VISVESVARAYA TECHNOLOGICAL UNIVERSITY
JNANASANGAMA, BELAGAVI – 590018

A Project Report
on

Controlling Adulteration in Public Food Grain Distribution

Submitted in partial fulfillment for the award of degree of

Bachelor of Engineering
in
COMPUTER SCIENCE AND ENGINEERING

Submitted by

(1BG16CS052)	**Kshithij R. Kikkeri**
(1BG16CS095)	**Shravya S Madhusudan**
(1BG16CS041)	**Harshith R M**

Internal Guide
Mrs. Kavita V Horadi
Assistant Professor
Department of CSE
BNMIT, Bengaluru

Vidyayāmruthamashnuthe

B.N.M. Institute of Technology

(Approved by AICTE, Affiliated to VTU and Accredited as Grade A Institution by NAAC)
All UG Branches- CSE, ECE, EEE, ISE & Mech. E Accredited by NBA for Academic Years 2018-19 to 2020-21 & valid up to 30.06.2021
Post box no. 7087, 27th cross, 12th Main, Banashankari II Stage, Bengaluru- 560070, INDIA
Ph: 91-80- 26711780/81/82 Email: principal@bnmit.in, www. bnmit.org

Department of Computer Science and Engineering
2019 – 2020

B.N.M. Institute of Technology

(Approved by AICTE, Affiliated to VTU and Accredited as Grade A Institution by NAAC)
All UG Branches- CSE, ECE, EEE, ISE & Mech. E Accredited by NBA for Academic Years 2018-19 to 2020-21 & valid upto 30.06.2021
Post box no. 7087, 27th cross, 12th Main, Banashankari II Stage, Bengaluru- 560070, INDIA
Ph: 91-80- 26711780/81/82 Email: principal@bnmit.in, www. bnmit.org

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Vidyayāmruthamashnuthe

CERTIFICATE

Certified that the project report entitled **CONTROLLING ADULTERATION IN PUBLIC FOOD GRAIN** carried out by **Mr. Kshithij R. Kikkeri (1BG16CS052), Ms. Shravya S Madhusudan (1BG16CS095), Mr. Harshith RM (1BG16CS041)** the bonafide students of **B.N.M Institute of Technology** in partial fulfillment for the award of **Bachelor of Engineering** in **Computer Science & Engineering** of the **Visvesvaraya Technological University**, Belagavi during the year 2019-2020. It is certified that all corrections / suggestions indicated for Internal Assessment have been incorporated in the report deposited in the departmental library. The project report has been approved as it satisfies the academic requirements in respect of Project work prescribed for the said Degree.

Smt. Kavita V Horadi **Asst. Professor** **Department of CSE** **BNMIT, Bengaluru**	**Dr. Sahana D. Gowda** **Professor and HOD** **Department of CSE** **BNMIT, Bengaluru**	**Dr. Krishnamurthy G N** **Principal** **BNMIT, Bengaluru**

Name & Signature

Examiner 1:

Examiner 2:

ACKNOWLEDGEMENT

The satisfaction and euphoria that accompany the successful completion of any task would be incomplete without the mention of the people who made it possible, whose constant guidance and encouragement crowned my effort with success.

We would like to thank **Shri. Narayan Rao R. Maanay,** Secretary, BNMEI, Bengaluru for providing excellent academic environment in the college.

We would like to sincerely thank **Prof. T. J. Rama Murthy**, Director, BNMIT, Bengaluru for having extended his support and encouraging us during the course of the work.

We would like to express our gratitude to **Prof. Eishwar N Maanay**, Dean, BNMIT, Bengaluru for his relentless support, guidance and assistance.

We would like to thank **Dr. Krishnamurthy G.N**, Principal, BNMIT, Bengaluru for his constant encouragement.

We would like to thank **Dr. Sahana D Gowda**, Professor and Head of the Department of Computer Science and Engineering who has shared her opinions and thoughts which helped us in the completion of our project successfully.

We consider it is a privilege to express gratitude to our guide **Mrs. Kavita V Horadi,** Assistant Professor, Department of Computer Science and Engineering, BNMIT who has given us all the support and guidance in completing the project successfully.

We would also like to thank our project coordinators **Mrs. Sajitha N**, Assistant Professor, Department of Computer Science and Engineering, BNMIT and **Mrs. Manikantha K** Assistant Professor, Department of Computer Science and Engineering, BNMIT for guiding in a systematic manner.

Kshithij R. Kikkeri (1BG16CS052)

Shravya S Madhusudan (1BG16CS095)

Harshith RM (1BG16CS041)

Table of Contents

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1

INTRODUCTION

1.1 PREFACE

Almost as soon as digital computers became available, it was realized that they could be used to process and extract information from digitalized images. Need of accurate grading, sorting of fruits and foods, or agriculture products arises because of increased expectations in quality food and safety standards. This is being assessed through visual inspection by human inspectors. This process is tedious and time consuming. After hours of working the operator may lose concentration which in turn will affect the evaluation process. The farmers are very much affected by this manual activity in terms of returns for their crop. Hence these tasks require automation, so as to have a computer vision system as an alternative to this manual practice. Automated system of sorting food and agriculture products provides rapid and hygienic inspection with computer vision. Computer vision and image processing are non-destructive, accurate and reliable methods to achieve target of grading. Machine Vision Systems are successfully used for Identification and Classification of plants, leaves, flowers, bulk grain samples. In order to perform this task of pattern recognition by machines, considerable design effort is necessary. Characterization models were based on morphological features, color features or textural features. After isolating the grain, the region of interest was selected around the boundary of the edge. The morphological features were obtained from the binary images containing only pixels of the grain edge. Grain quality is a term that refers to the quality of grain. However, what constitutes quality depends on the use of the grain. Overall quality of grain are affected by several factors which includes, growing practices, time and type of harvesting, postharvest handling, storage management and transportation practices. Food is a basic need of life. Without food no one can survive. So, it is a basic need to have food daily which should be of good quality. As India is a highest producer of wheat and rice (agriculture) across the globe, people don't get good quality of food. The quality of food is an important factor for proper nourishment and today's market should be free of adulterated food grains. These grains consist of several impurities like stones, damaged seeds, broken granules etc. The addition of impurities in food affects the composition and quality of food. There is no convenient method to identify these inferior quality grains in the market. Human perception based on visual inspection has long been recognized as a guide to quality assessment but the results are not accurate and reliable. To overcome this problem, image processing has been used to classify food grains according to its quality. The challenges are: x Quality issue x Automation in quality assessment. In this paper, for grain quality assessment we have used wheat and rice grains. Our system divides food in three classes i.e. good, bad and medium according to training.

1.2 MOTIVATION

Quality of grains is an important requirement for today's market, to protect the consumers from substandard products. The government imposes price control for essential commodities in order to protect the consumers from black marketing and inflated prices. As a result, some traders unethically release sub-standard products to the consumer market. Because of such practices there are so many inferior quality grains arriving to the market day by day. These grains consist of several impurities like stones, damaged seeds, more broken granules etc. This is often seen today in rice trade where rice of low quality is sold without being noticed. However, there is no convenient method to identify these inferior quality grains in the market. Therefore, this has become a serious issue for both the consumer and the government. Hence an automated quality analysis of the food grains which are distributed could be considered to be extremely helpful.

1.3 PROBLEM STATEMENT

The existing system proves the inability of the bureaucratic setup which has failed to deliver good quality grains to the needy. The aim of this project is to overcome the failures of the present system and to design an automated grain recognition which performs quality analysis of rice grains using its features, which classifies the type of grain and its quality and grade.

1.4 OBJECTIVES

The aim of this project is to develop a real-time application that can classify the type of grain given according to its quality.

1. To capture the image of the grain using a digital camera.
2. To store them as the dataset.
3. To pre-process the image.
4. To perform segmentation to extract each grain image.
5. To extract the features from each segmented image, store the extracted features in feature vector for training.
6. To build the PNN for training and recognizing the grain type and its quality. Finally, test the system by giving different images as input.

1.5 SUMMARY

The intent of this project is to address and try to minimize the food adulteration and contamination of the ration distributed to the poor and the needy, in possession of Below Poverty Line (BPL) card holders by deploying smart sensing devices with Internet of Things (IoT) to preserve the quality and quantity of food grains stored at warehouse and packaging with appropriate quantity and assigning barcode for each package and commodity. The BPL card holders will be advised to open the sealed package after the ration card is validated. With this project, we are proposing to change the distribution system, by packaging the ration in a secured packing after checking the quality of the commodity.

CHAPTER 2

LITERATURE SURVEY

2.1 INTRODUCTION

Neural networks are frequently employed to classify patterns based on learning from examples. Different neural network paradigms employ different learning rules, but all in some way determine pattern statistics from a set of training samples and then classify new patterns on the basis of these statistics. It has been proven that the structure of the Probabilistic neural network, while similar in structure to back-propagation and differing primarily in that the sigmoid activation function is replaced by a statistically derived one, has the unique feature that under certain easily met conditions the decision boundary implemented by the probabilistic neural network (PNN) asymptotically approaches the Bayes optimal decision surface.

There is a striking similarity between parallel analog networks that classify patterns using nonparametric estimators of a PDF and feed-forward neural networks used with other commonly known training algorithms. Figure 2.1 shows a neural network organization for classification of input patterns X into two categories. In Figure 2.2, the input units are merely distribution units that supply the same input values to all of the pattern units. Each pattern unit forms a dot product of the input pattern vector X with a weight vector $\mathbf{W_i}$, $Z_i = \mathbf{X} \bullet \mathbf{W_i}$, and then performs a nonlinear operation on Zi before outputting its activation level to the summation unit. Instead of the sigmoid activation function commonly used for back-propagation the nonlinear operation used here is $\exp[(Z_i - 1)/ \sigma^2]$. Assuming that both X and W_i are normalized to unit length, this is equivalent to using the equation

$$\exp[-(\mathbf{W_i} - \mathbf{X})'(\mathbf{W_i} - \mathbf{X}) / 2\sigma^2]$$

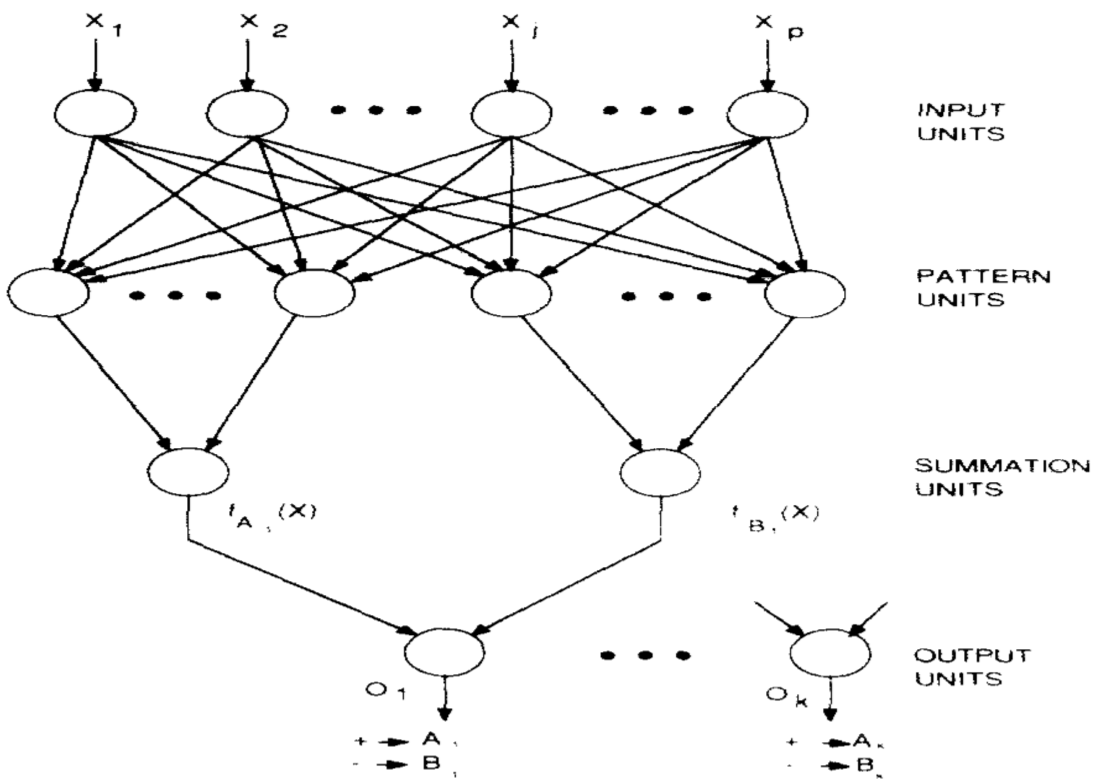

Figure 2.1 Architecture of classification of patterns into categories

The summation units simply sum the inputs from the pattern units that correspond to the category from which the training pattern was selected. The output, or decision, units are two-input neurons as shown in Figure 2.3. These units produce binary outputs.

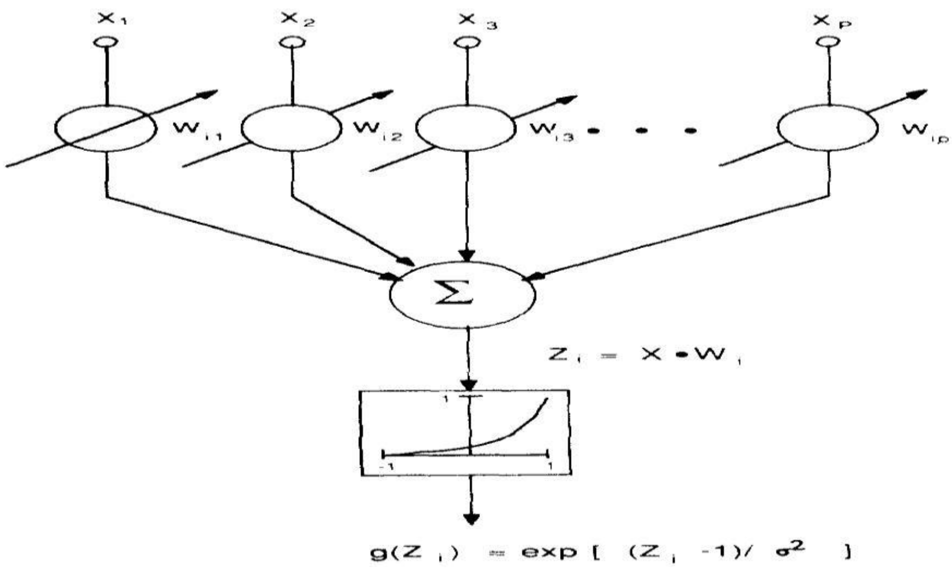

Figure 2.2 The Pattern Unit

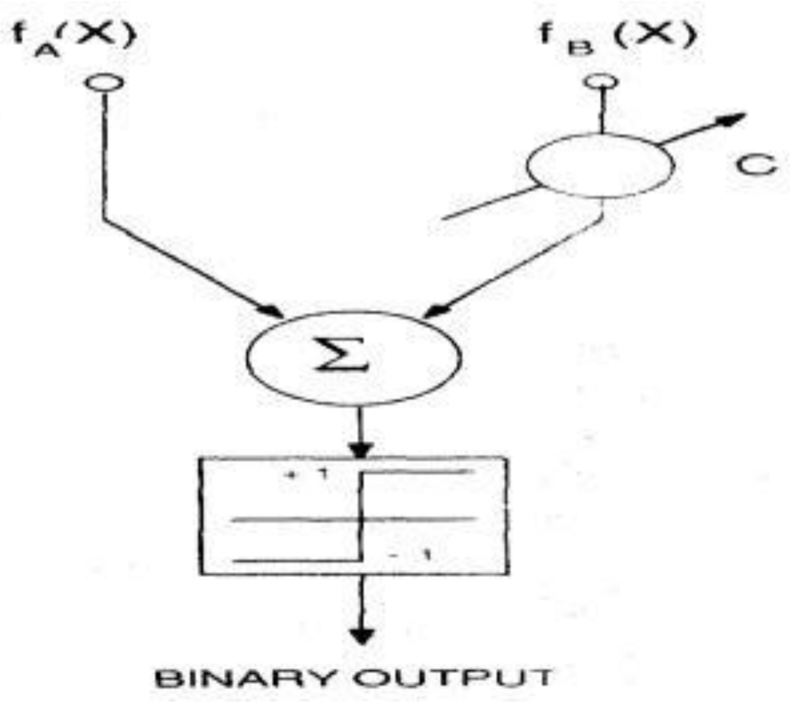

Figure 2.3 The Output Unit

The network is trained by setting the W_i weight vector in one of the pattern units equal to each of the X patterns in the training set and then connecting the pattern unit's output to the appropriate summation unit. A separate neuron (pattern unit) is required for every training pattern. As indicated in Figure 2.1, the same pattern units can be grouped by different summation units to provide additional pairs of categories and additional bits of information in the output vector.

Over the last few years, there has been a lot of research that has been done on Grain Quality Detection by using Image Processing. Various researchers have proposed systems with different methods and algorithms with different accuracies. But the accuracy of these systems is not 100 percent. Given below are the different papers on which the literature survey has been conducted.

2.2 LITERATURE SURVEY

There is research which has been done on the Grain Quality Detection using Image Processing over the past few years. Different researchers have proposed systems with different accuracies using different methods and algorithms. But these systems are not effective and not very accurate. The numerous papers on which the literature survey was conducted are given below.

2.2.1 Grain Quality Detection by using Image Processing for public distribution

Deepika Sharma and Sharad D. Sawant, IEEE International Conference on Intelligent Computing and Control Systems, 2017

In our life food is necessary for nourishment and sustenance. There are some additional impurities present in the food such as stones, damaged seeds, broken granules which affects the composition and the quality of food. Determining the quality of grains is a big challenge. Wheat and rice are used by most of the population across the world. In this paper, we have proposed a system that determines the quality of food. Initially, the grain samples run on the conveyor belt and then random images of grains are captured by the camera. The image processing algorithm is applied on the grain samples through MATLAB. The classification has been done according to colour, shape and size. It results good, bad and medium quality by using Neural Network (NN) classifier. The final output is displayed on the LCD also the message will be sent to higher authority through GSM module. This system can be implemented in food industries at later stage for grading purpose which will ma the task of classification of grains simpler for the public.

2.2.2 Image Processing Techniques for Analysing Food Grains

Harpreet Singh, Chandan Singh Rawat IEEE International Conference on Computing Methodologies and Communication (ICCMC), 2019

Food consumed in daily diet consists of fruits, cereal grains and spices. Cereal grains are considered to be the most important part as it meets the nutrition needs of the human population. It is necessary to check the quality of food before consuming as it directly impacts on health. Amongst the various food analysis techniques this paper focuses on a semi-automated, an image processing and two machine learning techniques with their advantages and limitations. The quality parameters of grain are size, area, major axis length, minor axis length and perimeter. These features of grain are extracted from the image of grain sample. Based on these features, grain quality is assessed.

2.2.3 Rice Grain Identification and Quality Analysis using Image Processing based on Principal Component Analysis

Muhammad Junaid Asif, Tayyab Shahbaz, at IEEE International Symposium on Recent Advances in Electrical Engineering (RAEE), 2018

Different types of foods are available in grain form, but rice is one of the important and most used cereal grains of Pakistan and all over the world. Quality inspection of rice grain is also important for both local as well as export purpose. It is necessary to propose an automatic solution to perform the quality analysis as well as to distinguish between different classes of rice. Main purpose of this paper is to present an image processing-based solution to classify the different varieties of rice and its quality analysis. An approach based on the combination of principal component analysis and canny edge detection is used for the classification. Quality analysis of rice grain is determined by morphological features of rice grains. These morphological features include eccentricity, major axis length, minor axis length, perimeter, area and size of the grains. Six different varieties of rice are classified and analyzed in this paper. A database is trained by feeding the 100 images of each variety of rice grains. Classification and quality analysis is done by comparing the sample image with database image. Canny edge detector is applied to detect the edges of rice grains. Eigen values and Eigen vectors are calculated on the basis of morphological features. Then by applying the PCA, different varieties of rice are classified by comparing the sample image with a database. Results obtained in terms of classification and quality analysis are 92.3% and 89.5% respectively. Proposed system can work well within minimum time and low cost.

2.2.4 Assessment of Quality of Rice Grain using Optical and Image Processing Technique

Engr. Zahida Parveen , Dr. Muhammad Anzar Alam , Engr. Hina Shakir IEEE International Conference on Communication, Computing and Digital Systems (C-CODE) 2017

Rice is the most favourable and most consuming food for human being in all over the world and researchers are working to improve the quality of rice. The quality measurement of rice is also important because it is consumed as food as well as it is used for milling process in the national and international market. Many researchers have already worked on the quality of grain and proposed different techniques to characterize the quality of rice. Chalky is whiteness part in the rice grain and it is one of the most important parameter that is used to evaluate the quality of rice grain. We proposed an image processing technique using extended maxima operator to detect the chalky area in the rice. We also calculated the dimensions and color to classify rice grains. The experiment was performed on 22 sample images of rice grain to test the proposed method and was validated using visual inspection.

2.2.5 Improving the Grain Quality Assessment Fusing Data from Image and Spectra Analyses

Miroljub Mladenov, Stanislav Penchev and Tsvetelina Draganova, IEEE International Conference Intelligent Systems, 2012

The paper presents approaches, methods and tools for assessment of main quality features of grain samples that are based on color image and spectra analyses. Visible features like grain color, shape, and dimensions are extracted from the object images. Information about object color and surface texture is obtained from the object spectral characteristics. The categorization of the grain sample elements in three quality groups is accomplished using two data fusion approaches. The first approach is based on the fusion of the results about object color and shape characteristics obtained using image analysis only. The second approach fuses the shape data obtained by image analysis and the color and surface texture data obtained by spectra analysis. The results obtained by the two data fusion approaches are compared.

2.2.6 Grain quality assessment for rationing system

Shraddha N. Shahane and S. D. Sawant, Online IEEE International Conference on Green Engineering and Technologies (IC-GET), 2016

Food is the basic need of life. The quality of food is very important factor for proper nourishment of life so that, there is a requirement of assessment of food quality. In this paper, we proposed the system that assesses the food grain quality using image processing. It is proposed to work at ration shop to avoid the distribution of low quality grain. In this system, visual inspection for quality assessment is replaced by image processing technique. The quality parameters of grain are size, area, major axis length, minor axis length and perimeter. These features of grain are extracted from the image of grain sample. Based on these features, grain quality is assessed. This system distributes ration (food and non-food items) at subsidized price to poor. There are many challenges in this system like quantity issue, lower quality of food, system transparency etc.

2.2.7 Colour and Texture Based Identification and Classification of food Grains using different Colour Models and Haralick features

Neelamma K. Patil, Virendra S. Malemath and Ravi M. Yadahalli, International Journal on Computer Science and Engineering (IJCSE) 2017

This paper presents the study on identification and classification of food grains using different color models such as L*a*b, HSV, HSI and YCbCr by combining color and texture features without performing pre-processing. The K-NN and minimum distance classifier are used to identify and classify the different types of food grains using local and global features. Texture and color features are the important features used in the classification of different objects. The local features like Haralick features are computed from co-occurrence matrix as texture features and global features from cumulative histogram are computed along with color features. The experiment was carried out on different food grains classes. The non-uniformity of RGB color space is eliminated by L*a*b, HSV, HSI and YCbCr color space. The correct classification result achieved for different color models is quite good.

2.3 Existing System

The existing systems suggest that the manual classification and segregation can be done using the processes of sieving, cleaning, milling among the others. On the other hand, automated process using advanced techniques are done using algorithms like K-Means clustering algorithm, K-Nearest neighbor algorithm, morphological feature extraction and Detection. Not highly secure and hence leads to inconsistency and tampering of food grains for illicit purposes. Immediate disbursal of the food grains does not take place which results in rotting of food grains due to inexplicable weather conditions.

2.4 Proposed System

The process starts before the customer encounters the food grains. It's an initial process in food industries. Henceforth, the cost is decided for each type of grain. In this system, images of wheat and rice samples are taken by a camera and then it's given to the image processing unit. All these food grains are run along a conveyor belt and the images are captured randomly of different samples. The features of the colour, shape and size of the rice grains in order to grade them for quality. A unique bar code will be generated for each commodity and each warehouse. The bar code will be attached to the package at a place which is easily noticeable. The FPS dealer verifies the ration card of the BPL card holder and if it is successfully verified, the dealer determines how much quantity of ration, the card holder is eligible. The information of the package delivered to the card holder will be stored in cloud. This will help is getting the quantity of ration handed over to the card holder, from which FPS the package was delivered and the type of commodity.

2.4 Methodologies

In this section, a detailed description about the various modules (figure 2.4) in the proposed project are elucidated.

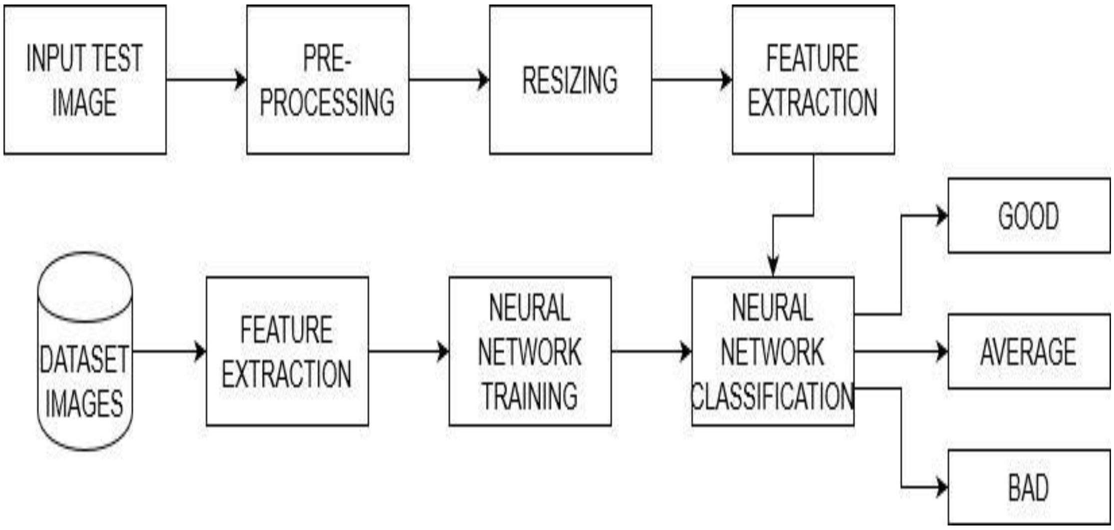

Figure 2.4 Model Framework

2.4.1 Image acquisition

The rice grains taken from the silos are manually examined for the freshness and other impurities. Randomly images of the rice grains are taken, preferably in the same background, so as to eliminate the process of background filtering. The images are taken by using a high definition camera so as to get good quality images for further processes

2.4.2 Image Pre-processing

Preprocessing is one of the important steps for the enhancement of quality of the captured image. The preprocessing methods use a small neighborhood of a pixel in a image, to get a new brightness value in the output image. This Research proposes to apply the Gaussian filter for image smoothening. Also this process involves a threshold method in order to eliminate background. Then the grayscale image is binarized. Once the image is binarized morphological operations are performed, first erosion operation is used to eliminate the shadow of the grains, this is followed by dilation to enhance the image after the erosion and improve the boundary sharpness.

2.4.1.1 Image filtering

Images are often corrupted due to the variation in illumination, intensity or may have poor contrast and can’t be used directly. Filtering helps to transform pixel intensity values to reveal certain image characteristics

- Enhancement: helps to improve the contrast of the image
- Smoothing: Remove the noise from the image.

2.4.1.2 Background Elimination

Background elimination which is also called as Foreground Detection, is a technique in the fields of computer vision and image processing wherein an image's foreground is extracted for further processing, i.e. Object Recognition. Usually the images region of interest are a part of image foreground. Here checking the intensity of each pixel of the image with a precalculated value, and the pixel values falling with this range are set to zero.

2.4.1.3 Binarization

Binarization of an image is a process representing an image using only two different pixel values. It is generally performed by classifying a gray scale image into two groups of pixels based on certain threshold value. Those pixel values greater than or equal to the threshold is set to a particular grey value and those below the threshold to another grey value. The quality of the binary image is much dependent on how appropriately the threshold for binarization are chosen or how fairly the pixels are classified into two groups of pixels.

2.4.1.3 Morphological Operations

Morphological image processing is a collection of non-linear operations related to the shape or morphology of features in an image. Morphological techniques probe an image with a small shape or template called a structuring element. The structuring element is positioned at all possible locations in the image and it is compared with the corresponding neighborhood of pixels.

2.4.3 Image Segmentation

The aim of image segmentation is to cluster pixels into salient image regions. Image segmentation is an essential preliminary step in most automatic pictorial pattern recognition and scene analysis problem. Here performing the segmentation using Component Labeling plays an important role. Once the image is binarized, perform labeling of connected components. By using labels and the similarity of grey level values, grains are segmented.

2.4.4 Feature Extraction

In this process some qualitative information is being extracted from the objects to be analyzed in the image. These extracted attributes are called features and a pattern is defined as a vector of such features. The various features that could be extracted are color features, geometrical features and texture features. In this method 3 color features and HoG, GLCM and Chalkiness features need to be extracted.

a) Histogram of Oriented Gradients

The Histogram of Oriented Gradients (HOG) is a feature descriptor used in computer vision and image processing for the purpose of object detection. The technique counts occurrences of gradient orientation in localized portions of an image.

b) Gray Level Co-Occurrence matrix

A gray level co-occurence matrix (GLCM) is a histogram of co-occurring grayscale values at a given offset over an image. To describe the texture of an image it is usual to extract features such as entropy, energy, contrast, correlation, etc. from several co-occurrence matrices computed for different offsets. In this case the entropy is defined as follows:

$$-\sum_{i=0}^{n-1}\sum_{j=0}^{n-1} p(i,j)\log_b p(i,j)$$

where n and b are again the number of gray levels and the base of the logarithm function, respectively, and $p(i,j)$ stands for the probability of two pixels separated by the specified offset having intensities i and j

2.5.5 Classification

The classification approach is mainly based on the assumption that the digital image under consideration depicts one or more features, and these features correspond to one of the several distinct and exclusive classes. The two phases that are typically employed by the classification algorithms are training and testing. In the initial training phase, characteristic properties of typical image features are isolated and, based on these, a unique description of each classification category, i.e. training class, is created. In the subsequent testing phase, these feature-space partitions are used to classify image features. In this method we use the PNN classifier. When an input is presented, the first layer computes distances from the input vector to the training input vectors and produces a vector whose elements indicates how close the input is to a training input. The second layer sums these contributions for each class of inputs to produce as its net output a vector of probabilities. Finally, a complete transfer function on the output of the second layer picks the maximum of these probabilities to choose the class.

2.5 Summary

The goal of this project is to resolve and attempt to reduce food adulteration and ratio n contamination, distributed to the poor and needy, in possession of Below Poverty Li ne (BPL) card holders by deploying smart sensing devices with the Internet of Things (IoT) to conserve the quality and quantity of food grains stored in warehouses and pac kaging with the correct quantity and allocation.

CHAPTER 3

SOFTWARE AND HARDWARE REQUIREMENTS

3.1 INTRODUCTION

The requirement analysis specifies the requirements needed to develop a project. The developer needs to have clear and thorough understanding of the product to be developed in order to derive the requirements. The requirement analysis includes the tasks that determine the needs or conditions to develop a new product or alter an existing product taking account of all possible requirements of different stakeholders, analyzing, documenting, validating and managing the requirements of a system. Requirement analysis is a crucial phase as it decides the success or failure of a product. It should be documented, implementable, testable, traceable and defined to a level sufficient enough to design a system. Requirements can be collected using use cases, scenarios, prototypes, feedback from the customer and management.

Nonfunctional requirements describe the general characteristics of a system. They are also known as quality attributes. Functional requirements describe how a product must behave, what its features and functions.

3.2 SOFTWARE AND HARDWARE REQUIREMENTS

The components which are employed for the smooth running of the proposed system are elucidated as below in terms of software and hardware requirements.

3.2.1 SOFTWARE REQUIREMENTS

Software is commonly known as programs or apps, consists of all the instructions that tell the hardware how to perform a task. The required software support is:

- Operating System: Windows 7 or higher.
- MATLAB2018a

3.2.2 HARDWARE REQUIREMENTS

Hardware refers to the physical elements of a computer. This is also sometime called the machinery or the equipment of the computer. The required hardware support is:

- Processor : Intel Core i5-7400 or Core i7-7700k clocked @ 3.40Ghz
- Operating System type : Windows 10 pro 64 bit version 1803
- RAM : 16GB DDR3
- Hard Disk : 1 TB at 7200 RPM

3.2.3 FUNCTIONAL REQUIREMENTS

A functional requirement describes what a software system should do, the way it functions. A functional requirement defines a function of a system and its components. A function is described as a set of inputs, the behavior, and outputs. The proposed system has the following functional requirements:

- The images of the rice grains should be collected.
- The system pre-processes the image to obtain the parameters.
- The system shall be trained with an existing data set and will generate whether the quality of rice grain is good/average or bad

3.2.4 NON-FUNCTIONAL REQUIREMENTS

A non-functional requirement deals with operations and constraints on how the system will do so. Non-functional requirements are not straight forward requirement of the system rather it is related to usability. A non-functional requirement is a requirement that specifies criteria that can be used to judge the operation of a system, rather than specific behaviors. The proposed system has the following non-functional requirements:

- Portability: The application is developed in Matlab. It would be portable because it is platform independent, object-oriented programming language.
- Extensibility: The project work is also open for any future modification and hence the work could be defined as the one of the extensible works.
- Availability: It is the probability that a system, at a point in time, will be operational and able to deliver the requested services.

- Reliability: It describes the ability of a system or component to function under stated conditions for a specified period of time.
- Security: The image data should be confidential and not available to be used by others.
- Response: The system should provide a quick response.

3.3 SUMMARY

The requirements of this project include a Windows 10 system with MATLAB R2018a and a dataset from Github Repository in order to design the system such that the detection of the quality of rice grains can be automated without human intervention.

CHAPTER 4

SYSTEM DESIGN

4.1 INTRODUCTION

Once the requirements are listed and acquired, the data flow within the system must be identified. Data Flow Diagrams (DFDs) show the way that a system processes data. The DFDs illustrate how the data flows from input through various processes to the output, showing every step the system takes to process the data.

4.2 PROPOSED MODEL

The proposed model for classification of rice grains includes two actors: the admin and the consumer. The admin has a job of scanning the rice grains and upload the scanned image onto the system. The system automatically scans the grain image to grade of rice and gives a detailed report. It also classifies the accuracy of the quality of the rice grain under consideration. There are different phases (Figure 4.1) involved in the project, they are:

- In the pre-process phase, the rice grain image will be analyzed by the system and it will undergo all pre-processing techniques like gray scaling, binarization, thinning, erosion and dilation.
- In the prediction phase, the network utilizes two hidden layers
- A network classifier is created to determine the quality of the test grain used. The classifier will classify the retinal image from 3 categories – Good quality, Average quality and Bad quality.

4.3 DATASET

- The dataset is obtained from GitHub repository taken by a well-known open source contributor.
- All the grain images are taken from the same camera and is stored in a database in a .jpg file extension. The images are initially classified as good, average and bad quality grains based on the perception of the database creator.

- The training data is comprised of 1,500 images, which are augmented during preprocessing.

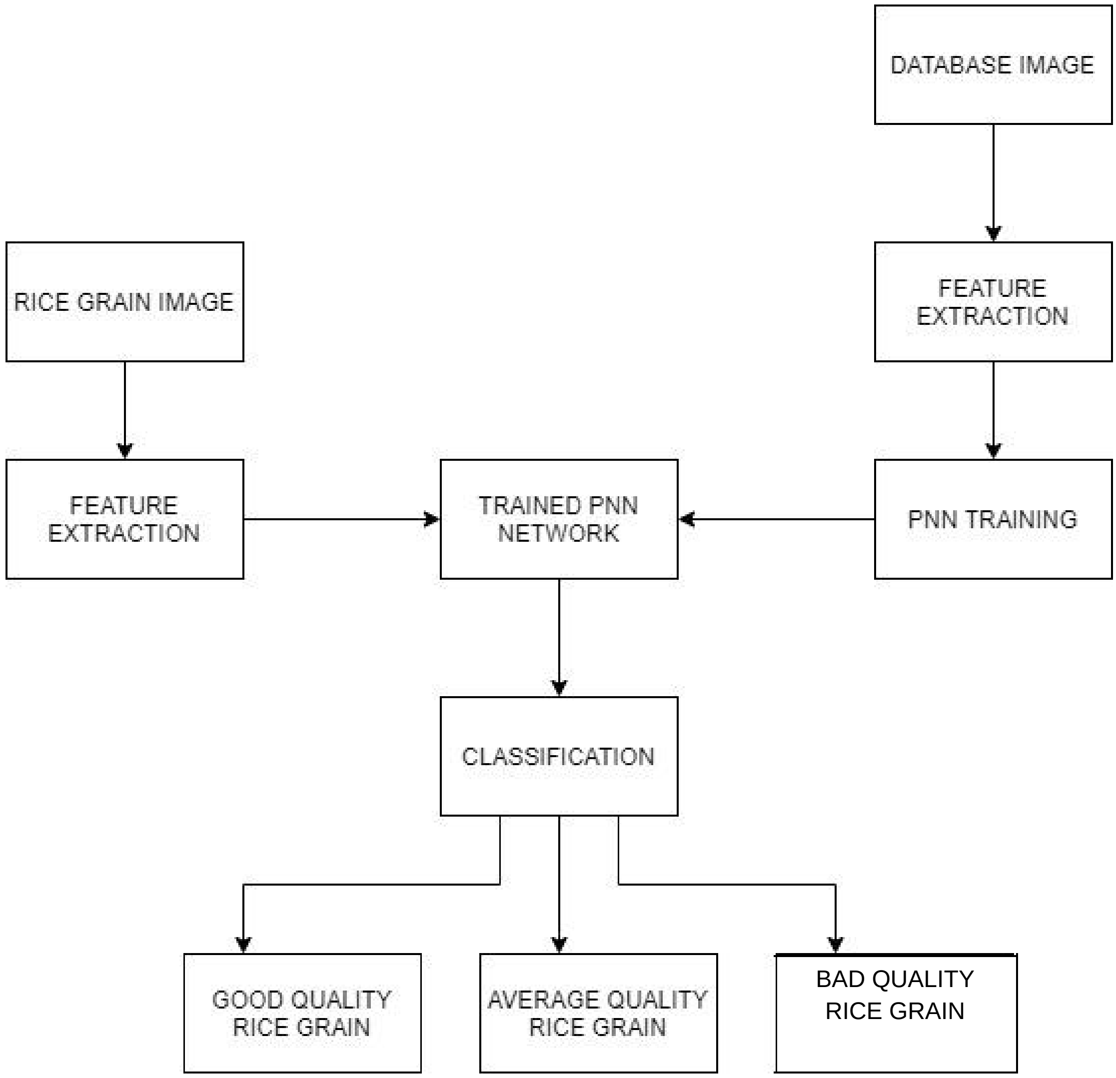

Figure 4.1 Proposed system architecture.

4.4 DATA FLOW DIAGRAM / SEQUENCE DIAGRAM

Data flow diagrams (DFDs) are popular for software design as the diagram makes the flow of data easy to understand and analyze. DFDs represent the functions a system performs hierarchically, starting with the highest-level functions and moving through various layers or levels of sub functions. As a modeling technique, DFDs are useful for performing a structured analysis of software problems, allowing developers to spot and pinpoint issues in software development.

Sequence diagram in UML are used to model the interaction between the actors and objects in a system and between the objects themselves.

4.4.1 DFD LEVEL 0

The level 0 DFD shown in Figure 4.2, represents a general data flow in the detection process. The admin uses different grain images to detect the quality of rice grain under consideration. After detection of the quality is completed, the result is returned to the admin.

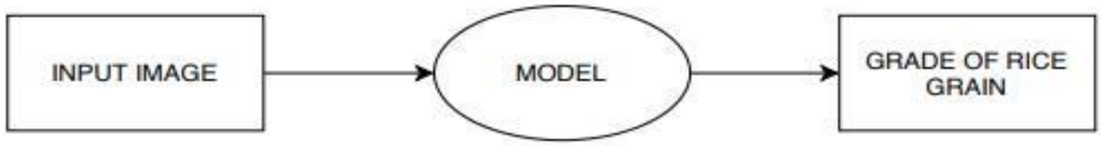

Figure 4.2 Level 0 DFD

4.4.2 DFD LEVEL 1

The level 1 DFD (Figure 4.3) describes the process of detecting the disease. The model first trains itself by using the pre-defined dataset. This involves the pre-processing of the image captured and then sending it to the proposed model for processing.

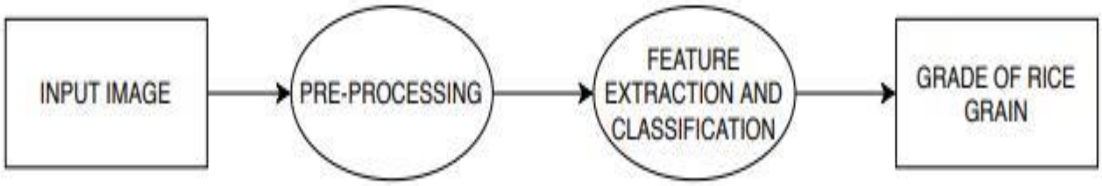

Figure 4.3 Level 1 DFD

4.4.3 DFD LEVEL 2

The level 2 DFD (Figure 4.4) describes how the entire process of ascertaining the quality of rice grain carried out at different stages. It clearly describes the multiple stages involved in the process of classification of the grain image. The end result obtained is the classification of the disease into multiple stage classes as described earlier.

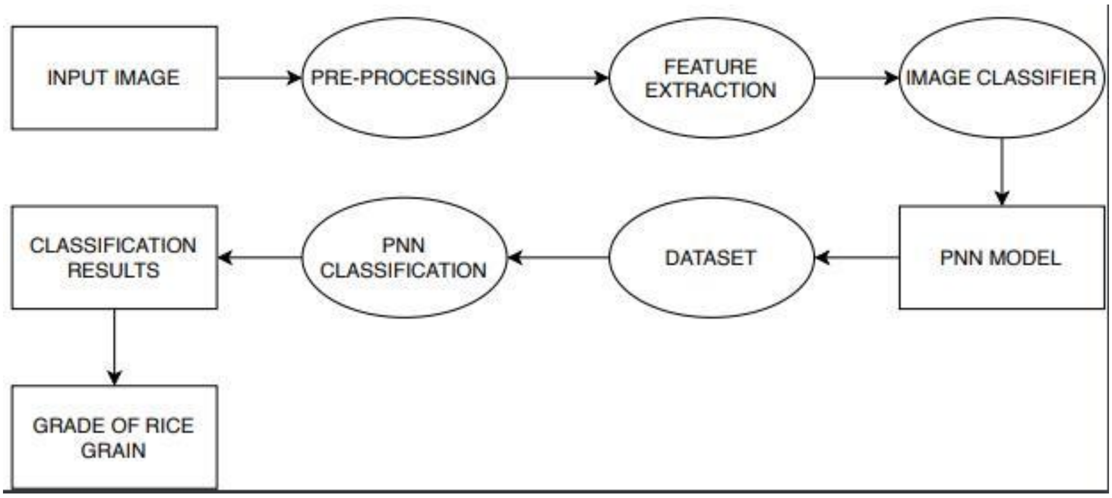

Figure 4.4 Level 2 DFD

4.4.4 SEQUENCE DIAGRAM

The sequence diagram (Figure 4.5) shows the sequence of steps in the detection process of Diabetic Retinopathy disease from patient’s retinal image. The system takes the retinal image from the patient/user as input. The image is pre-processed to rectify the imbalance. The pre-processed images are then given as input to the prediction model which is trained using a pre-defined dataset. The result will be provided with a class classification based on the stage of the disease.

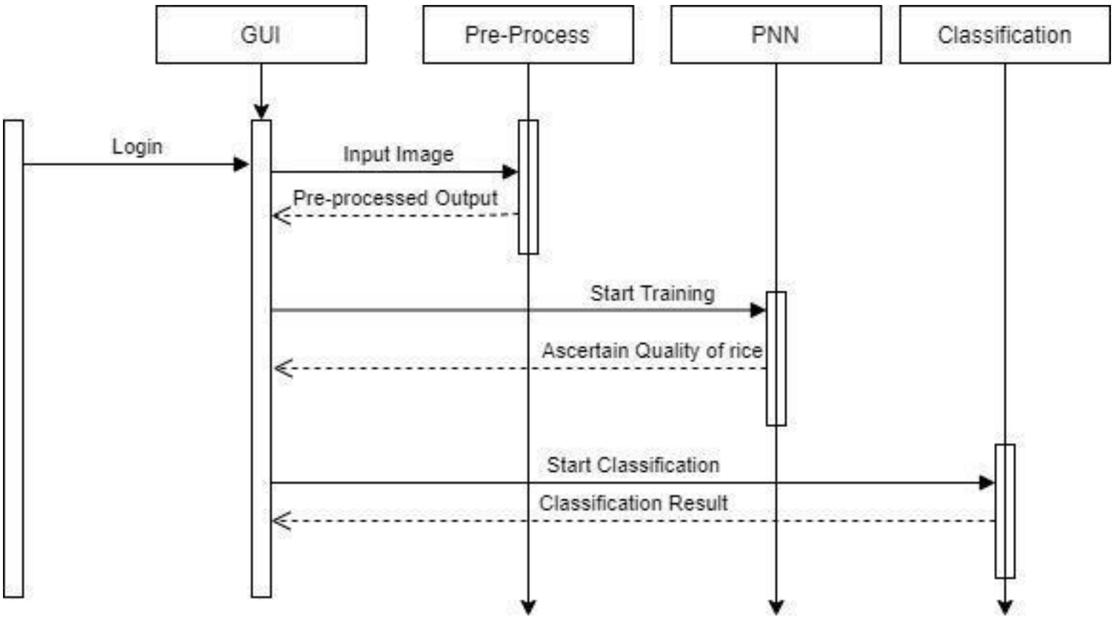

Figure 4.5 Sequence Diagram

4.5 SUMMARY

A model is proposed for detection of the quality of rice grain image is an extensive process which makes use of novel and complex state-of-the-art neural networks. The different levels of dataflow diagrams and the overall sequence diagram included in the diagnosis process were discussed and explained briefly.

CHAPTER 5

IMPLEMENTATION

5.1 INTRODUCTION

Recent advances in probabilistic neural networks (PNNs) have made it a state-of-the-art technique in image classification tasks, and its variants have begun to dominate many fields in computer vision, such as object detection, image classification, object tracking, edge detection. Instead of making essential use of handcrafted features, PNN can learn a hierarchy of features, which can be used for image classification purposes. As the hierarchy approach is available to learn more complex features, as well as translation and distortion features in higher layers, the accuracy of the PNN-based image classification method can be higher. This assumption explores the use of the PNN-based method for the classification of rice grain test in this work. Moreover, a specific multi-layer PNN architecture is designed, and experiments are conducted on real retina data. In addition, the results obtained demonstrate the assumptions that achieve more than 90% accuracy, which ranks as the highest in comparison with previous handcrafted feature-based classifiers.

5.2 SYSTEM DESIGN

System design is the process of designing the elements of a system such as the architecture, modules and components, the different interfaces of those components and the data that goes through that system. The purpose of the System Design process is to provide sufficient detailed data and information about the system and its system elements to enable the implementation consistent with architectural entities as defined in models and views of the system architecture.

Elements of a System are -

1. Architecture - This is the conceptual model that defines the structure, behavior and more views of a system. The use flowcharts to represent to illustrate the architecture.
2. Modules - These are the components that handle one specific task in a system. A combination of the modules makes up the system.

3. Components - This provides a particular function or group of related functions. They are made up of modules.
4. Interfaces - This is the shared boundary across which the components of the system exchange information and relate.
5. Data - This the management of the information and data flow.

5.2.1 ALGORITHMS / METHODS USED

The following list represents the list of modules that are used to form the recommender system –

1. Input image
2. Grayscale Conversion
3. Resizing and cropping
4. Pre-processing
5. HoG feature extraction
6. GLCM feature extraction
7. Chalkiness feature extraction
8. Classification
9. Result analysis

5.2.1.1 Image input

Input an image into the system for grading of the rice grain images into its respective classes. Input is taken in the form of .jpg or .jpeg file extensions.

```
global image
[file path]=uigetfile(['Dataset\.jpg']);
% reading one image
image=imread([path file]);
figure,imshow(image);
title('Input Image');
```

5.2.1.2 Grayscale Conversion

Grayscale is a range of monochromatic shades from white to black. This process removes all colored information, leaving only the luminance of each pixels.

```
global image
%rgb to gray conversion
image=rgb2gray(image);
figure,imshow(image);
title('Gray Image');
```

5.2.1.3 Resizing and Cropping

Resizing is an intrinsic change in the size of the image. Here the images are resized to 256px * 256px. This resizing affects not only the real size of the image but also potentially its weight. Cropping is just to cut the outer portions of an image to streamline the data.

```
im=imresize(image,[256,256]);
figure,imshow(im);
title('Resized Image');
```

5.2.1.4 Preprocessing

Data preprocessing includes, after resizing and cropping the images, they are rotated as well as mirrored to increase the number of images there are in the training data set.

Image sharpening is a powerful tool for emphasising texture and drawing viewer focus. It can improve image quality, even more than what is achieved through upgrading to a high-end camera lens.

```
ims=imsharpen(im, ‘Radius’,4,’Amount’,2);
figure,imshow(ims);
title(‘Sharpened Image’);
```

Gaussian filter is a type of linear filter. These are low-pass filters, based on the Gaussian probability distribution function as given below:

$$f(x)=e^{-x2/2\sigma2}$$

where ‘σ’ is the standard deviation. Gaussian filters have a blurring effect which looks very similar to that produced by neighbourhood averaging.

```
A = imread(im);
A_noise=imnoise(A,’gaussian’);
Sobel_A=edge(A,’sobel’);
imshow(Sobel_A)
title(‘Sobel Edge detection’);
```

5.2.1.5 HOG feature extraction

The **Histogram of Oriented Gradients** (HOG) is a feature descriptor used in computer vision and image processing for the purpose of object detection. The technique counts occurrences of gradient orientation in localized portions of an image.

```
global im fea_hog
fea_hog=extractHOGFeatures(im);
```

5.2.1.6 GLCM Feature Extraction

A gray level co-occurence matrix (GLCM) is a histogram of co-occurring grayscale values at a given offset over an image. To describe the texture of an image it is usual to extract features such as entropy, energy, contrast, correlation, etc. from several co-occurrence matrices computed for different offsets. In this case the entropy is defined as follows:

$$-\sum_{i=0}^{n-1}\sum_{j=0}^{n-1} p(i,j)\log_b p(i,j)$$

where $n$and $b$are again the number of gray levels and the base of the logarithm function, respectively, and $p(i,j)$ stands for the probability of two pixels separated by the specified offset having intensities $i$and j

```
global im features_glcm
[out] = GLCMFeatures(double(im));

Autocorrelation=(out.autoCorrelation);
clusterProminence=(out.clusterProminence);
clusterShade= (out.clusterShade);
contrast=(out.contrast);
correlation=(out.correlation);
differenceEntropy=(out.differenceEntropy);
differenceVariance=(out.differenceVariance);
dissimilarity=(out.dissimilarity);
energy=(out.energy);
entropy=(out.entropy);
homogeneity=(out.homogeneity);
informationMeasureOfCorrelation1=(out.informationMeasureOfCorre
lation1);
informationMeasureOfCorrelation2=
(out.informationMeasureOfCorrelation2);
inverseDifference=(out.inverseDifference);
maximumProbability=(out.maximumProbability);
sumAverage=(out.sumAverage);
sumEntropy= (out.sumEntropy);
```

```
sumOfSquaresVariance=
(out.sumOfSquaresVariance);
sumVariance=(out.sumVariance); features_glcm=[];
features_glcm=[mean(Autocorrelation) mean(clusterProminence)
mean(clusterShade) mean(contrast) mean(correlation)
mean(differenceEntropy) mean(differenceVariance)...
    mean(dissimilarity) mean(energy) mean(entropy)
mean(homogeneity) mean(informationMeasureOfCorrelation1)
mean(informationMeasureOfCorrelation2)
mean(inverseDifference) mean(maximumProbability)...
    mean(sumAverage) mean(sumEntropy)
mean(sumOfSquaresVariance) mean(sumVariance)];
```

5.2.1.7 Chalkiness feature extraction

Chalkiness is a major concern in rice because it is one of the key factors in determining quality. Evaluation of chalkiness is traditionally performed by human visual inspection, and there is no standard objective method to effectively classify chalky grains into different categories.

```
global im Percentage_Chalkiness feat_test features_glcm
fea_hog [pixelCount, grayLevels] = imhist(im); thresholdValue
= 90;
binaryImage = im > thresholdValue;
binaryImage = imfill(binaryImage, 'holes');
MaskedImage = im;
MaskedImage(~binaryImage) = 0;
blobMeasurements = regionprops(binaryImage, im,
'all'); [~, grayLevels_1] = imhist(MaskedImage);
thresholdValue_1 = 180;
binary_MaskedImage = MaskedImage > thresholdValue_1;
binary_MaskedImage = imfill(binary_MaskedImage, 'holes');
blobMeasurements_subblobs = regionprops(binary_MaskedImage,
im, 'all');
Sum_Subblobs_Perimeter =
sum([blobMeasurements_subblobs(1:end).Perimeter]);
Sum_Blobs_Perimeter   =   sum([blobMeasurements(1:end).Perimeter]);
Percentage_Chalkiness=(Sum_Subblobs_Perimeter/Sum_Blobs_Perimet
er)*1;
temp=[fea_hog Percentage_Chalkiness];
msgbox(num2str(Percentage_Chalkiness),'Percentage_Chalkiness');
feat_test=temp(:);
```

5.2.1.8 PNN Classification

Neural network is the best tool in recognition and discrimination between different sets of signals. To get best results using the neural network, it is necessary to choose a suitable architecture and learning algorithm. The best way to do that is to choose what is expected to be suitable according to our previous experience and then to expand or shrink the neural network size until a reasonable output is obtained.

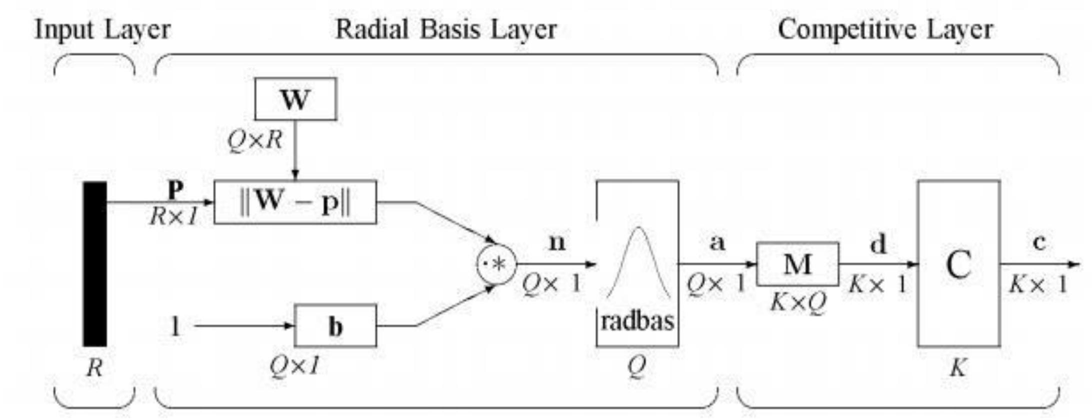

Figure 5.1 Architecture of a typical 3 Stage PNN

As shown in the above figure 5.1 the PNN has three layers: The Input Layer, The hidden layer and the output layer. The input image for classification is first given to the Input layer. Hidden Layer finds vector distances between input vector and row weight vectors in weight matrix. The obtained distances are ascended by Radial Basis Function. Output Layer finds the shortest distance among matrices, and thus finds the training pattern closest to the input pattern based on their distance. Here, supervised learning with non-knowledge-based classifier will be used for image classification. The neural network model PNN is used here to act as a classifier with radial basis function for network activation function. The training samples features with assigned target vectors are fed into created PNN model for supervised training to get network parameters such as node biases and weighting factors. Finally, test image features are simulating with trained network to make decision of brain stages like normal or abnormality (benign and malignant). The network classifies input vector into a specific class to a specific class that has the best probability to be correct. 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 illustrated as the dot product between two vectors. The dot product between P and the i^{th} row of **W** produces the i^{th} element of the distance vector Then, the bias vector b is joined with ||**W- p**|| by an element-by-element multiplication. The result is denoted as n = ||**W- p**||**.***b. The radial basis function in terms of n is given as radbas(n) = e^{-n^2} (Gaussian function). Each element of n is substituted and it produces corresponding element of a, the output values of Radial Basis Layer. The i^{th} element of a can be represented as the a_i = radbas(||$\mathbf{W_i}$ **- p**||**.***b) where W_i is the vector made of the i^{th} row of W and p. 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. The PNN is trained with reference features set and desired output is awaited. Here, target 1 for good, 2 for average and 3 for bad are taken as desired output. After the training, updated weighting factor and biases with other network parameters are stored to simulate with input features. At the classification stage, test image features are utilized to simulate with trained network model and the nearer stage is identified. Finally, it returns the classified value as 1, 2, or 3 based on that the decision will be taken as good, average or bad.

Advantages and disadvantages of PNN networks: It is usually much easier to train a PNN network and they are more accurate than a multilayer perceptron network, PNN networks are relatively insensitive to wild points, PNN networks generate accurate forecasted target probability scores, PNN networks are unhurried than multilayer perceptron networks at distinguishing new cases, PNN networks are hunting for more memory space to store the model.

5.3 SUMMARY

The contribution of this project is this, a special neural network architecture is proposed for the rice grain image classification task, which demonstrates superior performance over conventional feature extraction-based methods. Moreover, a relatively new method was introduced for the proposed setup, which also improves the algorithm’s performance. The results are encouraging compared to the reports of human grading, thus an automated evaluation will be undertaken in order to be able to integrate the presented algorithm into a tool to grade rice grains.

CHAPTER 6

SYSTEM TESTING

6.1 INTRODUCTION

Rice grain quality detection using PNN produces more accuracy than the other models which have been previously deployed. The proposed model uses a large set of rice grain images and a probabilistic neural network along with all pre-processing steps of the dataset in order to improve the model efficiency and accuracy. The below test cases, results and performance evaluation of the current model symbolizes the increase in efficiency and performance. Different graphical representations and tabular comparisons have been used to explain and discuss the results and performance of the model.

Software testing is a process of executing a program or application with the intent of finding the software bugs. It can also be stated as the process of validating and verifying that a software program or application or product.

The purpose of testing is to discover errors. Testing is the process of trying to discover every conceivable fault or weakness in a work product. It provides a way to check the functionality of components, sub-assemblies, assemblies and/or a finished product. It is the process of exercising software with the intent of ensuring that the software system meets its requirements and user expectations and does not fail in an unacceptable manner. Each test type addresses a specific testing requirement.

The different parts of testing are:

1. The process of **designing tests** early in the life cycle can help to prevent defects from being introduced in the code. Sometimes it is referred as "verifying the test basis via the test design"
2. The **Test Basis** includes documents such as the requirements and design specifications.

3. **Static Testing:** It can test and find defects without executing code. Static Testing is done during verification process. This testing includes reviewing of the documents (including source code) and static analysis. This is useful and cost effective way of testing. For example: reviewing, walkthrough, inspection, etc.

4. **Dynamic Testing:** In dynamic testing the software code is executed to demonstrate the result of running tests. It is done during validation process. For example: unit testing, integration testing, system testing, etc.

5. **Planning:** A Software Test Plan is a document describing the testing scope and activities. It is the basis for formally testing any software/product in a project.

6. **Preparation:** It involves the selection of the type of testing, the test conditions and the test cases.

7. **Evaluation:** It involves checking of the results and evaluating the software under test and the completion criteria, which helps to decide whether the testing has completed and whether the software product has passed the tests.

6.1.1 LEVELS OF TESTING

There are different levels during the process of testing. Levels of testing include different methodologies that can be used while conducting software testing. The main levels of software testing are:

1. Unit Testing
2. Integration Testing
3. System Testing

Unit Testing

Unit testing involves the design of test cases that validate that the internal program logic is functioning properly, and that program inputs produce valid outputs. All decision branches and internal code flow should be validated. It is the testing of individual software units of the application. It is done after the completion of an individual unit before integration. This is a structural testing, that relies on knowledge of its construction and is invasive. Unit tests perform basic tests at component level and test a specific business process, application, and/or system configuration. Unit tests ensure that each unique path of a business process performs accurately to the documented specifications and contains clearly defined inputs and expected results. The units of the system such as the resizing, cropping of images were tested to check if they were actually functioning as required.

Integration Testing

Integration tests are designed to test integrated software components to determine if they actually run as one program. Testing is event driven and is more concerned with the basic outcome of screens or fields. Integration tests demonstrate that the components were individually satisfactory. Integration testing specifically aims at exposing the problems that arise from the combination of components. Once all the units of the system were integrated, it was tested to check if each module was functioning correctly one stage after another.

System Testing

System testing ensures that the entire integrated software system meets requirements. It tests a configuration to ensure known and predictable results. An example of system testing is the configuration oriented system integration test. System testing is based on process descriptions and flows, emphasizing pre-driven process links and integration points. The entire application was tested on different labeled inputs. It was found that the model was able to predict the correct diagnosis in most cases.

System testing is important because of the following reasons:

1. System testing is the first step in the Software Development Life Cycle, where the application is tested as a whole.
2. The application is tested thoroughly to verify that it meets the functional and technical specifications.
3. The application is tested in an environment that is very close to the production environment where the application will be deployed.
4. System testing enables us to test, verify, and validate both the business requirements as well as the application architecture.

6.2 TEST CASES

Test case 1: Checking for a class of rice grain quality using the input image

Table 6.1 Test case of rice sample 1

Test Case ID:-	**1**
Name of the test:-	Checking for quality of sample 1
Sample input:-	 Rice Grain image
Expected output:-	Class 2: Average quality rice grain
Actual output:-	Class 2: Average quality rice grain
Remarks:-	Pass

Test case 2: Checking for a class of rice grain quality using the input image

Table 6.2 Test case of rice sample 2

Test Case ID:-	**2**
Name of the test:-	Checking for quality of grain sample 2
Sample input:-	 Rice Grain image
Expected output:-	Class 3: Bad quality rice grain
Actual output: -	Class 3: Bad quality rice grain
Remarks: -	Pass

Test case 3: Checking for a class of rice grain quality using the input image

Table 6.3 Test case of rice sample 3

Test Case ID:-	**3**
Name of the test:-	Checking for quality of grain sample 3
Sample input:-	 Rice grain image
Expected output:-	Class 3: Good quality rice grain
Actual output:-	Class 3: Good quality rice grain
Remarks:-	Pass

Test case 4: Checking for a class of rice grain quality using the input image

Table 6.4 Test case of rice sample 4

Test Case ID:-	**4**
Name of the test:-	Checking for quality of grain sample 4
Sample input:-	 Rice grain image
Expected output:-	Class 3: Bad quality rice grain
Actual output:-	Class 3: Bad quality rice grain
Remarks:-	Pass

Test case 5: Checking for a class of rice grain quality using the input image

Table 6.5 Test case of rice sample 5

Test Case ID:-	5
Name of the test:-	Checking for quality of grain sample 5
Sample input:-	 Rice grain image
Expected output:-	Class 3: Good quality rice grain
Actual output:-	Class 3: Good quality rice grain
Remarks:-	Pass

6.3 RESULTS

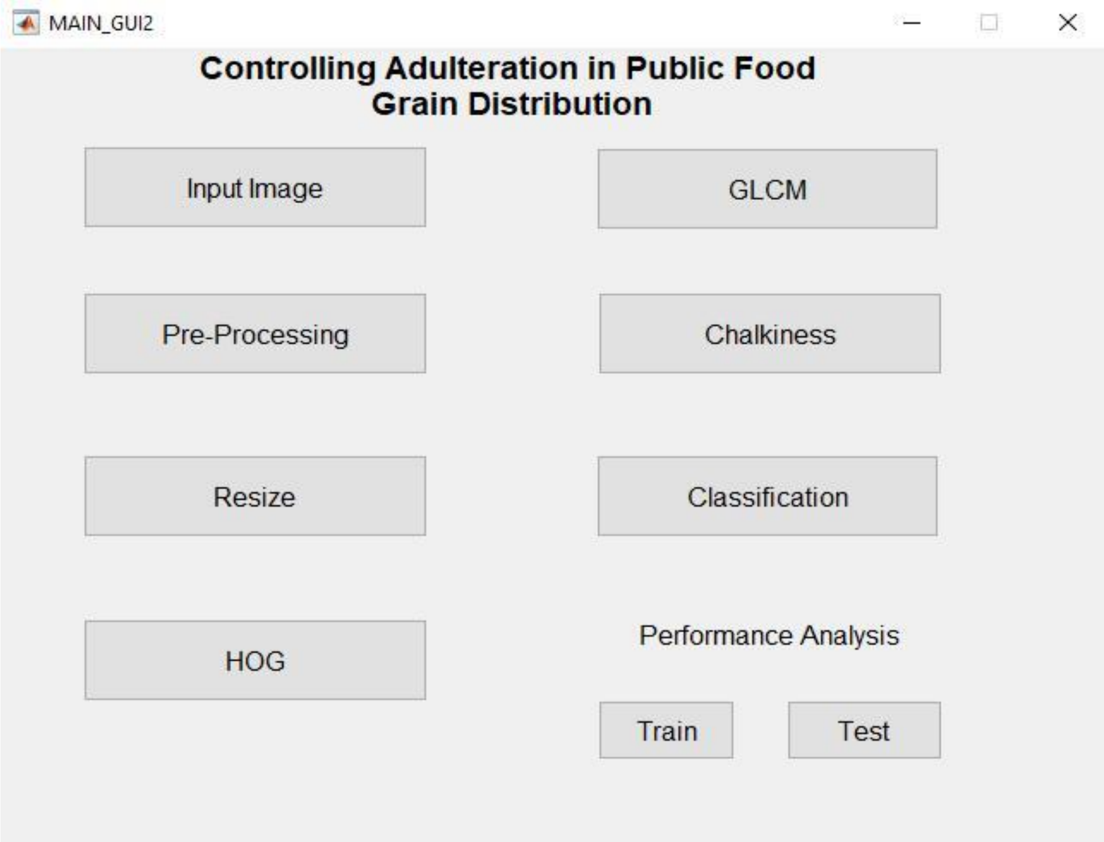

Figure 6.1 The UI/UX of the model

The figure 6.1 shows the finished UI/UX part of the project. The grain images are uploaded and when relevant buttons are pressed the output is generated by the backend trained PNN model which detects the class of rice grain.

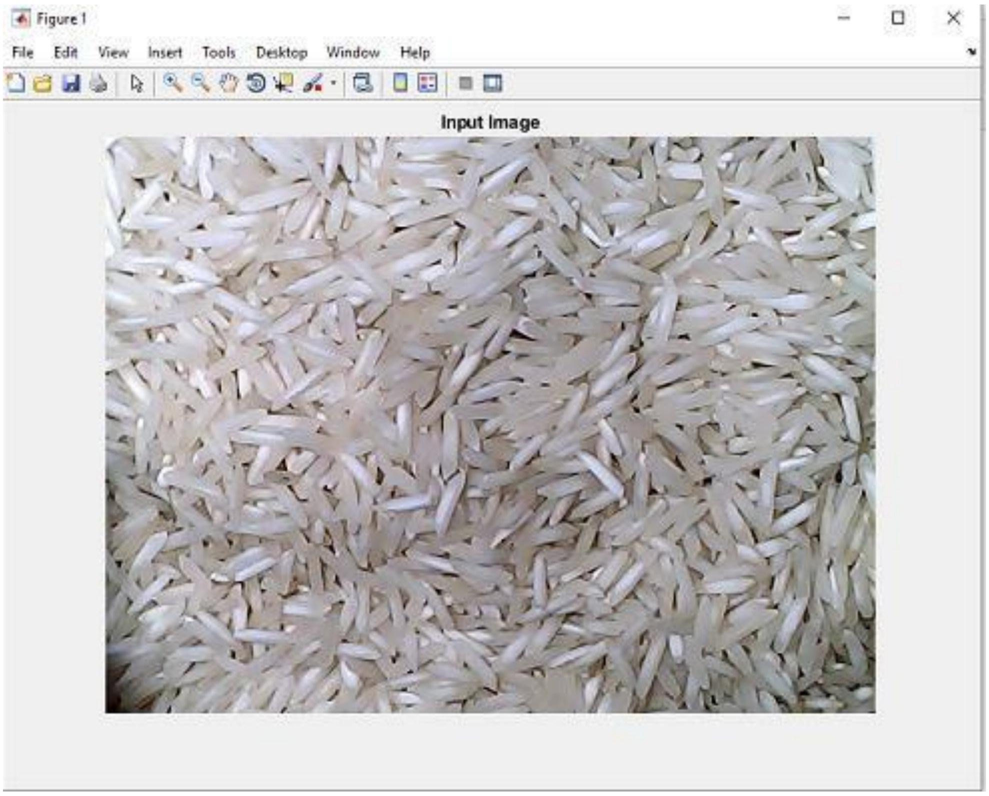

Figure 6.2 Input sample image

The above Figure 6.2 shows the sample input image given to the network. This image is selected from any of the images present in the Test images folder.

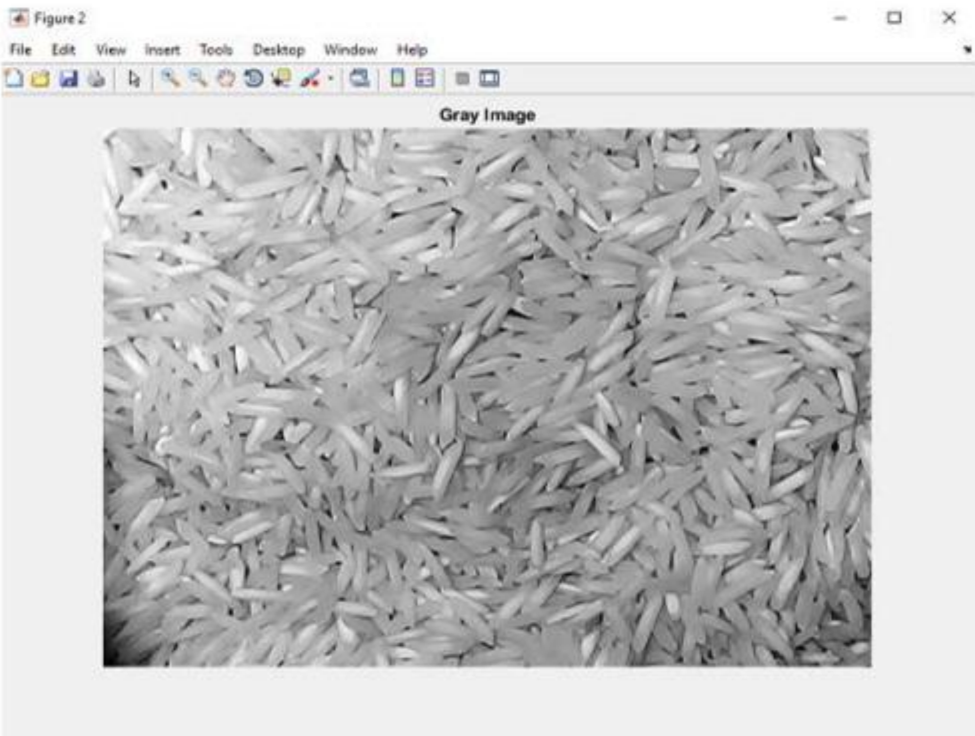

Figure 6.3 Grayscale Image

The above Figure 6.3 shows the grayscale of the original input image. Grayscale images are devoid of any colors except white and black.

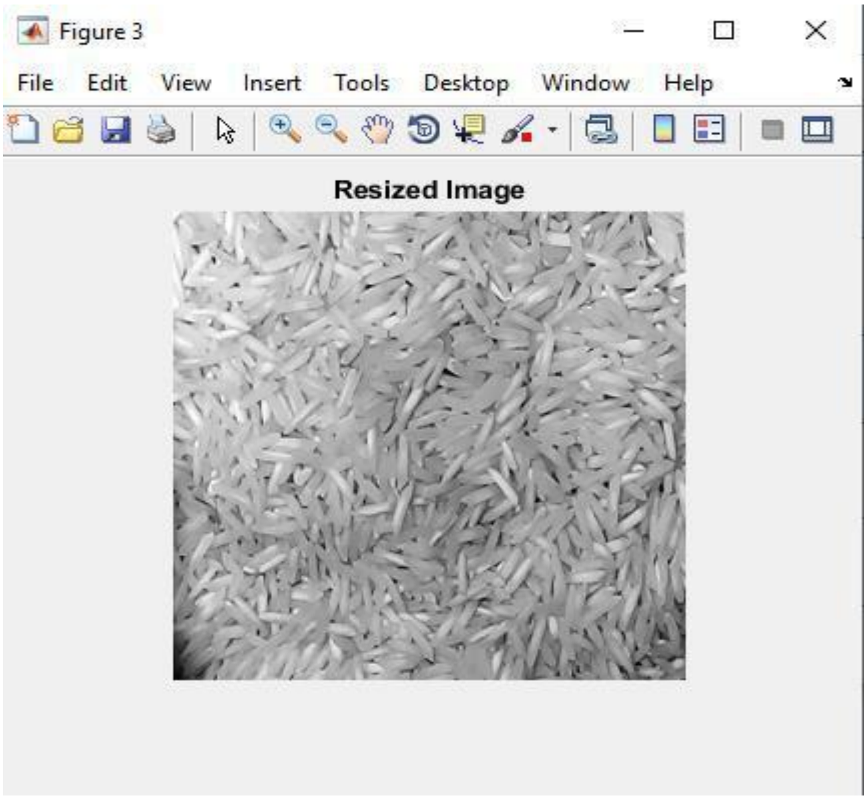

Figure 6.4 Resized Image

Figure 6.4 depicts the resized version of the original input image which makes it easy for processing

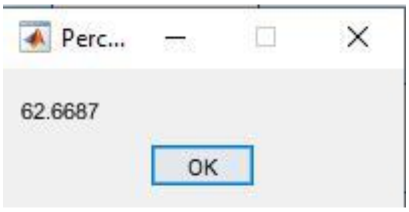

Figure 6.5 Chalkiness

The percentage of chalkiness of the input image is shown in the dialogue box as shown in Figure 6.5

Figure 6.6 Classification Result

The final classification of the category of image is shown in the dialogue box as depicted in Figure 6.6. The classification is done using decision tree, which is a widely used machine learning concept.

6.4 PERFORMANCE EVALUATION

Precision and recall is a simple yet useful way to measure the quality of predictions. Assume creating a classification scenario and see how the precision and recall numbers and the F-score can be applied- to determine how effective the algorithms are.

Table 6.6 Recall and Precision of the model

Class	Recall	Precision
1: Good	96.1905	95.2830
2: Average	96.8340	96.8372
3: Bad	97.1963	98.1132

The given table 6.6 specifies how the model is responding to the given training and testing data along with the values of recall and precision, the accuracy can also be found easily.

Table 6.7 Accuracy average of the model

Metric	Value
Accuracy (Train)	100
Accuracy (Test)	97.8903
Precision	96.8331
Recall	97.3269

The above table 6.7 specifies the average accuracy by considering the overall performance of the proposed model, recall and precision values obtained while testing the model with the given dataset.

6.6 SUMMARY

Software testing is a process of executing a program or application with the intent of finding the software bugs. The purpose of testing is to discover errors.

The main levels of software testing are: Unit Testing, Integration Testing and System Testing. Unit testing involves the design of test cases that validate that the internal program logic is functioning properly, and that program inputs produce valid outputs. Integration tests are designed to test integrated software components to determine if they actually run as one program. System testing ensures that the entire integrated software system meets requirements. The various test cases of the PNN model used for rice grain classification are designed and are used to test the proposed model. It is found that the test cases are passed without any problem based on the results that are obtained, which is a perfect classification of the grain images of different datasets as 3 different classes based on the class of the rice grain employed. The performance is evaluated using the bar graphs, which shows the accuracy and efficiency measure of the proposed model. The overall performance is determined using the mean efficiency of all testing data by the model. Hence, the Improved grading of rice grains by using PNN provides a better performance than the previous versions of manual rice grading and classification.

CONCLUSION

An application of grading and classification of the rice grains and providing the actual seed quality for the given set of rice grains has been implemented. Hence the proposed objective was implemented on three different types of rice grains namely grade 1, grade 2 and grade 3 rice grains. Probabilistic neural networks was implemented to achieve high classification accuracies. The experimental results indicate the proposed approach can classify with a little computational effort. By this method, the quality of rice grains can be identified at the initial stage itself, before distribution to the general public so that they can receive highest quality rice grains, without any human intervention. Features considered are Histogram of Oriented Gradients, Gray level Co-occurrence matrix and Chalkiness. High levels of classification accuracies are realized in this process.

TABLE III. ACCURACY COMPARISON

Scheme	Accuracy
Deep CNN	95.5
LVQ neural network	70.3
BPNN	80.5
ANN	93
PNN	87.5
Region Proposals based CNN	67.25
Spatio-Spectral Deep CNN	93.27
Our Approach	97.89

The comparison reveals that the proposed algorithm outperforms other methods. Routinely, PNN is known to outpace other algorithms in a machine vision application. The proposed system is completely automated and proves to be of a great use to the agriculture-based industries. We have found out that the results for rice grain grading using probabilistic neural networks to be highly lucrative and obtained high levels of accuracy in comparison to other related works as shown in Table III.

In the future, locally grown rice grains in large quantities can be used to train the model. Further, high resolution cameras can be used to capture images so that the classification accuracies can be further increased. Cutting edge technologies could be used to classify the rice grains in the very inception.

REFERENCES

[1] Qing Yao, Jianhua Chen, Zexin Guan, Chengxiao Sun, Zhiwei Zhu "Inspection of rice appearance quality using machine vision", Global Congress on Intelligent Systems, 978-0-7695-3571-5/09, IEEE 2014.

[2] Yong Wu and Yi Pan "Cereal Grain Size measurement based on Image Processing Technology", International Conference on Intelligent Control and Information Processing, August 13-15, 2015 - Dalian, China.

[3] Megha R. Siddagangappa, A. H. Kulkarni "Classification and Quality Analysis of Food Grains," IOSR Journal of Computer Engineering

[4] Deepika Sharma et al "Grain Quality Detection by using Image Processing for public distribution", International Conference on Intelligent Computing and Control Systems 978-1-5386-2745-7/17/ 2017

[5] Shraddha N. Shahane et al, "Grain Quality Assessment for Rationing System", Online International Conference on Green Engineering

[6] Harpreet Singh et al, "Image Processing Techniques for Analysing Food Grains", International Conference on Computing Methodologies Communication (ICCMC) 978-1-4244-7164-5 ©2019 IEEE

[7] Zahida Praveen et al, "Assessment of Quality of Rice Grain using Optical and Image Processing Technique", International Conference on Computing Methodologies Communication (ICCMC) 978-1-4244-7164- ©2019 IEEE