task.

PYTHON FEATURES:

Python features a dynamic type system and automatic memory management. It supports multiple programming paradigms, including object-oriented, imperative, functional and procedural, and has a large and comprehensive library. Python is a multi-paradigm programming language. Object-oriented programming and structured programming are fully supported, and many of its features support functional programming and aspect-oriented programming (including by meta programming and meta objects (magic methods)). Many other paradigms are supported via extensions, including design by contract and logic programming.

PYTHON LIBRARIES:

Python's large standard library, commonly cited as one of its greatest strengths, provides tools suited to many tasks. For Internet-facing applications, many standard formats and protocols such as MIME and HTTP are supported. It includes modules for creating graphical user interfaces, connecting to relational databases, generating pseudorandom numbers, arithmetic with arbitrary precision decimals, manipulating regular expressions, and unit testing.

As of March 2018, the Python Package Index (PyPI), the official repository for third-party Python software, contains over 130,000 packages with a wide range of functionality, including:

- Graphical user interfaces
- Web frameworks
- Multimedia
- Databases
- Networking
- Test frameworks
- Automation
- Web scraping
- Documentation
- System administration
- Scientific computing
- Text processing
- Image processing

4.2 DEVELOPMENT ENVIRONMENT

Most Python implementations (including CPython) include a read–eval–print loop (REPL), permitting them to function as a command line interpreter for which the user enters statements sequentially and receives results immediately. Other shells, including IDLE and IPython, add further abilities such as auto-completion, session state retention and syntax highlighting.

As well as standard desktop integrated development environments, there are Web browser-based IDEs; Sage Math (intended for developing science and math-related Python programs); Python Anywhere, a browser-based IDE and hosting environment; and Canopy IDE, a commercial Python IDE emphasizing scientific computing.

BPNN:

Back-propagation is the essence of neural net training. It is the method of fine-tuning the weights of a neural net based on the error rate obtained in the previous epoch (i.e., iteration). Proper tuning of the weights allows you to reduce error rates and to make the model reliable by increasing its generalization . Back propagation is a short form for "backward propagation of errors." It is a standard method of training artificial neural networks. This method helps to calculate the gradient of a loss function with respects to all the weights in the network. Simplifies the network structure by elements weighted links that have the least effect on the trained network. You need to study a group of input and activation values to develop the relationship between the input and hidden unit layers. It helps to assess the impact that a given input variable has on a network output. The knowledge gained from this analysis should be represented in rules. Back propagation is especially useful for deep neural networks working on error-prone inter disciplinary projects, such as image or speech recognition .

Back propagation takes advantage of the chain and power rules allows back propagation to function with any number of outputs.

Neural network is a calculation model inspired by biological nervous system. The functionality of neural network is determined by its network structure and connection weights between neurons. Back propagation neural network (BPNN) is one of the very popular types for its training method called back propagation.

4.2.2 TYPES

- Static Back-propagation
- Recurrent Back propagation

4.3 SYSTEM ANALYSIS

EXISTING SYSTEM:

- Threshold Segmentation
- Cosine Transformation
- Svm
- K Means Clustering

4.4 SOFTWARE REQUIRED

Python Anaconda Navigator

4.5 ALGORITHMS

4.5.1 NEURAL NETWORK

Neural networks are predictive models loosely based on the action of biological neurons.

The selection of the name “neural network” was one of the great PR successes of the Twentieth Century. It certainly sounds more exciting than a technical description such as “A network of weighted, additive values with nonlinear transfer functions”. However, despite the name, neural networks are far from “thinking machines” or “artificial brains”. A typical artificial neural network might have a hundred neurons. In comparison, the human nervous system is believed to have about 3×10^{10} neurons. We are still light years from “Data”.

The original “Perceptron” model was developed by Frank Rosenblatt in 1958. Rosenblatt’s model consisted of three layers, (1) a “retina” that distributed inputs to the second layer, (2) “association units” that combine the inputs with weights and trigger a threshold step function which feeds to the output layer, (3) the output layer which combines the values. Unfortunately, the use of a step function in the neurons made the perceptions difficult or impossible to train. A critical analysis of perceptrons

published in 1969 by Marvin Minsky and Seymour Paper pointed out a number of critical weaknesses of perceptrons, and, for a period of time, interest in perceptrons waned.

Interest in neural networks was revived in 1986 when David Rumelhart, Geoffrey Hinton and Ronald Williams published “Learning Internal Representations by Error Propagation”. They proposed a multilayer neural network with nonlinear but differentiable transfer functions that avoided the pitfalls of the original perceptron’s step functions. They also provided a reasonably effective training algorithm for neural networks.

TYPES:

- Artificial Neural Network
- Probabilistic Neural Networks
- General Regression Neural Networks

4.5.2 ARCHITECTURE DIAGRAM

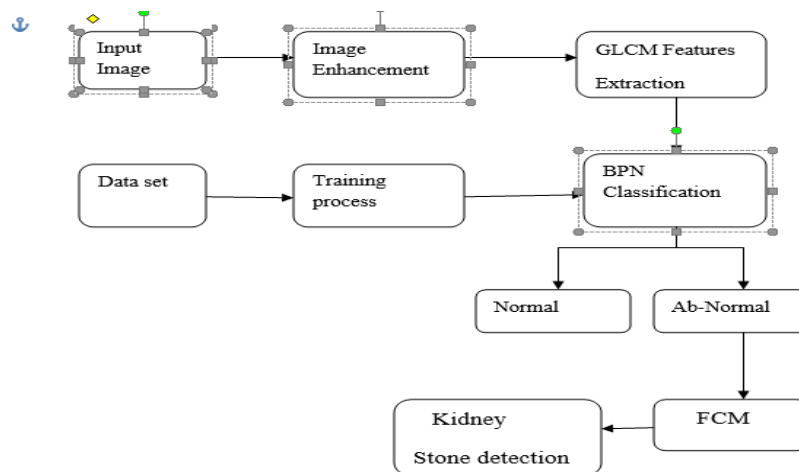


Fig.No.4.1 Architecture Diagram

4.5.3. LAYERS

- Hidden layer
- Input layer
- Pattern layer / Summation layer
- Decision layer

4.5.4. WORKING

Although the implementation is very different, neural networks are conceptually similar to *K-Nearest Neighbor* (k-NN) models. The basic idea is that a predicted target value of an item is likely to be about the same as other items that have close values of the predictor variables. Consider this figure:

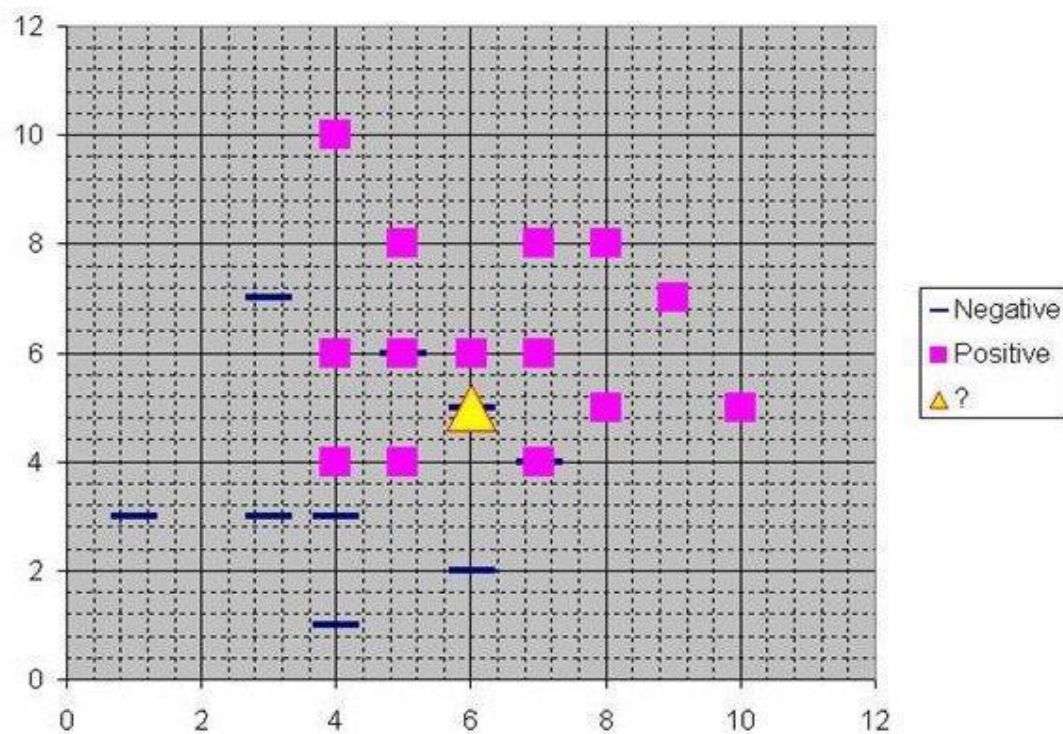


Fig. No.4.2 Representation of Training set

Assume that each case in the training set has two predictor variables, x and y . The

cases are plotted using their x,y coordinates as shown in the figure. Also assume that the target variable has two categories, *positive* which is denoted by a square and *negative* which is denoted by a dash. Now, suppose we are trying to predict the value of a new case represented by the triangle with predictor values $x=6, y=5.1$. Should we predict the target as positive or negative?

Notice that the triangle is position almost exactly on top of a dash representing a negative value. But that dash is in a fairly unusual position compared to the other dashes which are clustered below the squares and left of center. So it could be that the underlying negative value is an odd case.

The nearest neighbor classification performed for this example depends on how many neighboring points are considered. If 1-NN is used and only the closest point is considered, then clearly the new point should be classified as negative since it is on top of a known negative point. On the other hand, if 9-NN classification is used and the closest 9 points are considered, then the effect of the surrounding 8 positive points may overbalance the close negative point.

A neural network builds on this foundation and generalizes it to consider all of the other points. The distance is computed from the point being evaluated to each of the other points, and a *radial basis function* (RBF) (also called a *kernel function*) is applied to the distance to compute the weight (influence) for each point. The radial basis function is so named because the radius distance is the argument to the function.

Weight = RBF (*distance*)

The further some other point is from the new point, the less influence it has.

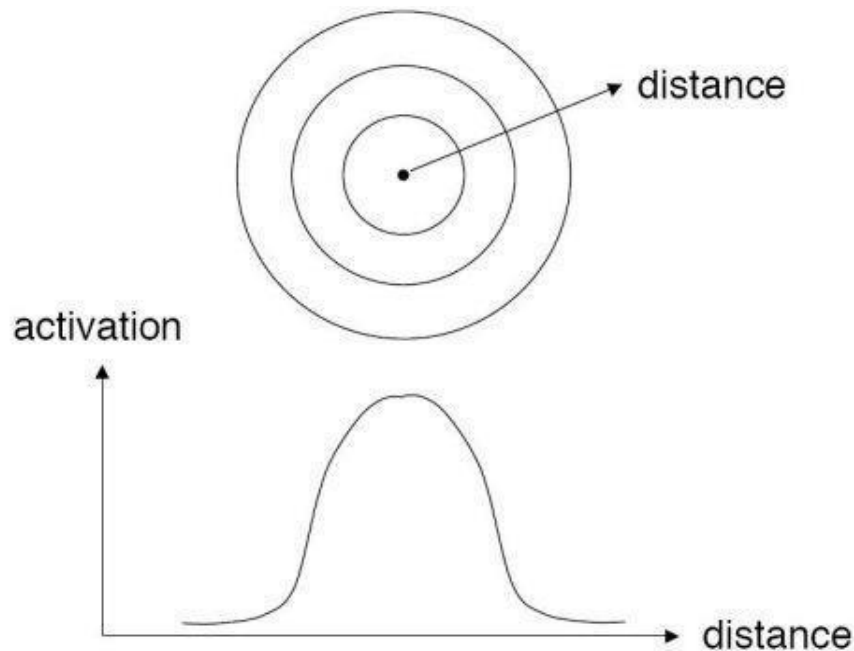


Fig.No.4.3 Radial Basis Function

ADVANTAGES & DISADVANTAGES OF NN

- It is usually much faster to train a NN/GRNN network than a multilayer perceptron network.
- NN/GRNN networks often are more accurate than multilayer perceptron networks.
- NN/GRNN networks are relatively insensitive to outliers (wild points).
- NN networks generate accurate predicted target probability scores.
- NN networks approach Bayes optimal classification.
- NN/GRNN networks are slower than multilayer perceptron networks at
- classifying new cases.
- NN/GRNN networks require more memory space to store the model.

CHAPTER- 5

PERFORMANCE, DISCUSSION AND RESULT

5.1.IMPLEMENTATION

5.1.1.LIST OF MODULES

- Preprocessing
- Dwt
- Glcm feature extraction
- Back propagation neural network
- Fuzzy clustering methods

5.2. PROPOSED METHOD

Neural Network (NN) and General Regression Neural Networks (GRNN) have similar architectures, but there is a fundamental difference: networks perform classification where the target variable is categorical, whereas general regression neural networks perform regression where the target variable is continuous. If you select a NN/GRNN network, DTREG will automatically select the correct type of network based on the type of target variable.

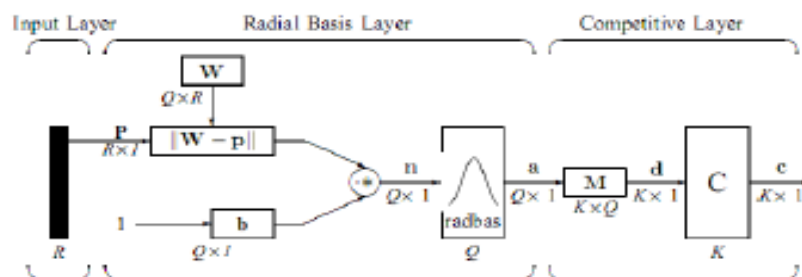


Fig.No.5.1 Types of Layers Used

1) Input Layer:

The input vector, denoted as p , is presented as the black vertical bar. Its dimension is $R \times 1$. In this paper, $R = 3$.

2) Radial Basis Layer:

In Radial Basis Layer, the vector distances between input vector p and the weight vector made of each row of weight matrix W are calculated. Here, the vector distance is defined as the dot product between two vectors [8]. Assume the dimension of W is $Q \times R$. The dot product between p and the i -th row of W produces the i -th element of the distance vector $\|W-p\|$, whose dimension is $Q \times 1$. The minus symbol, “-”, indicates that it is the distance between vectors. Then, the bias vector b is combined with $\|W-p\|$ by an element-by-element multiplication, .The result is denoted as $n = \|W-p\| \cdot b$. The transfer function in NN has built into a distance criterion with respect to a center. In this paper, it is defined as $radbas(n) = 2 n e^{-1}$ (1) Each element of n is substituted into Eq. 1 and produces corresponding element of a , the output vector of Radial Basis Layer. The i -th element of a can be represented as $a_i = radbas(\|W_i - p\| \cdot b_i)$ (2) where W_i is the vector made of the i -th row of W and b_i is the i -th element of bias vector b .

Some characteristics of Radial Basis Layer:

The i -th element of a equals to 1 if the input p is identical to the i th row of input weight matrix W . A radial basis neuron with a weight vector close to the input vector p produces a value near 1 and then its output weights in the competitive layer will pass their values to the competitive function. It is also possible that several elements of a are close to 1 since the input pattern is close to several training patterns. .

3) Competitive Layer:

There is no bias in Competitive Layer. In Competitive Layer, the vector a is firstly multiplied with layer weight matrix M , producing an output vector d . The competitive function, denoted as C in Fig. 2, produces a 1 corresponding to the largest element of d , and 0's elsewhere. The output vector of competitive function is denoted as c . The index of 1 in c is the number of tumor that the system can classify. The dimension of output vector, K , is 5 in this paper.

5.3. APPLICATION

- Medical application
- Research field

RESULTS:

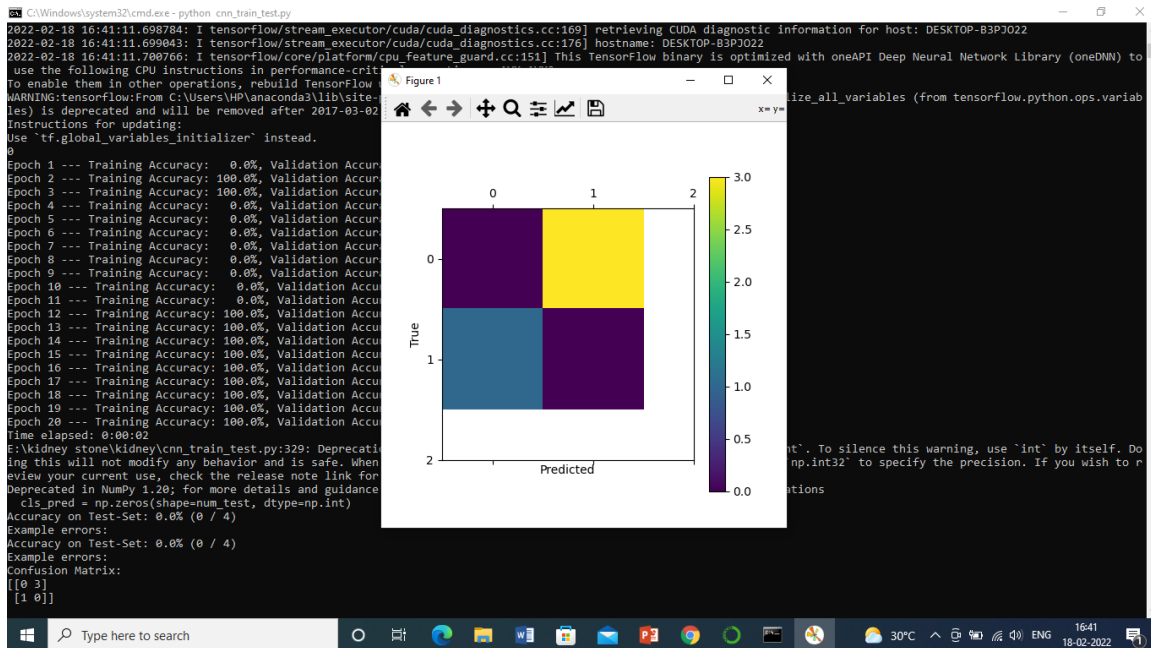


Fig.No. 5.2

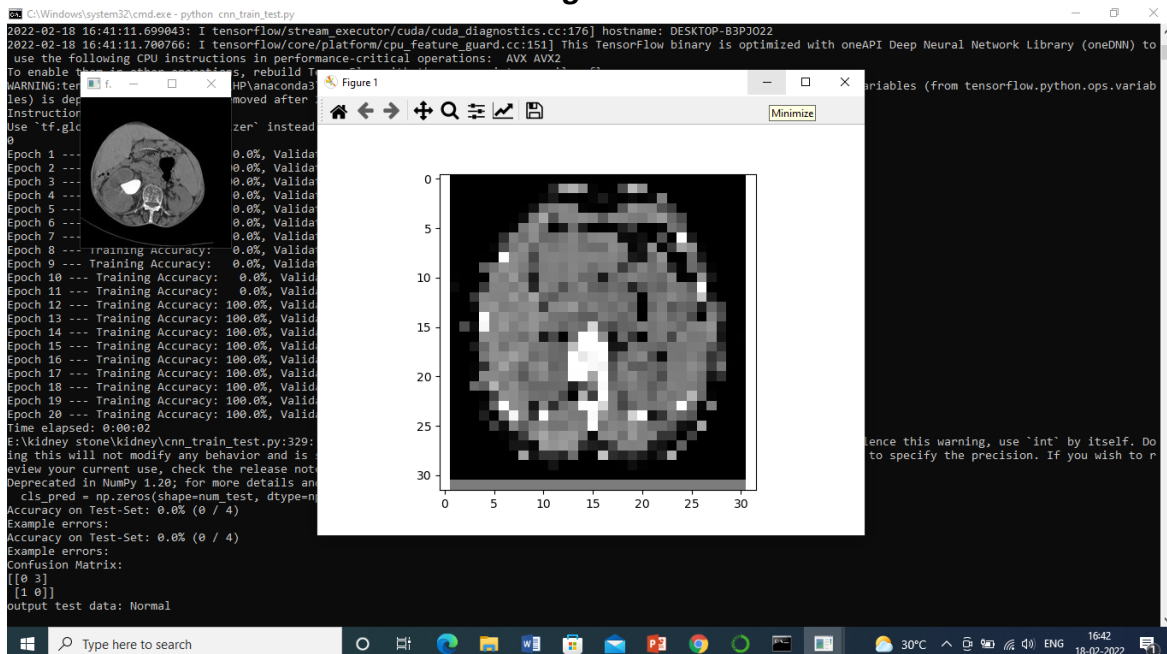


Fig.No. 5.3

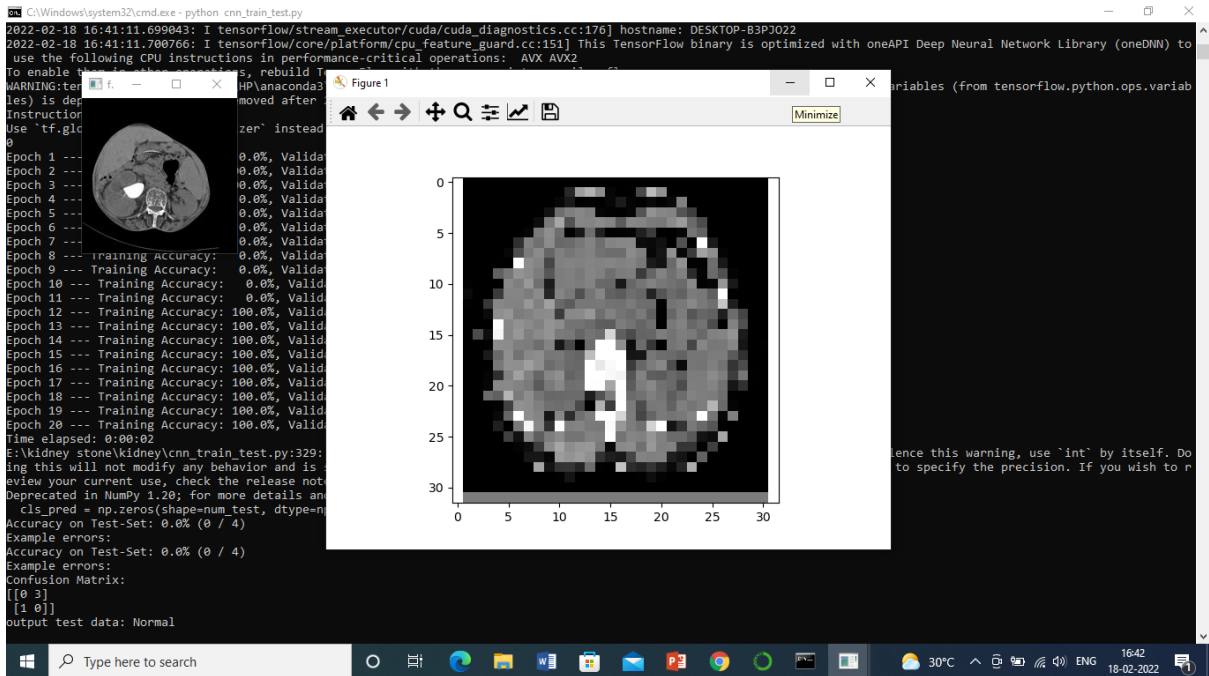


Fig . No. 5.4



Fig. No. 5.5

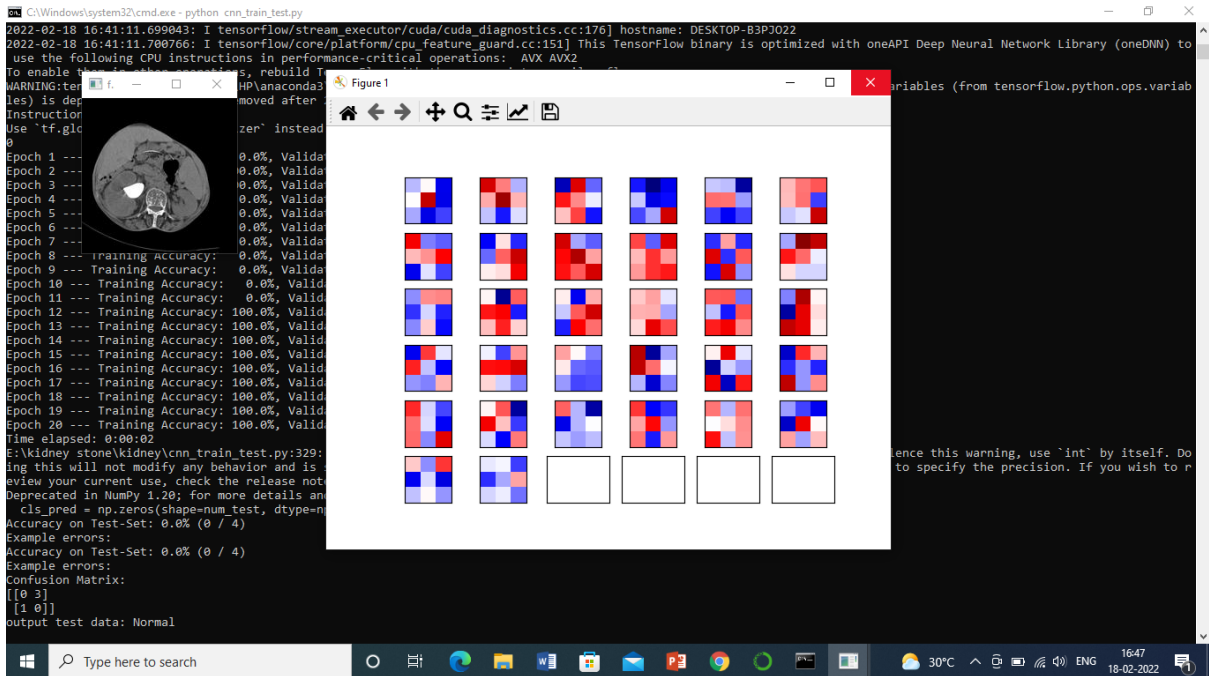


Fig. No. 5.6

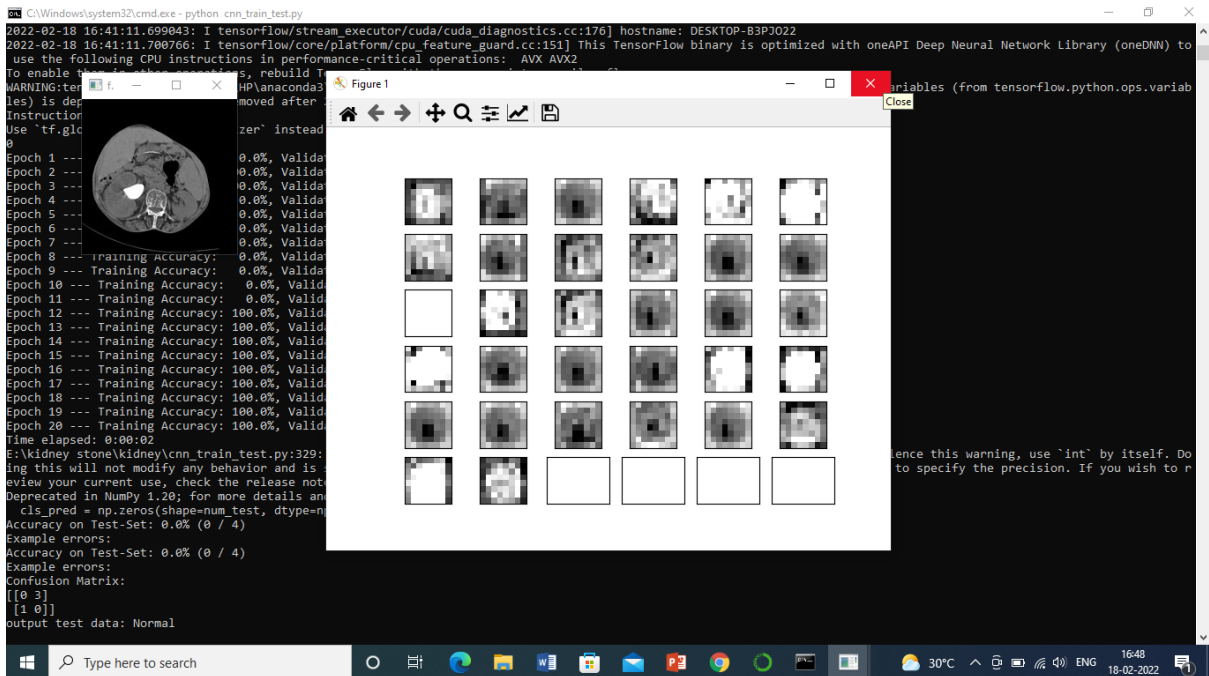


Fig. No.5.7

```
Select C:\Windows\system32\cmd.exe - python cnn_train_test.py
2022-02-18 16:41:11.699043: I tensorflow/stream_executor/cuda/cuda_diagnostics.cc:176] hostname: DESKTOP-B3PJ022
2022-02-18 16:41:11.700766: I tensorflow/core/platform/cpu_feature_guard.cc:151] This TensorFlow binary is optimized with oneAPI Deep Neural Network Library (oneDNN) to
use the following CPU instructions in performance-critical operations: AVX AVX2
To enable them in other operations, rebuild TensorFlow with the appropriate compiler flags.
WARNING:tensorflow:From C:\Users\HP\anaconda3\lib\site-packages\tensorflow\python\util\tf_should_use.py:247: initialize_all_variables (from tensorflow.python.ops.variab
les) is deprecated and will be removed after 2017-03-02.
Instructions for updating:
Use `tf.global_variables_initializer` instead.
0
Epoch 1 --- Training Accuracy: 0.0%, Validation Accuracy: 0.0%, Validation Loss: 1.119
Epoch 2 --- Training Accuracy: 100.0%, Validation Accuracy: 100.0%, Validation Loss: 0.991
Epoch 3 --- Training Accuracy: 100.0%, Validation Accuracy: 100.0%, Validation Loss: 0.897
Epoch 4 --- Training Accuracy: 0.0%, Validation Accuracy: 0.0%, Validation Loss: 0.811
Epoch 5 --- Training Accuracy: 0.0%, Validation Accuracy: 100.0%, Validation Loss: 0.646
Epoch 6 --- Training Accuracy: 0.0%, Validation Accuracy: 0.0%, Validation Loss: 0.849
Epoch 7 --- Training Accuracy: 0.0%, Validation Accuracy: 0.0%, Validation Loss: 0.854
Epoch 8 --- Training Accuracy: 0.0%, Validation Accuracy: 0.0%, Validation Loss: 0.854
Epoch 9 --- Training Accuracy: 0.0%, Validation Accuracy: 100.0%, Validation Loss: 0.595
Epoch 10 --- Training Accuracy: 0.0%, Validation Accuracy: 0.0%, Validation Loss: 0.823
Epoch 11 --- Training Accuracy: 0.0%, Validation Accuracy: 0.0%, Validation Loss: 0.875
Epoch 12 --- Training Accuracy: 100.0%, Validation Accuracy: 0.0%, Validation Loss: 0.878
Epoch 13 --- Training Accuracy: 100.0%, Validation Accuracy: 100.0%, Validation Loss: 0.618
Epoch 14 --- Training Accuracy: 100.0%, Validation Accuracy: 0.0%, Validation Loss: 0.798
Epoch 15 --- Training Accuracy: 100.0%, Validation Accuracy: 0.0%, Validation Loss: 0.938
Epoch 16 --- Training Accuracy: 100.0%, Validation Accuracy: 0.0%, Validation Loss: 0.938
Epoch 17 --- Training Accuracy: 100.0%, Validation Accuracy: 100.0%, Validation Loss: 0.678
Epoch 18 --- Training Accuracy: 100.0%, Validation Accuracy: 0.0%, Validation Loss: 0.785
Epoch 19 --- Training Accuracy: 100.0%, Validation Accuracy: 0.0%, Validation Loss: 1.057
Epoch 20 --- Training Accuracy: 100.0%, Validation Accuracy: 0.0%, Validation Loss: 1.071
Time elapsed: 0:00:02
E:\kidney stone\kidney\cnn_train_test.py:329: DeprecationWarning: `np.int` is a deprecated alias for the builtin `int`. To silence this warning, use `int` by itself. Do
ing this will not modify any behavior and is safe. When replacing `np.int`, you may wish to use e.g. `np.int64` or `np.int32` to specify the precision. If you wish to r
eview your current use, check the release note link for additional information.
Deprecated in NumPy 1.20; for more details and guidance: https://numpy.org/devdocs/release/1.20.0-notes.html#deprecations
cls_pred = np.zeros(shape=num_test, dtype=np.int)
Accuracy on Test-Set: 0.0% (0 / 4)
Example errors:
Accuracy on Test-Set: 0.0% (0 / 4)
Example errors:
Confusion Matrix:
[[0 3]
 [1 0]]
output test data: Normal
```

Fig.No. 5.8

CHAPTER – 6

SUMMARY AND CONCLUSION

The aim of the inter disciplinary project is to detect kidney stone from the person who is affected in it. In this we used back propagation neural network technique to identify the stone. We are using image processing to extract the glcm and dwt .

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 stone. For achieving higher accuracy, we can compare the results of other neural networks besides Back Propagation algorithm.

REFERENCE

- [1] Akkasaligar .P.T, Biradar .S and Kumbar .V, "Kidney stone detection in computed tomography images," 2017 International Conference On Smart Technologies For Smart Nation (SmartTechCon), 2017, pp. 353-356, doi: 10.1109/SmartTechCon.2017.8358395.
- [2] Chung .W -.Y, Ramezani .R .F,Li.C -.H,Tsai. V.F and Mayeni .M, "Development of Low-voltage Urine Sample Conductivity Measurement System for Kidney Stone Risk Assessment," 2021 IEEE 3rd Eurasia Conference on Biomedical Engineering, Healthcare and Sustainability (ECBIOS), 2021, pp. 13-15, doi: 10.1109/ECBIOS51820.2021.9510254.
- [3]Önal .E .G et al., "Kidney stone classification with electromagnetic properties: Preliminary results," 2017 25th Telecommunication Forum (TELFOR), 2017, pp. 1-4, doi: 10.1109/TELFOR.2017.8249371.
- [4] Soni .A and Rai .A, "Kidney Stone Recognition and Extraction using Directional Emboss & SVM from Computed Tomography Images," 2020 Third International Conference on Multimedia Processing, Communication & Information Technology (MPCIT), 2020, pp. 57-62, doi: 10.1109/MPCIT51588.2020.9350388.
- [5]Thein .N, Nugroho .H .A, Adjı .T .B and Hamamoto .K, "An image preprocessing method for kidney stone segmentation in CT scan images," 2018 International Conference on Computer Engineering, Network and Intelligent Multimedia (CENIM), 2018, pp. 147-150, doi: 10.1109/CENIM.2018.8710933.