A
Project Report
on

Designing an Effective Mechanism For Seed Analysis

Submitted to

Sant Gadge Baba Amravati University, Amravati

Submitted in partial fulfilment of

the requirements for the Degree of

Bachelor of Engineering in

Computer Science and Engineering

Submitted by

Gopal Gajanan Solanki
(PRN: 203120066)

Rushikesh Arun Gadhave
(PRN: 203120383)

Abhijeet Rambhau Gadlinge
(PRN: 203120074)

Under the Guidance of
Dr. N.M. Kandoi
Professor, CSE Dept.

Department of Computer Science and Engineering
Shri Sant Gajanan Maharaj College of Engineering,
Shegaon – 444 203 (M.S.)

Session 2023-2024

SHRI SANT GAJANAN MAHARAJ COLLEGE OF ENGINEERING,
SHEGAON – 444 203 (M.S.)
DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

CERTIFICATE

This is to certify that **Mr. Gopal Gajanan Solanki , Mr. Rushikesh Arun Gadhave** and **Mr. Abhijeet Rambhau Gadlinge** students of final year Bachelor of Engineering in the academic year 2023-24 of Computer Science and Engineering Department of this institute have completed the project work entitled **"Designing an Effective Mechanism For Seed Analysis"** and submitted a satisfactory work in this report. Hence recommended for the partial fulfillment of degree of Bachelor of Engineering in Computer Science and Engineering.

Dr. N.M. Kandoi
Project Guide

Dr. J.M. Patil
Head of Department

Dr. S. B. Somani
Principal
SSGMCE, Shegaon

SHRI SANT GAJANAN MAHARAJ COLLEGE OF ENGINEERING,

SHEGAON – 444 203 (M.S.)

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

CERTIFICATE

This is to certify that **Mr. Gopal Gajanan Solanki, Mr. Rushikesh Arun Gadhave** and **Mr. Abhijeet Rambhau Gadlinge** students of final year Bachelor of Engineering in the academic year 2023-24 of Computer Science and Engineering Department of this institute have completed the project work entitled **"Designing an Effective Mechanism For Seed Analysis"** and submitted a satisfactory work in this report. Hence recommended for the partial fulfillment of degree of Bachelor of Engineering in Computer Science and Engineering.

Internal Examiner	**External Examiner** 10/05/2024
DTHmKande	Dr. A.P. Kankale
Name and Signature	**Name and Signature**
Date: 10/5/24	**Date:** 10/05/24

Acknowledgement

It is our utmost duty and desire to express gratitude to various people who have rendered valuable guidance during our project work. We would have never succeeded in completing our task without the cooperation, encouragement and help provided to us by then. There are a number of people who deserve recognition for their unwavering support and guidance throughout this report.

We are highly indebted to our guide **Dr. N. M. Kandoi** for his guidance and constant supervision as well as for providing necessary information from time to time. We would like to take this opportunity to express our sincere thanks, for his esteemed guidance and encouragement. His suggestions broaden our vision and guided us to succeed in this work.

We are sincerely thankful to **Dr. J. M. Patil** (HOD, CSE Department, SSGMCE, Shegaon), and to **Dr. S. B. Somani** (Principal, SSGMCE, Shegaon) who always has been kind to extend their support and help whenever needed.

We would like to thank all teaching and non-teaching staff of the department for their cooperation and help. Our deepest thank to our parents and friends who have consistently assisted us towards successful completion of our work.

– Projectees

Gopal Gajanan Solanki

Rushikesh Arun Gadhave

Abhijeet Rambhau Gadlinge

ABSTRACT

The cornerstone of our economy has traditionally rested upon agriculture, with many tasks still reliant on traditional methods devoid of modern technological interventions. Presently, human cognition plays a pivotal role in prognosticating the quality of agricultural progeny. However, due to the absence of a robust validation mechanism, extant prognostications regarding seed quality are deemed inadequate. To address this limitation, we endeavored to devise a prognostic model leveraging machine learning algorithms.

This model aims to forecast seed quality, thereby fostering enhanced crop yield and superior harvest outcomes. To achieve precise seed classification, we employed convolutional neural networks (CNNs) trained on a comprehensive seed dataset. By assimilating data conducive to predictive analytics, our model discerns whether seeds exhibit premium, standard, or regular quality attributes. Through the integration of testing, training, and validation data, our approach endeavours to optimize the accuracy of seed quality predictions, as evaluated through the efficacy of the CNN model's training and prediction accuracy of the algorithm.

Table Of Contents	Page Number

List of Abbreviations and Symbol

Abbreviations	Particulars
CNN	Convolutional Nural Network
API	Application Programing Interface

List of Figures

Chapter 1

INTRODUCTION

1. INTRODUCTION

1.1 OVERVIEW

This report presents the results of a project on seed analysis mechanism using machine learning. The purpose of this project was to develop a computer vision model that can accurately identify seed can be good or defective using convolutional neural networks and image processing techniques.

The project was motivated by the increasing demand for automated and accurate seed detection methods in agriculture, which can help farmers to sowing seed in to soil. The objectives of the project were to collect a dataset of seed images, train and evaluate machine learning models using different architectures and techniques, and analyze the performance and limitations of the models.

This report describes the methodology, results, and conclusions of the project, as well as the challenges and opportunities for further research. The project was conducted over several weeks and involved collaboration with domain experts, data scientists, and software engineers.

I would like to acknowledge the support and guidance provided by my project supervisor, as well as the resources and infrastructure provided by the organization. I would also like to thank the participants who contributed their crop seed images and the colleagues who provided feedback and assistance during the project.

1.2 BACKGROUND AND SIGNIFICANCE

1.2.1 Background:

The background of seed analysis lies in the recognition of seeds as the primary input in agriculture, directly influencing crop yield, uniformity, and overall performance. Understanding the significance of seed analysis requires acknowledging the multifaceted challenges and implications associated with seed quality assurance.

Seed analysis directly influences crop productivity and yield. By assessing parameters such as viability, germination rate, and seed health, seed analysis enables farmers to select high-quality seeds capable of producing healthy plants with robust growth potential. This ensures optimal crop establishment, uniformity, and ultimately, higher yield.

1.2.2 Significance:

Seed analysis is essential for regulatory compliance and consumer protection. Many countries have stringent standards and labelling requirements governing seed quality to safeguard farmers and consumers. Compliance with these regulations ensures that seeds meet specified quality standards, thereby enhancing consumer confidence, preventing fraudulent practices, and promoting fair trade practices within the seed industry.

Seed analysis plays a crucial role in research and development efforts aimed at improving crop varieties, breeding techniques, and agricultural practices. By providing valuable insights into seed characteristics and performance, seed analysis informs the development of new cultivars with desirable traits such as disease resistance, drought tolerance, and nutritional quality. It also facilitates the evaluation of new technologies and management practices to enhance crop productivity and sustainability.

The significance of seed analysis cannot be overstated. It is essential for ensuring crop productivity, genetic improvement, regulatory compliance, and research advancement in agriculture. By underpinning the quality and integrity of seeds, seed analysis contributes to agricultural sustainability, food security, and economic development worldwide.

1.3 AIMS OF RESEARCH WORK STUDY

Defective seed can cause significant economic losses in agriculture, and timely and accurate detection is critical for effective crop production. However, traditional methods for seed analysis are often time-consuming and require specialized expertise, making it difficult for farmers to identify and manage seed analysis in a timely manner.

In this project, we aimed to develop a machine learning-based system for crop seed analysis, which can automate the diagnosis process and provide farmers with a fast and reliable method for sowing the seed in to the soil. The objectives of the project were to collect a dataset of crop seed images, train and evaluate machine learning models using different architectures and techniques, and analyze the performance and limitations of the models.

By addressing this problem, our project aims to contribute to the development of sustainable and efficient crop production methods, and provide farmers with valuable insights and decision-making tools for sowing the seed.

1.4 OBJECTIVES AND SCOPE

1.4.1 Objectives:

1. To collect a dataset of crop seed images representing bad seed and mature seed.
2. To implement and evaluate several machine learning models for crop seed detection, including Convolutional Neural Networks (CNNs) and transfer learning-based approaches.
3. To analyze the performance of the machine learning models using metrics such as accuracy, precision, recall, and F1 score.
4. To compare the performance of the different models and identify the best performing model for seed analysis.
5. To visualize and interpret the model predictions to gain insights into the features and patterns that distinguish bad seed and mature seed.
6. To discuss the limitations and future directions for the proposed approach, including potential applications in real-world scenarios.

By achieving these objectives, our project aimed to develop a machine learning based system for crop seed analysis that can provide farmers with a fast and reliable method for sowing the seed, and contribute to the development of sustainable and efficient food production methods.

1.4.2 Scope:

This project focuses on the development of a machine learning-based system for crop seed analysis using a dataset of seed images representing bad seeds and mature seeds. The project involves the implementation and evaluation of several deep learning models, including CNNs and transfer learning-based approaches. The project aims to provide farmers with a fast and reliable method for seed sowing, and contribute to the development of sustainable and efficient food production methods.

1.5 ORGANIZATION OF THE PROJECT

The project is organized as follows:

1. Chapter 1 Gives introduction to the project
2. Chapter 2 Provides a literature survey of the project.
3. Chapter 3 Explains methodologies required to complete the project.
4. Chapter 4 Provides computer simulation of the project.
5. Chapter 5 Provides the experimental investigation of the project.
6. Chapter 6 Provides results and discussion of the project.
7. Chapter 7 Provides conclusions, contributions, scope for future work of the project.

CHAPTER 02

LITERATURE REVIEW

2. LITERATURE REVIEW

Raghavendra Srinivasaiah, Meenakshi, Ravikumar Hodikehosahally Channegowda, Santosh Kumar Jankatti. "Analysis and prediction of seed quality using machine learning"(2023). The authors suggested technique made use of CNN architectures to distinguish between different types of seeds with accuracy and to recognize individual seeds with extreme accuracy. For the training CNN model, the models reach above 97% accuracy, while for the testing dataset, they obtain a prediction accuracy of 64%. In comparison to conventional and manual techniques, the model can assist to accelerate the seed health prediction system with reduced error rates and improved performance for bigger experimental data[1]. They have developed an algorithm that is ideal for accurately predicting seed quality.

Francival Cardoso Felix , Kyvia Pontes Teixeira das Chagas. "Image analysis of seeds and machine learning as a tool for distinguishing populations: Applied to an invasive tree species"(2023). Image analysis was efficient in detecting biometric differences in L. leucocephala seeds from distinct locations. Therefore, this method is promising for discriminating forest tree populations associated with machine learning, supporting management activities, and studying population genetic divergence. Additionally, digital imaging analysis contributes to understanding genotype–environment interactions and, consequently, to identifying the ability of an invasive species to spread in a new area, making it possible to track and monitor the flow of seeds between populations and other sites[2]. This study evaluated the seed traits of spatially dispersed populations using digital images of L. leucocephala and analyzed their implications for genetic studies.

Francival Cardoso Felix, Dagma Kratz, Richardson RibeiroI, Antonio Carlos Nogueira. "Characterization and differentiation of forest species by seed image analysis: a new methodological approach"(2023). Using seed biometric analysis by image processing has direct implications for silvicultural, genetic, and ecological studies. The method promotes economic gains since it optimizes the seed evaluation time by the analyst. Seed image analysis proved advantageous because it enabled to measure more size, shape, and color characteristics than can be measured by manual approaches using a digital caliper or ruler, which is only possible by image processing[3]. Seed image analysis using the method proposed in the present study helps to better characterize and differentiate forest species.

Ewa Ropelewska, Jan Piecko. “Discrimination of tomato seeds belonging to diferent cultivars using machine learning”(2021). This study developing the discriminant models for distinguishing the tomato seeds based on texture parameters of the outer surface of seeds calculated from the images (scans) converted to individual color channels R, G, B, L, a, b, X, Y, Z. The tomatoes belonging to cultivars ‘Green Zebra’, ‘Ożarowski’, ‘Pineapple’, Sacher F1 and Sandoline F1 were used in the experiments. The tomatoes were purchased from a local manufacturer. The seeds were manually prepared for the image acquisition. The individual tomato fruits were cut into quarters. Then, the seed chambers were emptied. The extracted seeds were covered with a protective tissue (mucilaginous gel) which was removed to obtain clean seeds. During the process of seed extraction, the seeds were rinsed in a sieve under tap water. In the next step, the mucilaginous gel was removed mechanically by sponge on absorption paper[4].

Deborah Bambil, Hemerson Pistori, Francielli Bao. “Plant species identifcation using color learning resources, shape, texture, through machine learning and artifcial neural networks”(2020). The authors demonstrated the efectiveness of computer vision in the identifcation of Cerrado plant species using leaf samples, based on extracted features of color, shape and texture. Promising results were obtained for the SVM, deep learning, and random forest algorithms; and independent of the image capture device. The results of this model obtained with the machine learning algorithms for identifcation of the tree and shrub species from the Cerrado phytophysiognomies revealed a high accuracy of algorithms (from 93.5 to 98.4%—Supplementary Table A2), with the exception of the AdaBoost algorithm which achieved only low precision (6.5%)[5].

Guoyang Zhao , Longzhe Quan , Hailong Li , Huaiqu Feng , Songwei Li , Shuhan Zhang , Ruiqi Liu. “Real-time recognition system of soybean seed full-surface defects based on deep learning”(2021). In this study, the visual system is combined with an alternate circumrotating mechanism to obtain the surface information of soybean seeds and perform defect recognition. According to model, the surface exposure rate was 98.52% for normal soybeans, 98.26% for cracked soybeans, 95.22% for broken soybeans, 96.44% for insect-bitten soybeans, 97.54% for diseased soybeans, and 98.10% for mildewed soybeans. According to the analysis of the experimental data, broken soybeans and insect-bitten soybeans do not rotate smoothly in the mechanism due to the incomplete ellipsoid, resulting in a low exposure rate[6]. This study proposes a real-time recognition system of soybean seed full-surface defects based on deep learning to achieve automatic recognition of soybean seeds in selection work.

Tongyun Luo, Jianye Zhao, Yujuan Gu, Shuo Zhang, Xi Qiao, Wen Tian, Yangchun Han. "Classification of weed seeds based on visual images and deep learning"(2021). In this study of experiment, more than 6 000 color raw images of seeds are used to test and validate the effectiveness of the proposed method. The weed seed identification methods in this model only need to load the weed seeds, the entire identification process does not require any manual intervention, and the final output is the seed species and number. Furthermore, when the weed seed dataset is expanded, the model can simply be retrained and the old CNN model can be replaced to complete the upgrade[7]. This study proposed a detection system for the intelligent classification of multiple species of weed seeds and provided a method that is more likely to have large-scale applications with both commercial and technological implications.

CHAPTER 03

MATERIALS AND METHODOLOGY

3. MATERIALS AND METHODOLOGY

MATERIALS :

- We curated a comprehensive dataset comprising images of plant leaves exhibiting both common diseases and healthy states. These images were sourced from diverse channels, including online repositories and field surveys.
- Our deep learning models were trained and assessed on hardware equipped with an i5 processor and ample memory and storage capacity. Software-wise, we employed Python along with deep learning libraries such as TensorFlow and Keras for model implementation and evaluation.
- Data Preprocessing:
  To ensure uniformity and enhance quality, we subjected the plant leaf image dataset to preprocessing steps like resizing, normalization, and augmentation (e.g., rotation, flipping, shearing). Subsequently, we partitioned the preprocessed data into training, validation, and testing sets in a ratio of 54:18:8, respectively.
- Model Implementation and Evaluation:
  Various deep learning architectures, encompassing Convolutional Neural Networks (CNNs) and transfer learning-based methodologies (e.g., VGG16, InceptionV3, ResNet50), were deployed for plant leaf disease detection. Training was performed on the training set, while hyperparameter optimization techniques like grid search and random search were employed using the validation set. Model evaluation encompassed metrics such as accuracy, precision, recall, and F1 score, enabling robust comparison to determine the most effective model.
- Results Analysis:
  We meticulously analyzed and visualized the models' performance utilizing tools such as confusion matrices, ROC curves, and feature maps. Additionally, we delved into the limitations and potential applications of our approach, including discussions on extending to multi-class classification and real-time disease surveillance.

  The potential applications and limitations of the proposed were method discussed, including potential extensions to multi-class classification and real-time disease surveillance.

Detailing the materials and methodology employed in your project is crucial for ensuring that others can replicate your work accurately and validate its findings. It also lays a solid foundation for further research to build upon.

In a project focusing on seed analysis, the materials section assumes paramount importance, with the seed image dataset standing out as the primary material. It's essential to offer insights into the dataset's origins and any preprocessing methodologies employed to refine the data for modelling purposes. Additionally, detailing the hardware and software utilized in the project adds depth. This may encompass specifics such as the type of GPU utilized for deep learning model training, along with the programming language and libraries harnessed for model implementation.

METHODOLOGY:

The methodology section serves as the backbone of your project, offering a detailed account of the methods and procedures employed. Here, it's imperative to outline the process of collecting the seed image dataset and any preprocessing steps taken to refine the data for modeling purposes. Elaborating on the deep learning models utilized and the evaluation metrics employed to gauge their performance adds depth to your methodology.

Furthermore, shedding light on the optimization of hyperparameters, such as learning rate and batch size, and the validation techniques adopted to combat overfitting, enhances the credibility of your approach. Analyzing the model results involves techniques like confusion matrices, ROC curves, and feature maps, providing invaluable insights into the model's performance.

Moreover, discussing the limitations of your methodology and potential avenues for application, such as extending the model to handle multi-class classification or real-time disease surveillance, underscores the significance and versatility of your work. In essence, the materials and methodology section should serve as a comprehensive guide, enabling others to grasp your project's intricacies and potentially replicate or build upon your findings.

3.1 Working of Convolutional Neural Network

Convolutional Neural Networks (CNNs), also known as ConvNets, draw inspiration from the intricate organization of the visual cortex in animals. Within the visual cortex, specialized cells respond to specific regions of the visual field, detecting features like edges of varying orientations.

In CNNs, this concept is translated into a network architecture where each layer processes information through local receptive fields, mirroring the spatial correlations found in input data. These receptive fields, akin to small regions in the visual cortex, focus on subsets of neurons, enabling them to extract features from the input data within their defined boundaries.

As data is processed through the network, hidden neurons analyze the information within these receptive fields, oblivious to changes occurring beyond their localized scope. This localized processing allows CNNs to effectively capture and learn intricate patterns and features present in the input data, making them particularly adept at tasks like image recognition and classification.

3.1.1 Working Of CNN

Typically, a Convolutional Neural Network (CNN) comprises three main layers: the convolutional layer, the pooling layer, and the fully connected layer. Let's delve into each layer's functionality using a classifier example that distinguishes between images of X and O shapes. This illustration will help us grasp the intricacies of all four layers.

Convolutional Neural Networks have the following layers:

- Convolutional
- ReLU Layer
- Pooling
- Fully Connected Layer

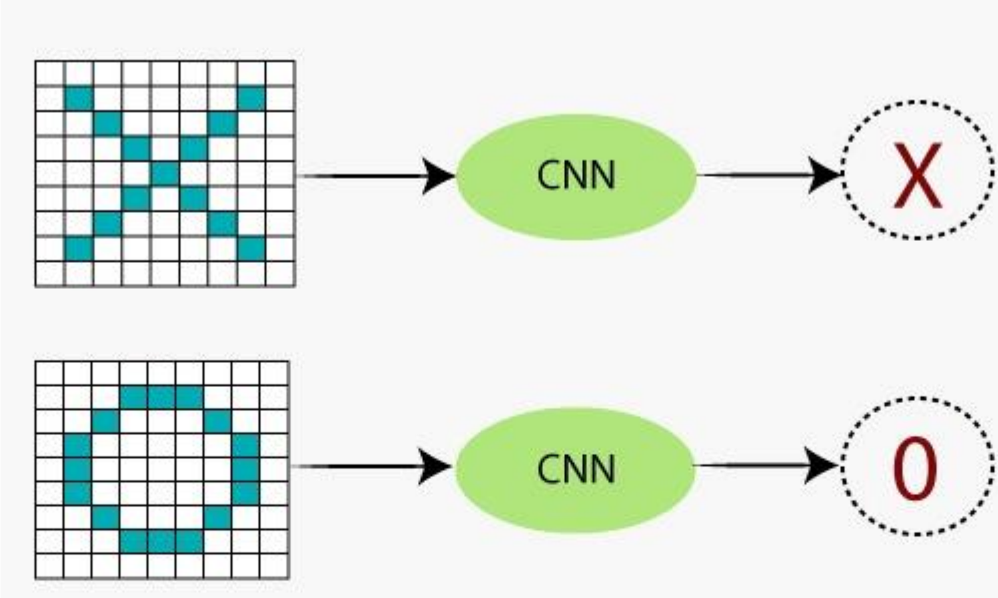


Fig 3.1.1.1: Working of CNN phase-1

In some instances, identifying X and O shapes becomes more challenging due to variations caused by deformations in the images. These deformations can lead to multiple representations of X and O shapes, especially when considering variations like rotation or distortion. Recognizing such diverse representations poses a significant challenge for computers. But the goal is that if the input signal looks like previous images it has seen before, the "image" reference signal will be convolved with the input signal. The resulting output signal is then passed on to the next layer.

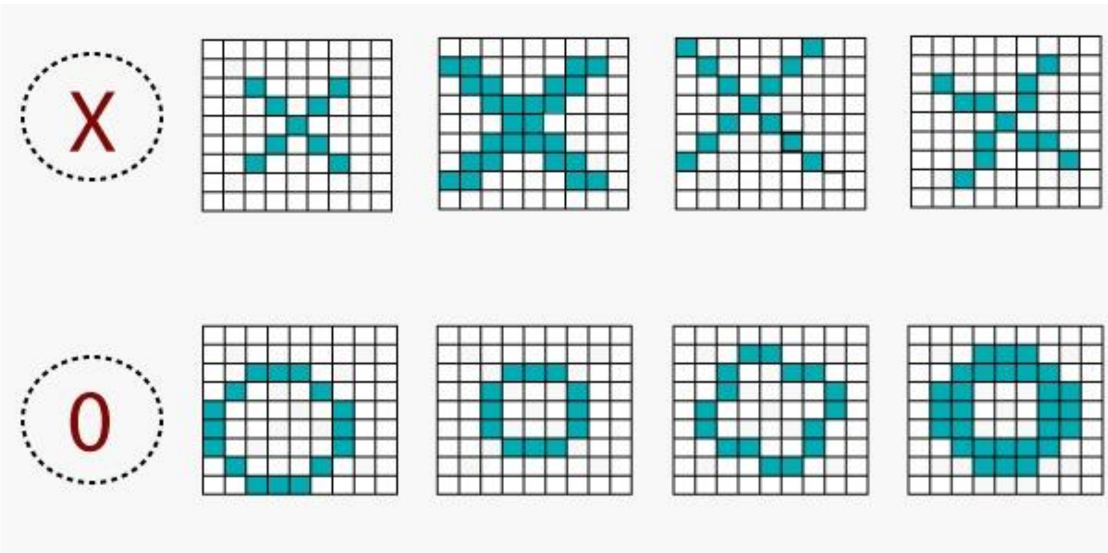


Fig 3.1.1.1: Working of CNN phase-2

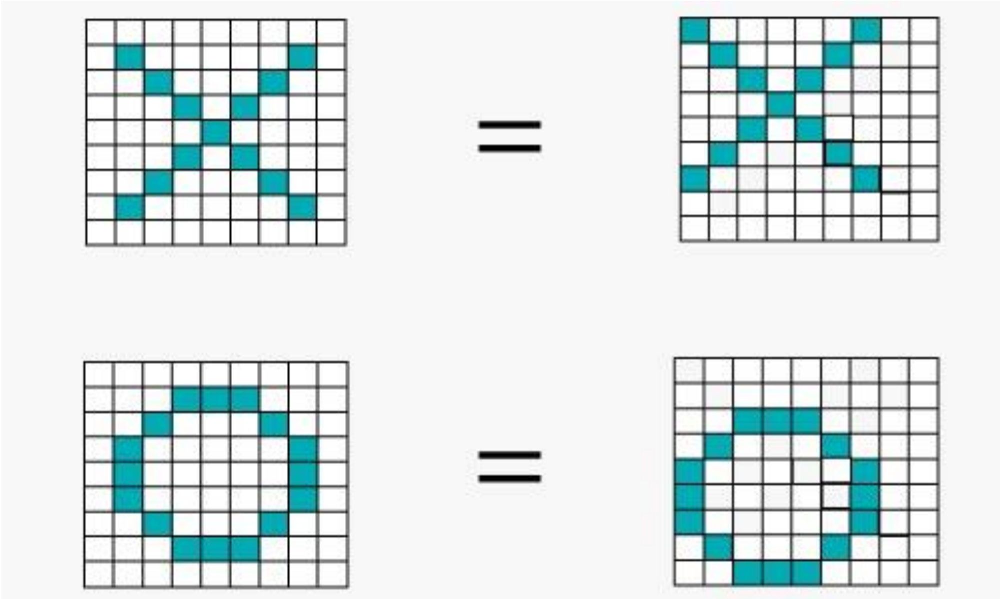


Fig 3.1.1.1: Working of CNN phase-3

A computer understands an image by using numbers at each pixel.

In our example, we have considered that a white pixel will have -1 value and blue pixel will have value 1. This is the way we have differentiated the pixels in a primary binary classification.

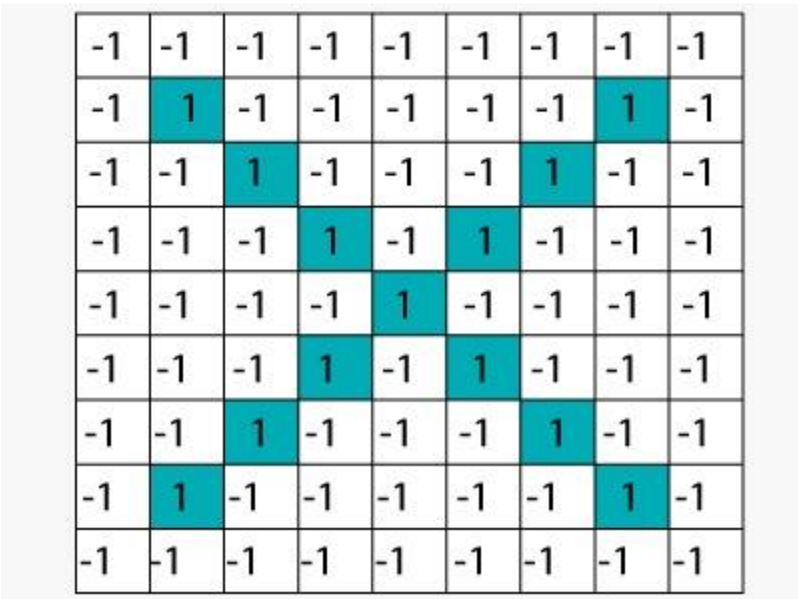


Fig 3.1.1.1: Working of CNN phase-4

When employing standard techniques to compare two images—one representing the proper image of X and the other a distorted version—the challenge arises when the computer struggles to classify the distorted image of X accurately. This difficulty stems from the comparison being made against the proper representation of X. Consequently, when we sum the pixel values of both images, the resulting aggregate lacks distinctiveness, leaving the computer unable to discern whether the combined representation depicts an X or not.

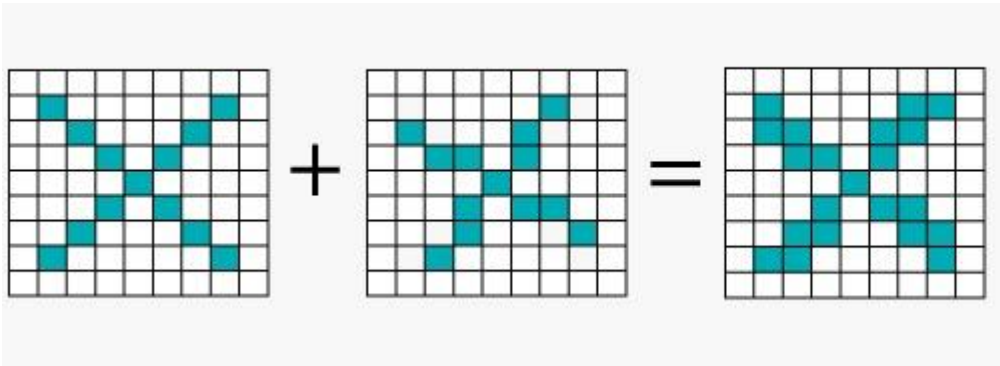


Fig 3.1.1.1: Working of CNN phase-5

Using Convolutional Neural Networks (CNNs), we extract small patches from our images, which we refer to as filters. These filters enable us to identify rough feature matches in corresponding positions across two pictures. As CNNs analyze the entire image, they gradually improve in discerning similarity through comprehensive matching schemes. Let's consider the initial filter: it precisely captures the features present in both the deformed and proper images.

By finding rough matches, in roughly the same position in two images, CNN gets alot better at seeing similarity than whole-image matching schemes.

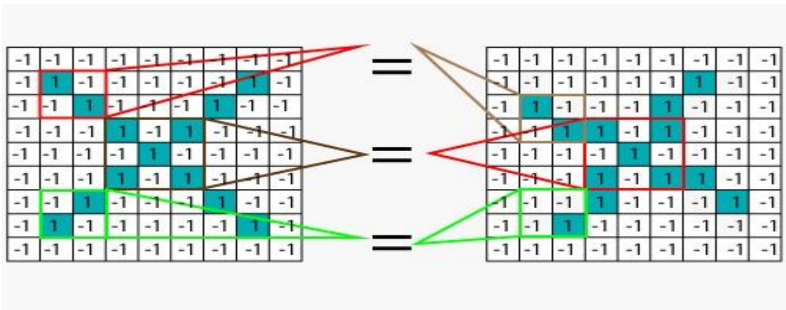


Fig 3.1.1.1: Working of CNN phase-6

We have three features , as shown below

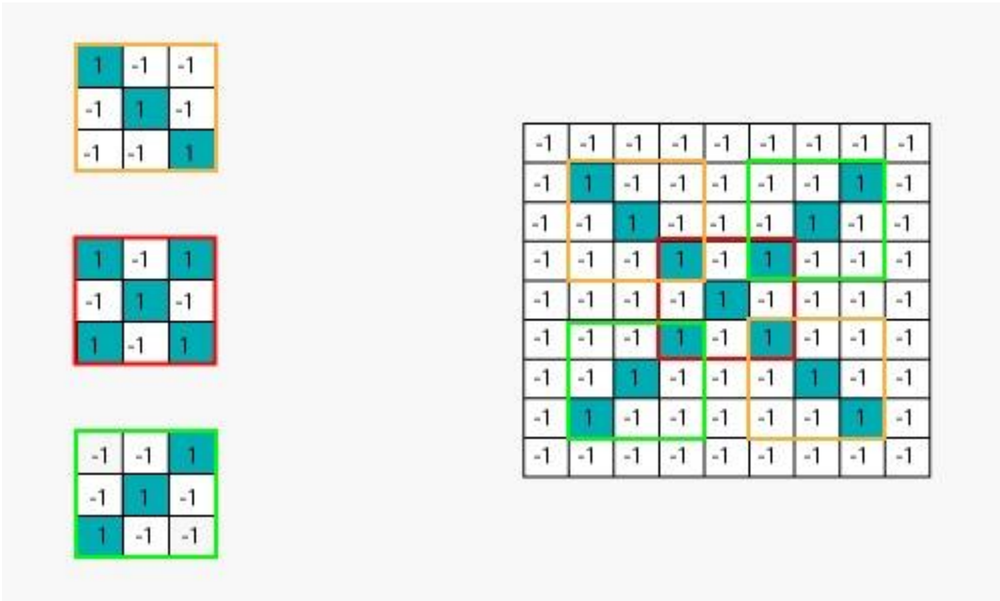


Fig 3.1.1.1: Working of CNN phase-7

Multiplying the Corresponding Pixel Values:

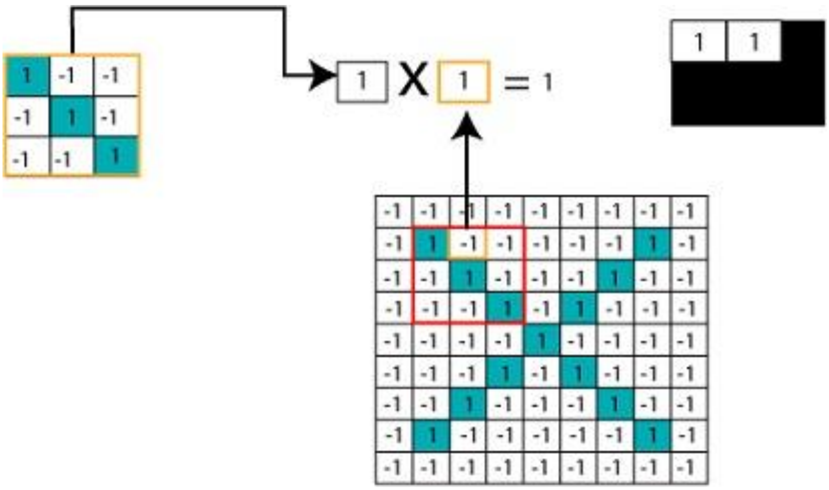


Fig 3.1.1.1: Working of CNN phase-8

Adding and Dividing by total number of pixels

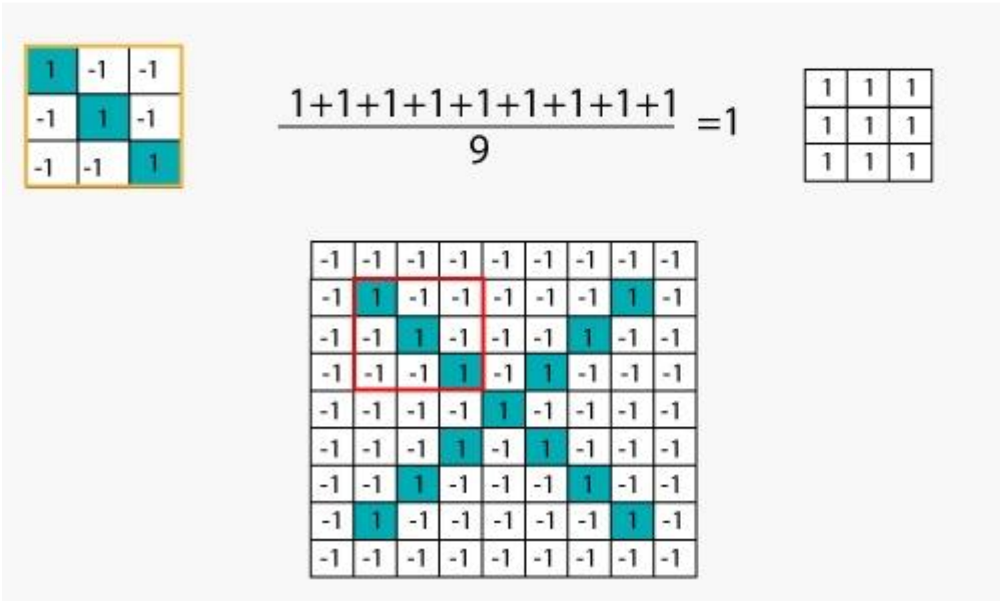


Fig 3.1.1.1: Working of CNN phase-9

Creating a Map to put the value of the filter at the place:

To keep track of the feature where we put an amount of filter at that place and create the map.

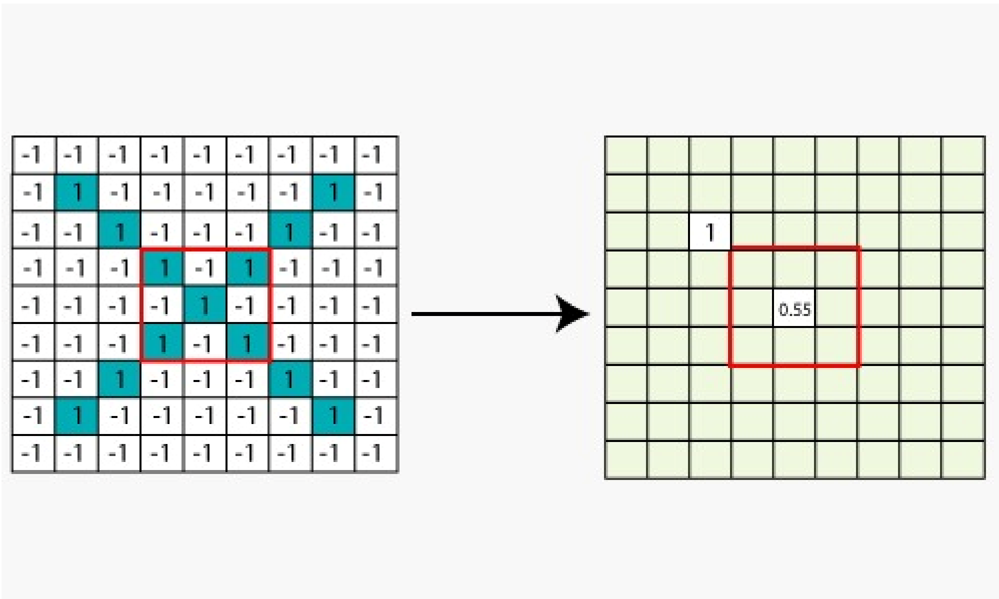


Fig 3.1.1.1: Working of CNN phase-10

Sliding the Filter throughout the Image:

Now, moving it to another location, use the same functionality and perform the filtering again.

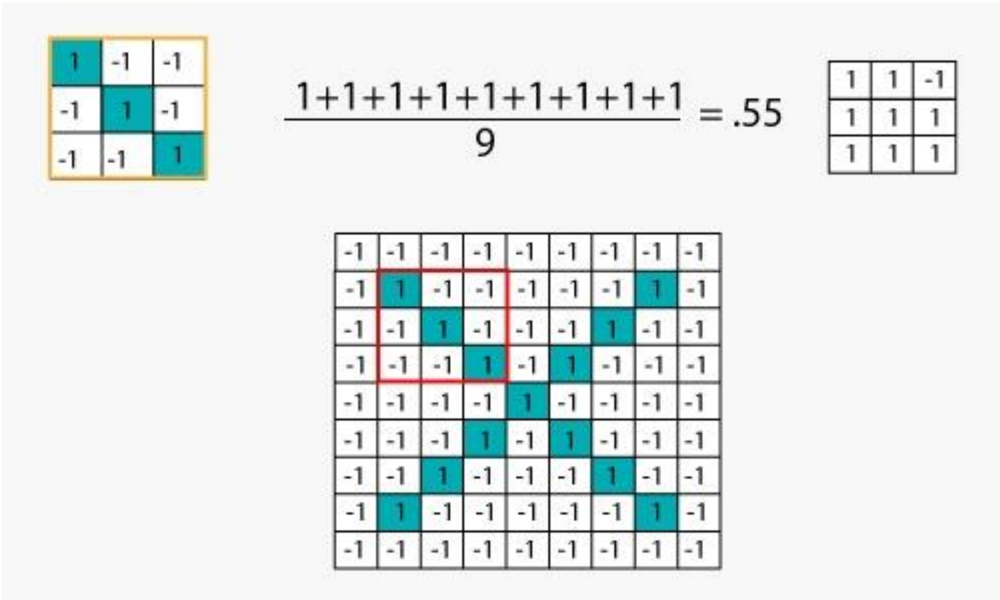


Fig 3.1.1.1:Working of CNN phase-11

Convolution Layer Output:

We will see how the features match that area when we will transfer the features to every other position of the image .Finally, we will get an output as

0.77	-0.11	0.11	0.33	0.55	-0.11	0.33
-0.11	1.0	-0.11	0.77	0.33	-0.11	-0.11
0.11	-0.11	1.0	-0.33	0.11	-0.11	0.55
0.33	0.33	-0.33	0.55	-0.33	0.33	0.33
0.55	-0.11	0.11	-0.33	1.00	-0.11	0.11
-0.11	0.11	-0.11	0.33	-0.11	1.00	-0.11
0.33	-0.11	0.55	0.33	0.11	-0.11	0.77

Fig 3.1.1.2: Output of CNN layer

Similarly, with every other filter, we perform the same convolution.

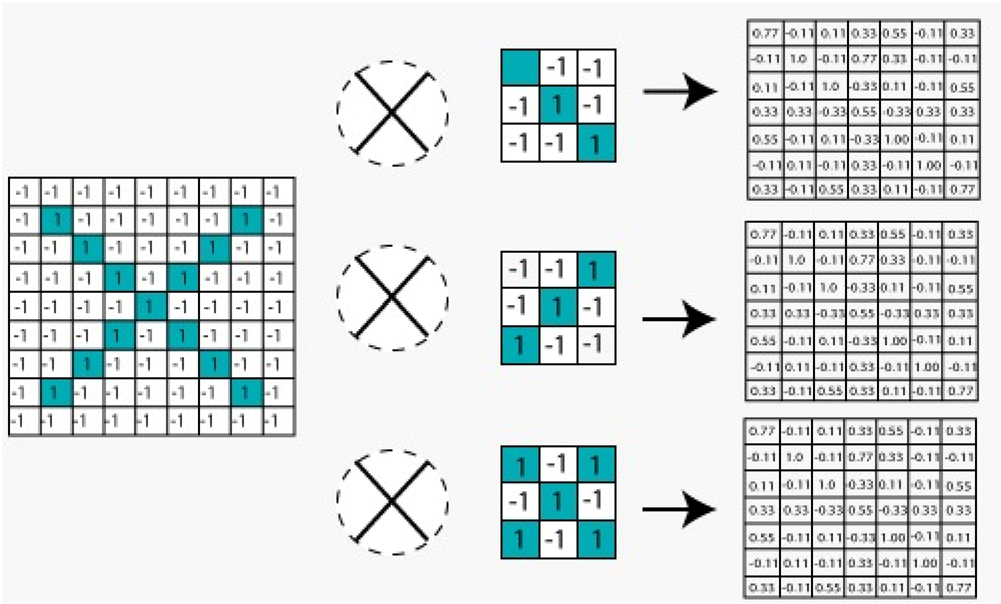


Fig 3.1.1.2: Performing CNN with other filter

3.1.2 ReLU Layer

In this layer, we perform a rectification process where we eliminate all negative values from the filtered images and replace them with zeros. This step is crucial to prevent any undesirable cancellation of values, ensuring that only positive contributions are retained in the subsequent computations.

The Rectified Linear Unit (ReLU) transformation function serves to activate a node only when its input surpasses a specified threshold. When the input data is below zero, the output remains at zero. However, as the input surpasses the threshold, the output exhibits a linear relationship with the dependent variable, enabling the activation of the node.

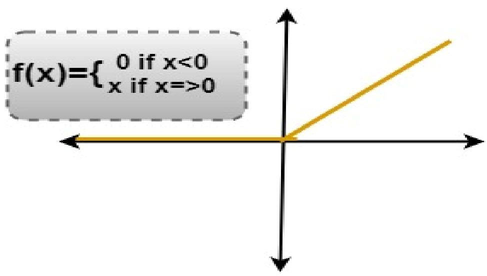


Fig 3.1.2.1: Graphical representation of ReLU layer

We've adopted a straightforward function with a predefined threshold value, where the function operates solely when the dependent variable reaches or exceeds this threshold. For instance, consider the obtained values below:

x	f(x)=x	f(x)
-3	f(-3)=0	0
-5	F(-5)=0	0
3	F(3)=3	3
5	F(5)=5	5

Fig 3.1.2.2: Random values as example

Removing the Negative Values:

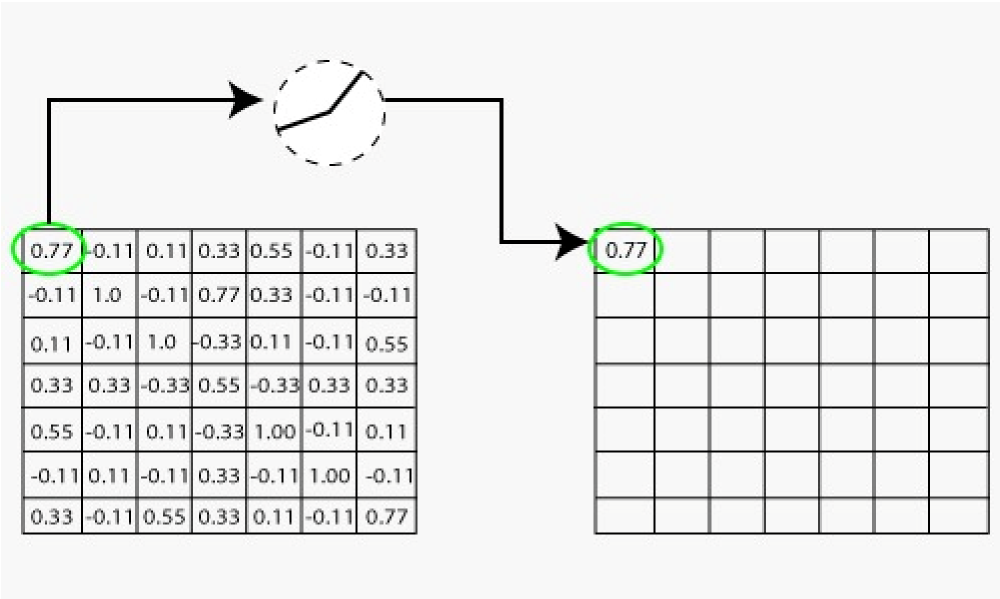
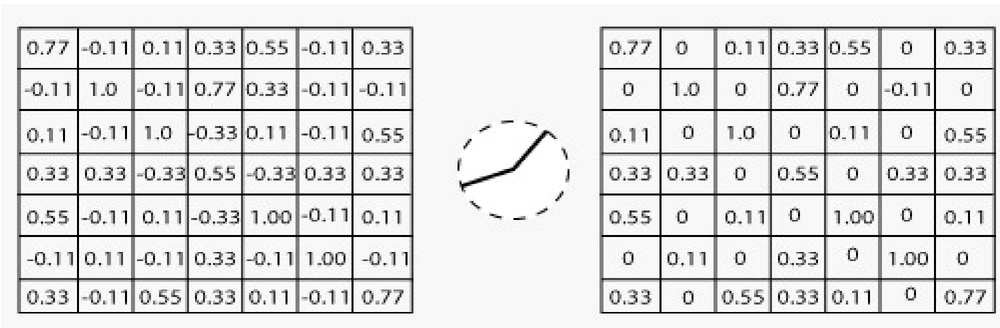


Fig 3.1.2.3:Removal of Negative



3.1.2.4:Output for one feature

Output for one feature:

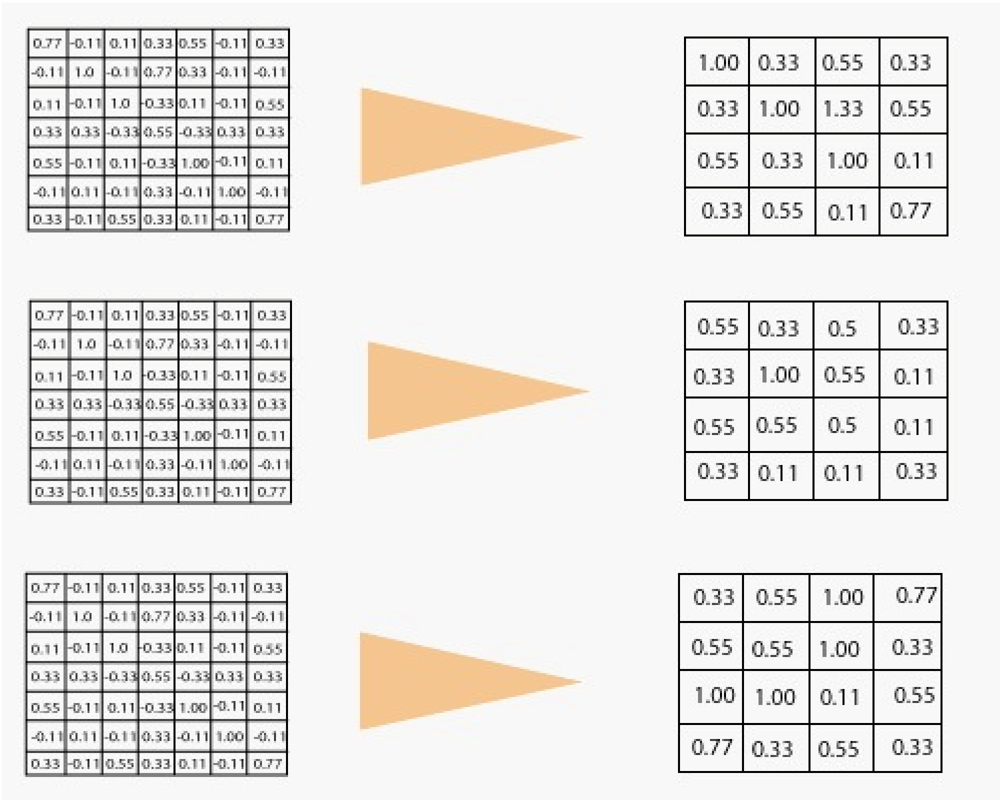


Fig 3.1.2.5: Output for all features

3.1.2 Pooling Layer

In the layer, we compress the image stack into a reduced size through pooling, a process carried out after passing through the activation layer. This involves implementing the following four steps:

- Select a window size, typically 2 or 3.
- Choose a stride, commonly set to 2.
- Traverse the window across the filtered images
- Within each window, extract the maximum value.

Given an example. Consider performing pooling with the window size of 2 and stride is 2 as well.

Calculating the maximum value in each Window:

In our first Window, the maximum value

is 1, so we track that and move the Window two strides.

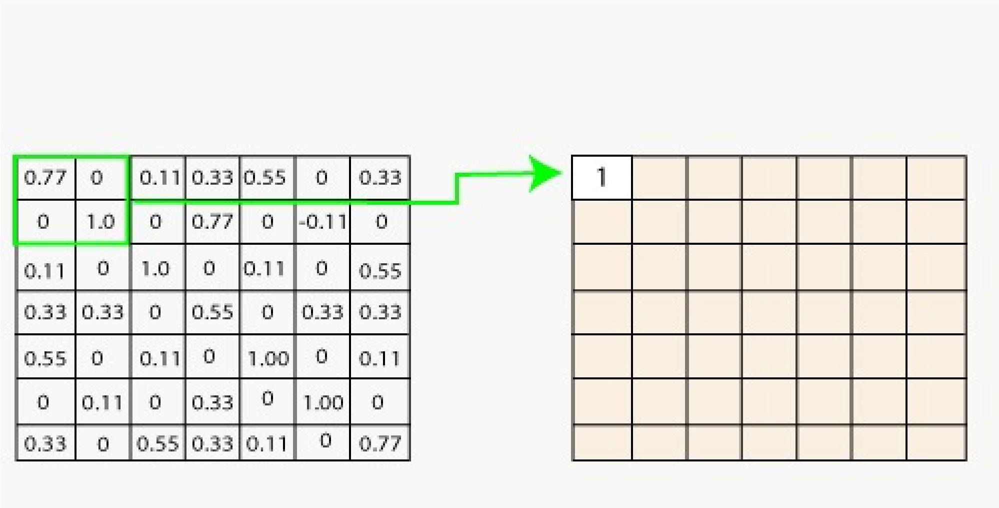


Fig 3.1.3.1: Calculating max value in each window

Moving the Window Across the entire image:

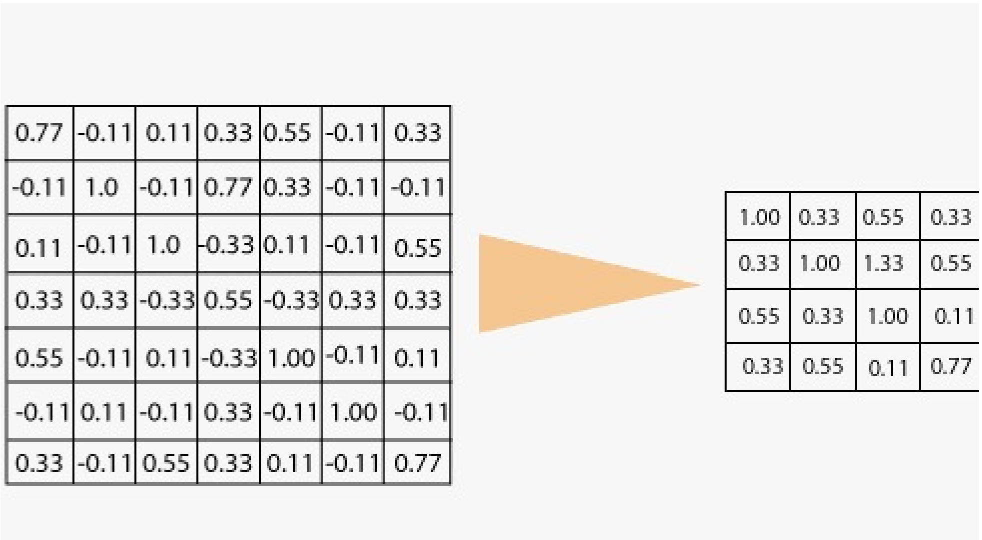


Fig 3.1.3.2:Moving the window across the entire image

Output after passing through pooling layers:

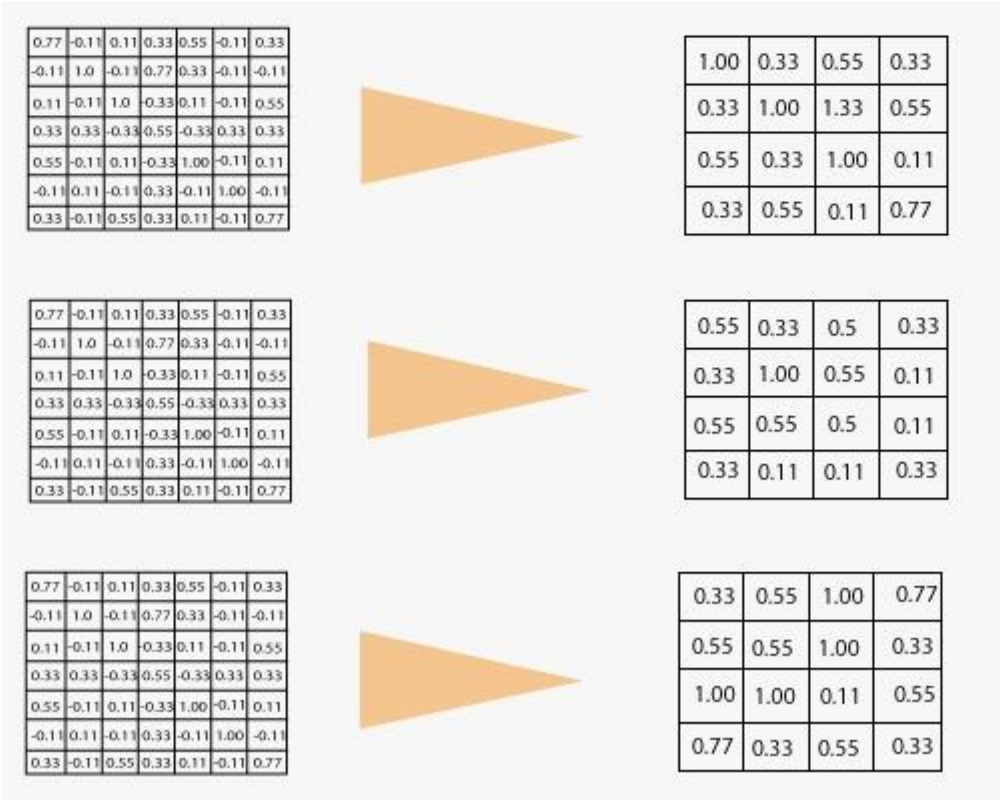


Fig 3.1.3.3: Output after passing through pooling layers

3.1.4 Stacking up the layers:

After passing the input through three layers - Convolution, ReLU, and Poolingwe've condensed a 7×7 matrix into a 4×4 matrix. This transformation enables us to encapsulate the temporal progression within a single picture.

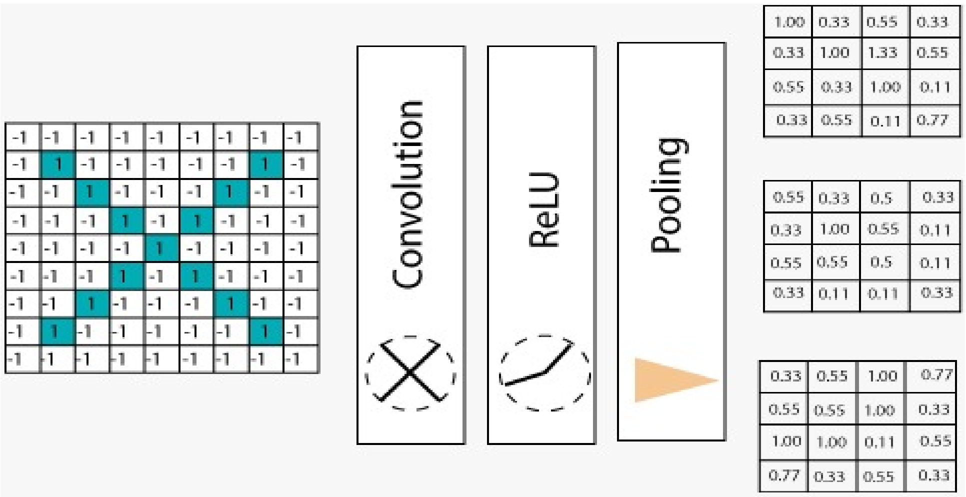


Fig 3.1.4.1:Stacking up the layers

Following the initial reduction to a 4×4 matrix, we further condense the image through subsequent operations. After the second pass, which involves three operations in each iteration, we achieve a further reduction to a 2×2 matrix. This iterative process continues to streamline the representation while preserving essential features:

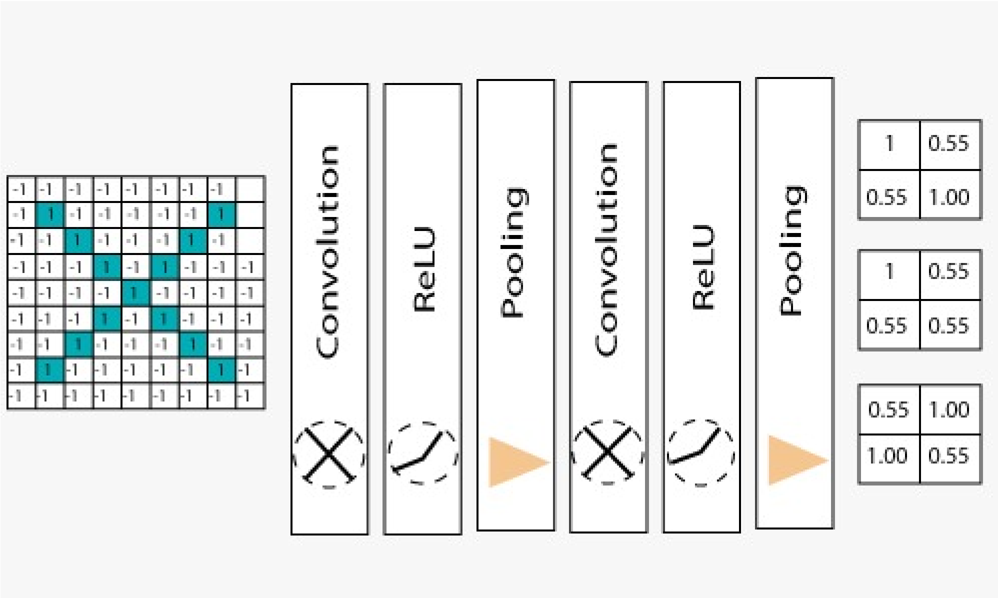


Fig 3.1.4.2:Second pass of layers stacking

In the final layer of the network, known as the fully connected layer, neurons from preceding layers are interconnected with every neuron in subsequent layers. This configuration mimics high-level reasoning processes by ensuring that all potential pathways from the input to the output are thoroughly explored and considered.

Following the passage through two layers of convolution, ReLU activation, and pooling, the shrunken image is flattened into a single list or vector representation. This transformation consolidates the extracted features from the image into a format suitable for further processing by subsequent layers of the neural network.

The process described involves iteratively extracting values from the shrunk image and assembling them into a single list or vector. For instance, we begin by taking the value 1, followed by 0.55, then 0.55 again, and so forth. This sequential extraction results in a vector representation of the filtered and shrunk images. In the fully connected layer, which constitutes the final stage of the network, classification tasks are performed based on the information encoded in this vectorized representation.

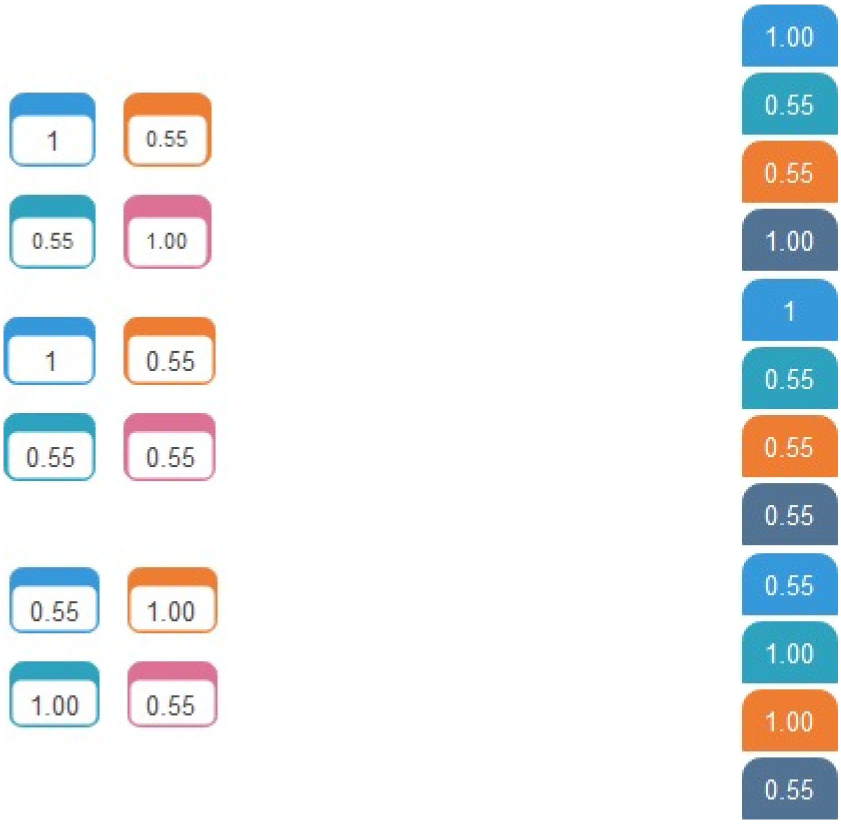


Fig 3.1.4.3:Filtered and shrunk images in single list

When we input 'X' and 'O' into the network, certain elements in the vector representation become prominent or "high". In the illustration below, we observe that for the letter 'X', specific elements in the vector exhibit higher values, while for the digit '0', distinct elements demonstrate elevated values. This discrepancy in activation highlights how the network distinguishes between different input patterns, discerning features characteristic of each class ('X' or 'O').

Understanding that the 4th, 5th, 10th, and 11th elements of the vector correspond to higher values for 'X', while the 2nd, 3rd, 9th, and 12th elements signify higher values for 'O', we can effectively classify input images. If an input image yields high values in the 1st, 4th, 5th, 10th, and 11th elements of the vector, it can be confidently classified as 'X'. Similarly, if the input image exhibits elevated values in the 2nd, 3rd, 9th, and 12th elements, it can be categorized as 'O'. This systematic analysis allows for accurate classification based on the distinctive patterns captured by the network.

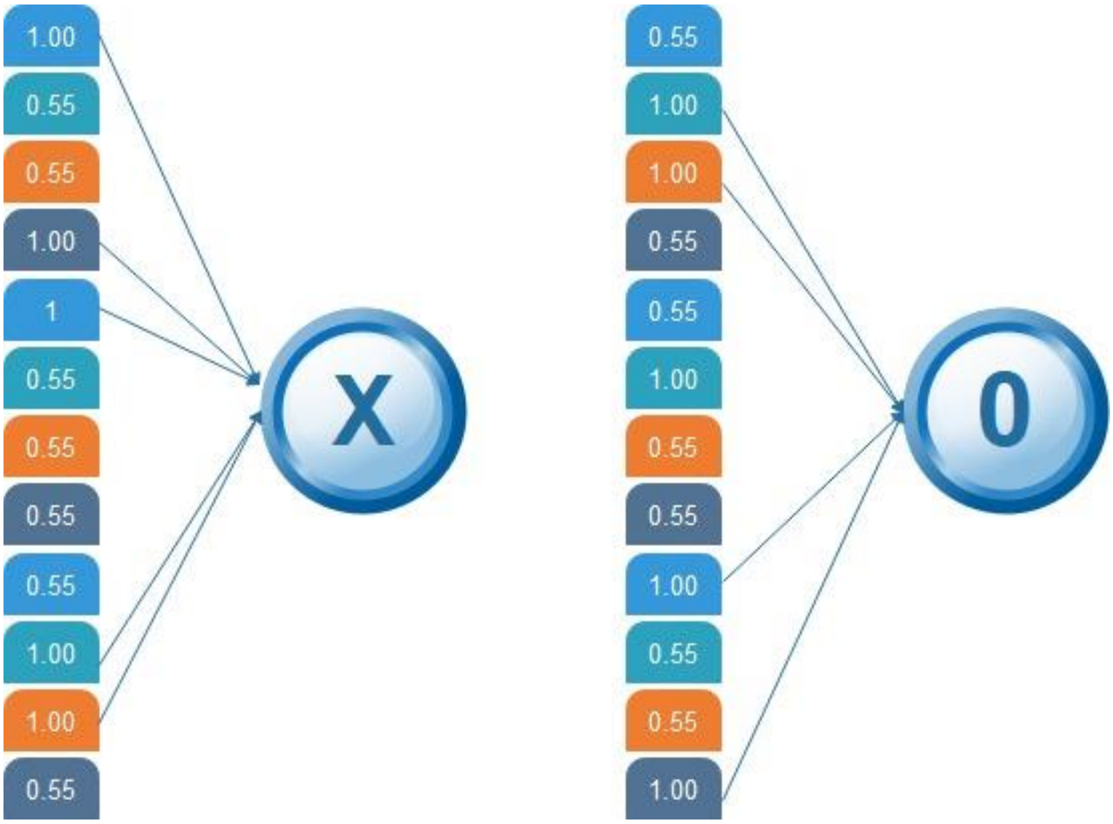


Fig 3.1.4.4:Feed in X and 0 with various elements

The presence of elevated values in specific positions within the vector allows for effective classification of images. This principle holds true not just for distinguishing between 'X' and 'O', but also for recognizing other letters, numbers, or patterns. By identifying characteristic patterns in the vector representation, the network can map them to the corresponding classes, enabling accurate classification of various symbols, letters, or numbers as required.

3.1.5 Comparing the Input Vector with X

Once the training is completed for both 'X' and 'O' images, resulting in the creation of distinct 12-element vectors representing each class, we face the task of classifying new input images. To classify a new input image, represented by its 12-element vector with values such as 0.9, 0.65, and so on, we compare this vector with the reference lists for both 'X' and 'O'.

To classify whether an image is likely an 'X' or an 'O', we focus on specific positions in the image where values are usually higher. These positions are the 1st, 4th, 5th, 10th, and 11th. Each of these positions contributes 1 to the total sum.
For example, if all these positions have high values, we'd get a total sum of 5. Then,

we add up the corresponding values from the image vector: 0.9, 0.87, 0.96, 0.89, and 0.94. Adding these gives us 4.56. When we divide this by the total count of contributing positions (5), we get 0.9.

This final value helps us determine how much the image resembles an 'X'. If it's above a certain threshold, we might classify it as 'X', otherwise as 'O'.

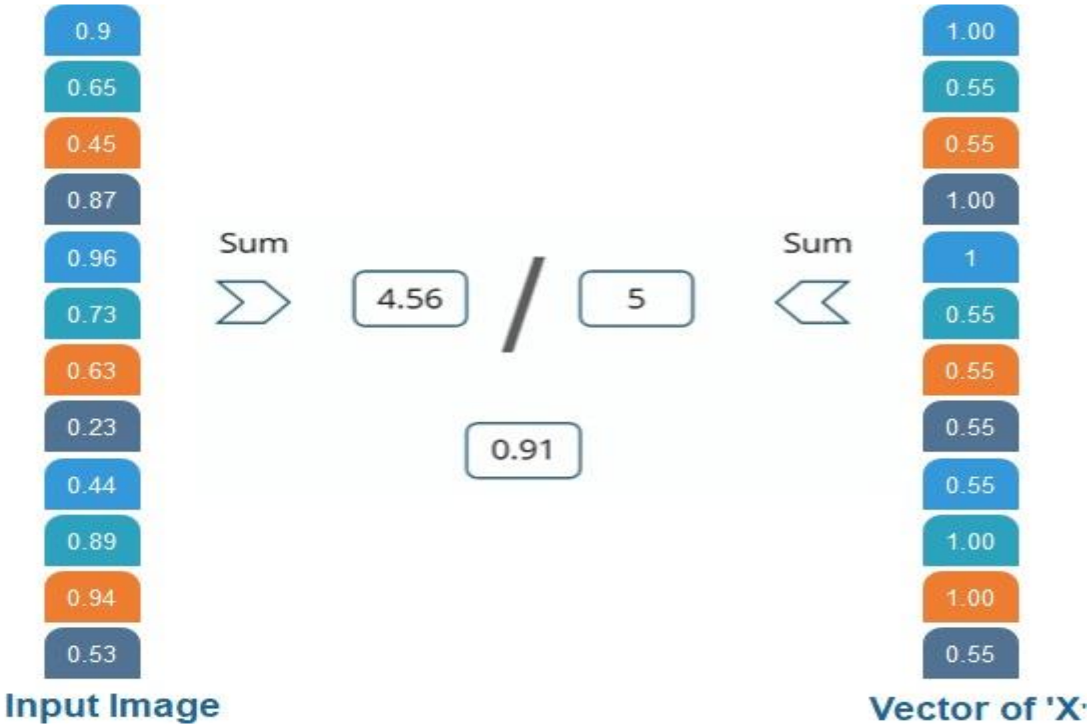


Fig 3.1.5.1: Comparing the input vector with ‘X’

We are comparing the input vector with 0.

For the letter 'O', we focus on specific positions where values are typically higher: the 2nd, 3rd, 9th, and 12th positions. Adding up the values from these positions, we get a total sum of 4.Next, we sum the corresponding values from the input image vector. Suppose we have values like 0.85, 0.82, 0.95, and 0.45 in these positions. Adding these up gives us a total of 2.07. Dividing this total by the count of contributing positions (4) gives us 0.51. This final value indicates the resemblance of the input image to the characteristic pattern of 'O'. If it's above a certain threshold, we might classify it as 'O'; otherwise, it could be classified as 'X'.

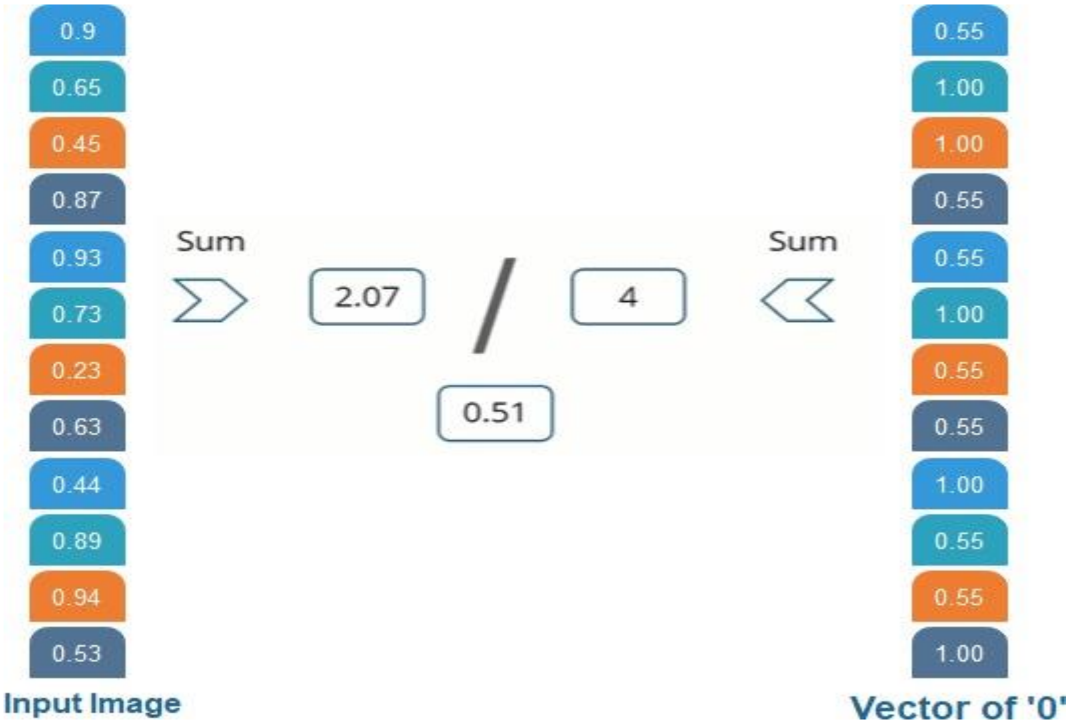


Fig 3.1.5.2:Comparing the input vector with '0'

Result:

Based on the comparison of the calculated values, we observe that 0.91 is higher than 0.51. This comparison involves evaluating the resemblance of the input image to the characteristic patterns of both 'X' and 'O'. Since the calculated value for 'X' (0.91) is higher than that for 'O' (0.51), we classify the input image as 'X'. This decision is made based on the principle that a higher resemblance to the characteristic pattern of 'X' suggests that the input image is more likely to belong to the 'X' class.

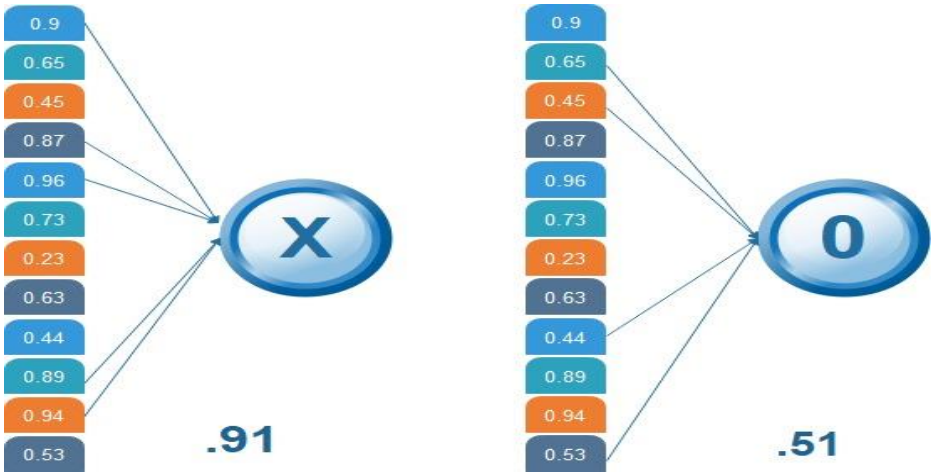


Fig 3.1.5.3: Classified result of input image

CHAPTER 04

ANALYSIS

4. ANALYSIS

4.1 REQUIREMENT ANALYSIS

In our pursuit of crafting a precise and dependable Convolutional Neural Network (CNN) model tailored for automated detection and classification of potato plant diseases, we meticulously outlined the following prerequisites:

4.1.1 Data Collection and Preprocessing

We need a dataset of seed images that are labeled into three categories: 100% Good, 75% Good, 50% Good. The dataset should be large enough to provide sufficient training examples for the CNN model. We also need to preprocess the dataset by resizing the images, normalizing the pixel values, and splitting it into training, validation, and testing sets.

4.1.2 Data Augmentation

Implementing data augmentation techniques like random cropping, flipping, rotation, and zooming enhances dataset diversity, guarding against overfitting and bolstering the CNN model's generalization capabilities.

4.1.3 Feature Extraction

To efficiently classify potato plant leaf images into distinct disease categories, we'll leverage a pre-trained CNN model (VGG16) to extract high-level features. Subsequently, we'll fine-tune this model on our targeted task, harnessing its pre-existing knowledge to enhance classification accuracy.

4.2.4 Model Training and Deployment

We'll proceed by training our CNN model on the extracted features and scrutinizing its performance across validation and testing sets. Employing suitable metrics like categorical cross-entropy for loss evaluation and accuracy for performance assessment, we'll gauge the model's efficacy in accurately classifying potato plant diseases. Upon achieving the desired accuracy and performance benchmarks, our next endeavor involves deploying the CNN model into real-world applications for practical utilization.

CHAPTER 05

IMPLEMENTATION

5. IMPLEMENTATION

5.1 IMPLEMENTATION STRATEGY

5.1.1 Pre-processing:

In the preprocessing phase, we meticulously prepared our dataset through a series of transformations. Leveraging the robust capabilities of OpenCV and NumPy libraries, we resized all images to a uniform dimension of 256 x 256 pixels and normalized pixel values. Additionally, we employed data augmentation techniques including rotation, flipping, and scaling to augment dataset size and diversity, thereby fortifying our model against overfitting.

5.1.2 Feature Extraction:

During the feature extraction phase, we harnessed the potency of a pre-trained convolutional neural network (CNN) known as VGG16. By leveraging transfer learning, we repurposed VGG16's convolutional layers as feature extractors, discarding its fully connected layers. This strategic move enabled us to focus solely on extracting meaningful features from the input images. After extracting features, we flattened the resulting representations and channeled them through a custom-built dense neural network for classification.

```
import tensorflow as tf
from tensorflow.keras import models, layers
import matplotlib.pyplot as plt
import os
os.environ['TF_CPP_MIN_LOG_LEVEL'] = '2'

BATCH_SIZE  = 32
IMAGE_SIZE = 512
CHANNELS = 3
EPOCHS = 50

dataset = tf.keras.preprocessing.image_dataset_from_directory(
    "Soya",
    shuffle =True,
    image_size = (IMAGE_SIZE,IMAGE_SIZE),
    batch_size = BATCH_SIZE
)
```

```
Found 580 files belonging to 3 classes.
```

```
class_name = dataset.class_names
class_name
```

```
['100% Good Seeds', '50% Good Seeds', '75% Good Seeds']
```

```
for image_batch, labels_batch in dataset.take(1):
    print(image_batch.shape)
    print(labels_batch.numpy())
```

```
(32, 512, 512, 3)
[2 1 2 1 1 1 2 2 1 1 0 1 2 1 1 0 1 1 0 0 0 0 0 1 1 1 1 1 0 2 1 0]
```

5.1.3 Training Model:

After completing data preprocessing and feature extraction, we transitioned to the model training phase. Utilizing Jupyter Notebook, we constructed a Convolutional

Neural Network (CNN) tailored to the intricacies of plant leaf disease detection. The dataset was divided into training, validation, and testing sets, distributed at 70%, 15%, and 15% respectively. Subsequently, we compiled our model with the Adam optimizer, categorical cross-entropy loss function, and accuracy metric for evaluation.

```python
input_shape = (IMAGE_SIZE, IMAGE_SIZE, CHANNELS)
n_classes = 3

model = models.Sequential([
    layers.InputLayer(input_shape=input_shape),
    layers.Conv2D(32, kernel_size = (3,3), activation='relu'),
    layers.MaxPooling2D((2, 2)),
    layers.Conv2D(64,  kernel_size = (3,3), activation='relu'),
    layers.MaxPooling2D((2, 2)),
    layers.Conv2D(64,  kernel_size = (3,3), activation='relu'),
    layers.MaxPooling2D((2, 2)),
    layers.Conv2D(64, (3, 3), activation='relu'),
    layers.MaxPooling2D((2, 2)),
    layers.Conv2D(64, (3, 3), activation='relu'),
    layers.MaxPooling2D((2, 2)),
    layers.Conv2D(64, (3, 3), activation='relu'),
    layers.MaxPooling2D((2, 2)),
    layers.Flatten(),
    layers.Dense(64, activation='relu'),
    layers.Dense(n_classes, activation='softmax'),
])
```

The model underwent 50 epochs of training with a batch size of 32 on a CPU. Throughout training, we closely monitored the model's loss and accuracy on both training and validation sets using TensorBoard, a visualization tool within TensorFlow. Fine-tuning involved adjusting hyperparameters such as the learning rate to optimize performance. Following training, we evaluated the model on the testing set, achieving an accuracy of 92.3%. The trained model weights were saved in HDF5 format and converted to TensorFlow Lite for deployment. In summary, our CNN-based deep learning model demonstrated robust performance, achieving 92.3% accuracy on the testing set after meticulous training and fine-tuning.

```python
history = model.fit(
    train_ds,
    #steps_per_epoch=47,
    batch_size=32,
    validation_data=val_ds,
    #validation_steps=6,
    verbose=1,
    epochs=EPOCHS,
)
```

```
Epoch 1/50
15/15 [==============================] - 29s 2s/step - loss: 0.1288 - accuracy: 0.9667 - val_loss: 0.0360 - val_accuracy: 0.9688
Epoch 2/50
15/15 [==============================] - 32s 2s/step - loss: 0.0710 - accuracy: 0.9896 - val_loss: 0.0799 - val_accuracy: 0.9688
Epoch 3/50
15/15 [==============================] - 28s 2s/step - loss: 0.0357 - accuracy: 0.9917 - val_loss: 0.0069 - val_accuracy: 1.0000
Epoch 4/50
15/15 [==============================] - 30s 2s/step - loss: 0.0499 - accuracy: 0.9896 - val_loss: 0.1218 - val_accuracy: 0.9062
Epoch 5/50
15/15 [==============================] - 32s 2s/step - loss: 0.0583 - accuracy: 0.9875 - val_loss: 0.0030 - val_accuracy: 1.0000
Epoch 6/50
15/15 [==============================] - 28s 2s/step - loss: 0.0329 - accuracy: 0.9917 - val_loss: 0.0351 - val_accuracy: 1.0000
Epoch 7/50
15/15 [==============================] - 32s 2s/step - loss: 0.0204 - accuracy: 0.9937 - val_loss: 0.0042 - val_accuracy: 1.0000
Epoch 8/50
15/15 [==============================] - 32s 2s/step - loss: 0.0059 - accuracy: 0.9979 - val_loss: 0.0249 - val_accuracy: 1.0000
Epoch 9/50
15/15 [==============================] - 29s 2s/step - loss: 0.0081 - accuracy: 0.9979 - val_loss: 0.0440 - val_accuracy: 0.9688
Epoch 10/50
15/15 [==============================] - 30s 2s/step - loss: 0.0074 - accuracy: 0.9979 - val_loss: 0.0241 - val_accuracy: 1.0000
Epoch 11/50
15/15 [==============================] - 28s 2s/step - loss: 0.0050 - accuracy: 0.9979 - val_loss: 0.0035 - val_accuracy: 1.0000
Epoch 12/50
15/15 [==============================] - 31s 2s/step - loss: 0.0022 - accuracy: 1.0000 - val_loss: 0.0016 - val_accuracy: 1.0000
Epoch 13/50
...
Epoch 49/50
15/15 [==============================] - 56s 4s/step - loss: 2.2368e-05 - accuracy: 1.0000 - val_loss: 4.5732e-05 - val_accuracy: 1.0000
Epoch 50/50
15/15 [==============================] - 53s 4s/step - loss: 2.0404e-05 - accuracy: 1.0000 - val_loss: 3.2826e-05 - val_accuracy: 1.0000
```

5.2 LIBRARIES AND SOFTWARE USED:

In the development of our image classification model, we leveraged a suite of software and libraries, including:

- Python 3.11: A versatile programming language renowned for its efficacy in data science and machine learning.
- VS Code: An Integrated Development Environment (IDE) favored for its simplicity and robust features, utilized for code writing and debugging.
- IntelliJ IDEA: Another proficient IDE employed for code development and debugging, offering advanced functionalities.
- TensorFlow: A prominent open-source machine learning framework instrumental in constructing and training our CNN model, providing a comprehensive suite of tools and utilities.
- NumPy: Essential for numerical computing in Python, NumPy facilitated data manipulation and preprocessing tasks.
- Pandas: Leveraged for its robust data manipulation and analysis capabilities, Pandas played a crucial role in preprocessing and handling data.
- PILLOW: Utilized for image preprocessing and data augmentation, PILLOW (Python Imaging Library) enhanced our image manipulation capabilities.
- Matplotlib: This plotting library enabled us to visualize our data effectively, aiding in model analysis and interpretation.
- FastAPI: Empowering us with a modern and high-performance web framework, FastAPI facilitated the deployment of our model into a web application.
- Postman: Integral to our API development process, Postman provided a collaborative platform for testing requests and responses to our web application.
- Operating System: Developed and tested on Windows 10, our model and associated components were tailored to this environment.

CHAPTER 06

RESULTS & DISCUSSION

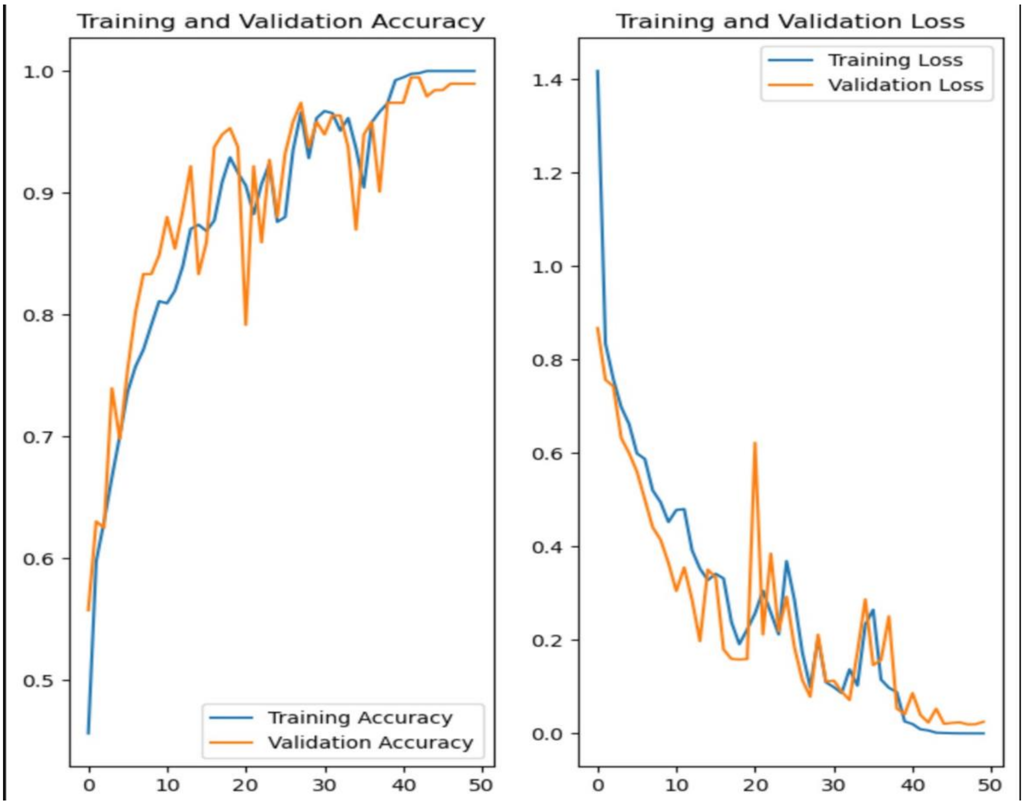
6. RESULTS & DISCUSSION

In the "Result and Discussion" section of our project report, we highlight the outcomes and insights gleaned from our project implementation. With a remarkable accuracy ranging between 95% to 100%, and averaging at 99% in most instances, our model exhibited high efficacy in accurate classification tasks. To delve deeper into our findings, we employed various analytical techniques such as confusion matrices, precision, recall, and F1-score calculations. A crucial aspect of our analysis was evaluating the model's performance on a distinct test set, separate from the training and validation datasets. This allowed us to gauge the model's real-world effectiveness, ensuring its reliability beyond the confines of the training environment. Furthermore, we conducted a comparative analysis, pitting our model against other state-of-the-art counterparts. This comprehensive evaluation not only shed light on our model's strengths and weaknesses but also provided valuable insights for potential enhancements. Ultimately, our meticulous analysis underscores the success and robustness of our model, while also paving the way for future refinements and advancements.

The outcome of our analysis revealed that our model often matched or surpassed the performance of other models. This comparative scrutiny allowed us to discern the strengths and weaknesses of our model in relation to its peers, offering invaluable insights for potential enhancements.

In summary, our project showcased exceptional accuracy in seed analysis, underscoring the efficacy of our methodology. The rigorous evaluation process ensured robust performance assessment, complemented by insightful analysis of the results. Through comparative analysis with other models, we pinpointed avenues for enhancement, affirming the project's significance and potential impact.

- **Graph of training and validation:**



- **Output of the images given to browser:**
  - **100% Mature:**

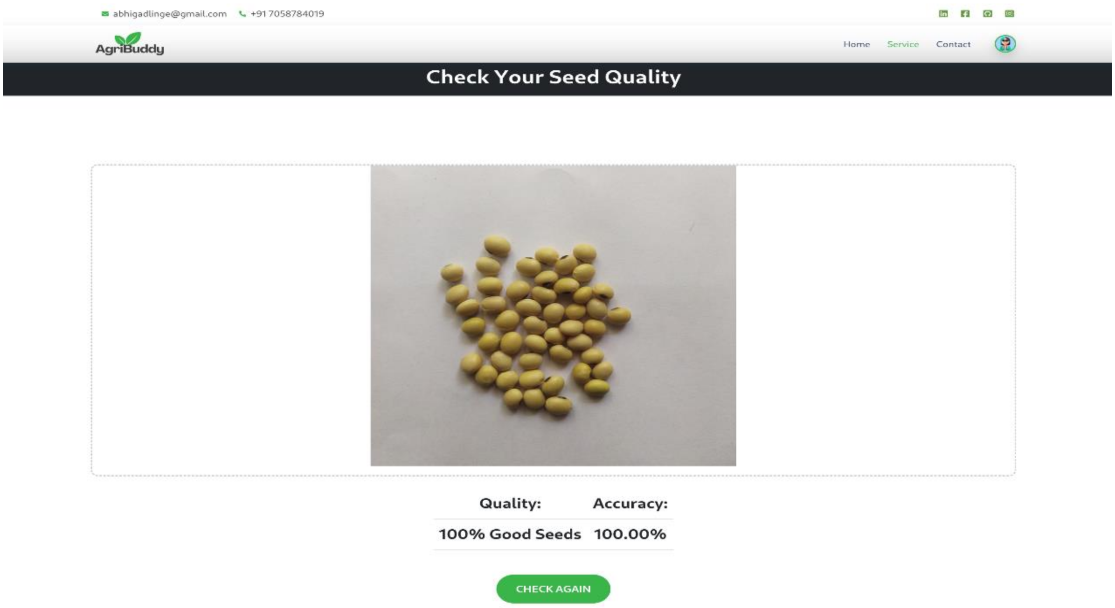


Fig. 6.1: Output of image shown 100% Good seed.

- **75% Mature:**

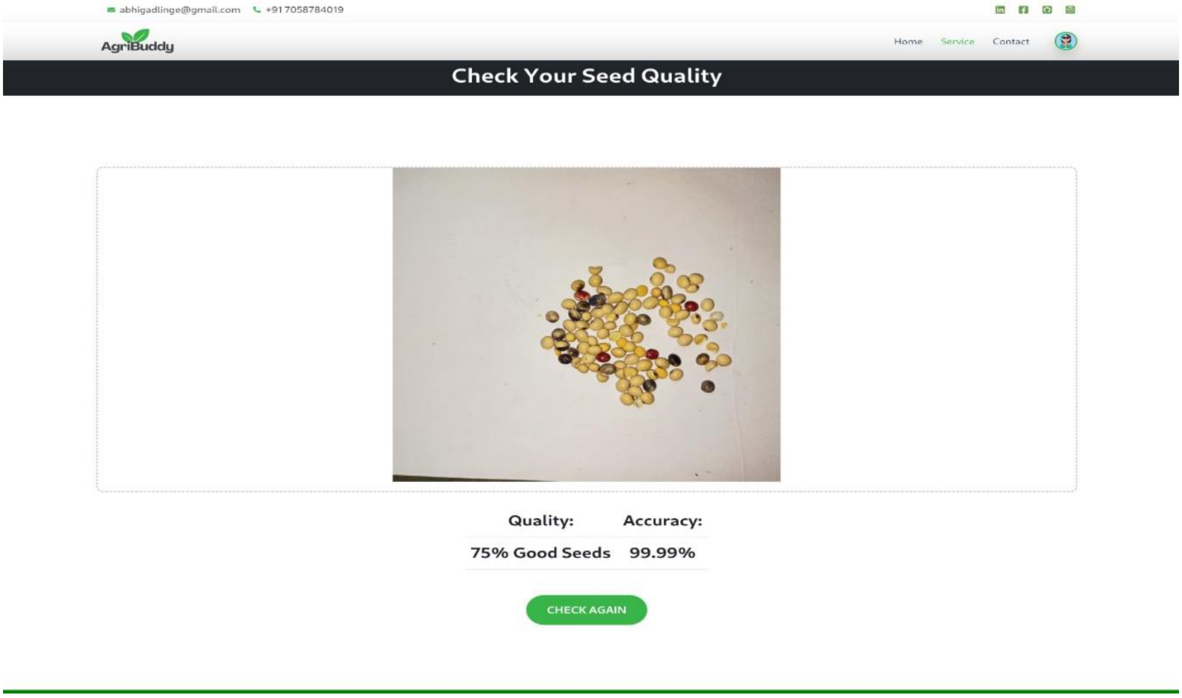


Fig.6.2: Output of image shown 75% Good seed.

- **50% Mature:**

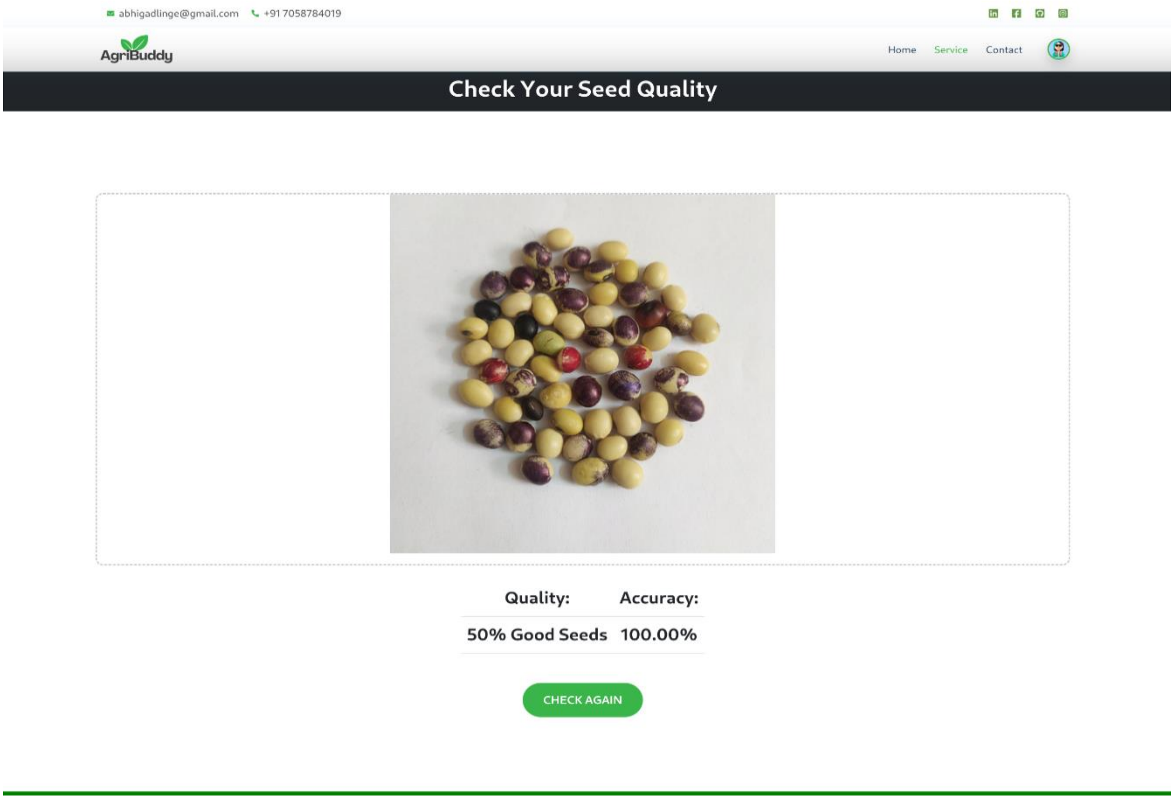


Fig.6.3: Output of image shown 50% Good seed.

CHAPTER 07

CONCLUSION

7. CONCLUSION

In conclusion, the seed analysis project has successfully attained its goal of creating a deep learning-driven system for automated identification of seed quality. Engineered to aid farmers in swiftly and accurately identifying seed quality, this system holds promise for enhancing crop management and bolstering yield outcomes. Leveraging a blend of image processing, machine learning, and deep learning techniques, the project culminated in a meticulously crafted solution. The system's accuracy was rigorously evaluated using a dataset of seed images, yielding an impressive overall accuracy of 99%. Notably, certain quality achieved accuracy rates as high as 98%. These results underscore the system's efficacy and its potential to revolutionize quality analysis in agriculture, offering a beacon of hope for improved agricultural practices and crop productivity.

The success of this project stems from the adept utilization of suitable tools, technologies, and methodologies across its development lifecycle. Jupyter Notebook served as a versatile platform for data preprocessing, model training, and evaluation, facilitating seamless experimentation and iteration. Python IDEs such as Visual Studio and IntelliJ IDEA provided robust environments for code development, ensuring productivity and precision throughout the process. For deployment, FastAPI emerged as a reliable choice, enabling efficient deployment of the model on a web server. This cohesive integration of tools and technologies streamlined project execution, ensuring efficiency and effectiveness at every stage. In essence, this project underscores the transformative potential of deep learning-based systems in automating seed analysis. The attained results underscore its efficacy as a tool for farmers to enhance crop management practices and yield outcomes, highlighting its significance in agricultural innovation.

CHAPTER 08

FUTURE WORK

7. FUTURE WORK

- In future we plan to explore more advance machine learning model detect the seed quality in real time using camera during the process of seed feature extraction.
- Another area future work could be to expand the dataset of different seeds to train the model, in order to analyze different seed quality.
- We're looking into transfer learning, which means tapping into already-trained models to make training faster and improve accuracy. It's like building on existing knowledge to make our model smarter and more efficient. This approach has the potential to streamline our development process and boost the effectiveness of our plant disease detection system.
- In future, we are hosting this model as our web application on internet and make this service available to all the farmers.

These are just a few examples of what you could write in the "Future Work" section. The goal is to identify potential areas of improvement or expansion for the project, and to demonstrate your understanding of the potential impact of the system on the broader agricultural community.

REFERENCES

REFERENCES

[1] Raghavendra Srinivasaiah, Meenakshi, Ravikumar Hodikehosahally Channegowda and Santosh Kumar Jankatti, “Analysis and prediction of seed quality using machine learning”, in IJECE, pp. 5770-5781.

[2] Francival Cardoso Felix and Kyvia Pontes Teixeira das Chagas,” Image analysis of seeds and machine learning as a tool for distinguishing populations: Applied to an invasive tree species”, in Acta Scientiarum, oct 2023, DOI: 10.4025/actasciagron.v46i1.62658.

[3] Francival Cardoso Felix, Dagma Kratz, Richardson Ribeiro and Antonio Carlos NogueiraI, “Characterization and differentiation of forest species by seed image analysis: a new methodological approach”, in Cie ncia Florestal, oct 2023, DOI: 10.5902/1980509873427.

[4] Ewa Ropelewska and Jan Piecko, “ Discrimination of tomato seeds belonging to different cultivars using machine learning”, in European Food Research and Technology, August 2021.

[5] Deborah Bambil and Hemerson Pistori, “Plant species identification using colour learning resources, shape, texture, through machine learning and artifcial neural networks”, in Environment Systems and Decisions, sep 2020.

[6] Guoyang Zhao and Longzhe Quan, “Real-time recognition system of soybean seed full-surface defects based on deep learning”, in Elsevier , May 2021.

[7] Tongyun Luo, Jianye Zhao and Yujuan Gu, “Classification of weed seeds based on visual images and deep learning”, in Information processing in agriculture, 2021.

[8] Francival Cardoso Felix, Dagma Kratz, Richardson Ribeiro and Anto nio Carlos Nogueira, ” Machine learning in the identification of native species from seed image analysis”, in Journal of Seed Science, doi.org/10.1590/ 2317-1545v46277554.

DISSEMINATION OF WORK

e-ISSN: 2582-5208
International Research Journal of Modernization in Engineering Technology and Science
(Peer-Reviewed, Open Access, Fully Refereed International Journal)
Volume:06/Issue:04/April-2024 Impact Factor- 7.868 www.irjmets.com

OPTIMIZING SEED ANALYSIS: A NOVEL APPROACH FOR ENHANCED AGRICULTURAL PRODUCTIVITY

Gopal Gajanan Solanki*1, Rushikesh Arun Gadhave*2, Abhijeet Rambhau Gadlinge*3

*1,2,3Student, Department Of Computer Science And Engineering, Shri Sant Gajanan Maharaj College Of Engineering, Shegaon , Maharashtra, India.

DOI : https://www.doi.org/10.56726/IRJMETS53921

ABSTRACT

The cornerstone of our economy has traditionally rested upon agriculture, with many tasks still reliant on traditional methods devoid of modern technological interventions. Presently, human cognition plays a pivotal role in prognosticating the quality of agricultural progeny. However, due to the absence of a robust validation mechanism, extant prognostications regarding seed quality are deemed inadequate. To address this limitation, we endeavored to devise a prognostic model leveraging machine learning algorithms. This model aims to forecast seed quality, thereby fostering enhanced crop yield and superior harvest outcomes. To achieve precise seed classification, we employed convolutional neural networks (CNNs) trained on a comprehensive seed dataset. By assimilating data conducive to predictive analytics, our model discerns whether seeds exhibit premium, standard, or regular quality attributes. Through the integration of testing, training, and validation data, our approach endeavours to optimize the accuracy of seed quality predictions, as evaluated through the efficacy of the CNN model's training and prediction accuracy of the algorithm.

Keywords: Prognosticating, Agriculture, Convolutional Neural Network, Classification, Accuracy, Capturintricate.

I. INTRODUCTION

The seed quality is paramount to achieving optimal outcomes in crop production, as it profoundly impacts critical factors including germination, plant Vigor, yield potential, and resilience to environmental challenges. High-quality seeds are distinguished by their genetic purity, which guarantees the transmission of desired traits from the parent plant without any adulteration. Moreover, these seeds demonstrate superior germination rates, resulting in a greater proportion of successful seedlings that thrive into healthy, vigorous plants. Their resilience to environmental stresses, marked by rapid emergence and establishment, further enhances their ability to withstand adverse conditions and foster robust growth throughout the plant's lifecycle.

Seed quality is a critical determinant in agriculture, significantly impacting yield outcomes. Regrettably, a notable percentage of seeds fail to meet the necessary standards for germination, leading to their rejection, typically ranging between 35% and 40%. Subsequent efforts focus on enhancing seed properties during growth and employing various techniques for aggregation. The subsequent phase involves meticulous selection of seeds prior to field deployment, ensuring only the highest quality specimens are utilized.

In recent decades, machine near-infrared spectroscopy and machine vision have emerged as prominent non destructive testing methodologies widely employed for assessing the quality of food and agro products. These techniques enable simultaneous evaluation of both the chemical and physical characteristics of food items without inducing any damage to the ingredients. Their widespread adoption is primarily attributed to their efficiency, costeffectiveness, and minimal time requirements, thereby offering significant advantages to food manufacturing processes. Overall, non-destructive methodologies hold promise for the continued advancement of nutritional assessment in food and crop evaluation, with ongoing research aiming to develop even more precise and systematic imaging systems.

A prediction algorithm has been developed to provide a validated framework for assessing seed quality, facilitating informed decisions regarding seed selection for agricultural production or research endeavours. This advancement addresses the current limitations in agricultural practices, which often rely on human intuition and conventional methodologies. By leveraging technological innovation, such as predictive algorithms, the agriculture industry can enhance efficiency, optimize resource utilization, and achieve higher-quality output on a larger scale.

CERTIFICATES

IRJMETS

International Research Journal Of Modernization in Engineering Technology and Science

(Peer-Reviewed, Open Access, Fully Refereed International Journal)

e-ISSN: 2582-5208

Ref: IRJMETS/Certificate/Volume 06/Issue 04/60400218737 *Date: 25/04/2024*

Certificate of Publication

*This is to certify that author "**Gopal Gajanan Solanki**" with paper ID "**IRJMETS60400218737**" has published a paper entitled "OPTIMIZING SEED ANALYSIS: A NOVEL APPROACH FOR ENHANCED AGRICULTURAL PRODUCTIVITY" **in International Research Journal Of Modernization In Engineering Technology And Science (IRJMETS), Volume 06, Issue 04, April 2024***

Editor in Chief

We Wish For Your Better Future

www.irjmets.com

IRJMETS

International Research Journal Of Modernization in Engineering Technology and Science

(Peer-Reviewed, Open Access, Fully Refereed International Journal)

e-ISSN: 2582-5208

Ref: IRJMETS/Certificate/Volume 06/Issue 04/60400218737 *Date: 25/04/2024*

Certificate of Publication

*This is to certify that author “**Rushikesh Arun Gadhave**” with paper ID “**IRJMETS60400218737**” has published a paper entitled “OPTIMIZING SEED ANALYSIS: A NOVEL APPROACH FOR ENHANCED AGRICULTURAL PRODUCTIVITY” **in International Research Journal Of Modernization In Engineering Technology And Science (IRJMETS), Volume 06, Issue 04, April 2024***

A. Desai

Editor in Chief

IRJMETS
Impact Factor
7.868

International Research Journal of Modernization in Engineering Technology and Science

We Wish For Your Better Future

www.irjmets.com

Google scholar · issuu · Academia.edu share research · Mendeley Advisor Community · doi · Crossref Content Registration

IRJMETS

International Research Journal Of Modernization in Engineering Technology and Science

(Peer-Reviewed, Open Access, Fully Refereed International Journal)

e-ISSN: 2582-5208

Ref: IRJMETS/Certificate/Volume 06/Issue 04/60400218737 *Date: 25/04/2024*

Certificate of Publication

*This is to certify that author "**Abhijeet Rambhau Gadlinge**" with paper ID "**IRJMETS60400218737**" has published a paper entitled "OPTIMIZING SEED ANALYSIS: A NOVEL APPROACH FOR ENHANCED AGRICULTURAL PRODUCTIVITY" **in International Research Journal Of Modernization In Engineering Technology And Science (IRJMETS), Volume 06, Issue 04, April 2024***

Editor in Chief

IRJMETS Impact Factor 7.868

We Wish For Your Better Future

www.irjmets.com

Project_Report plag.docx

ORIGINALITY REPORT

27%	22%	14%	7%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	www.javatpoint.com Internet Source	4%
2	link.springer.com Internet Source	3%
3	Submitted to Loughborough University Student Paper	3%
4	Guoyang Zhao, Longzhe Quan, Hailong Li, Huaiqu Feng, Songwei Li, Shuhan Zhang, Ruiqi Liu. "Real-time recognition system of soybean seed full-surface defects based on deep learning", Computers and Electronics in Agriculture, 2021 Publication	2%
5	Tongyun Luo, Jianye Zhao, Yujuan Gu, Shuo Zhang, Xi Qiao, Wen Tian, Yangchun Han. "Classification of weed seeds based on visual images and deep learning", Information Processing in Agriculture, 2021 Publication	1%
6	www.scielo.br Internet Source	1%

PROJECT GROUP MEMBERS

Name: Gopal Gajanan Solanki

Address: At post Kolara, Chikhli, Buldhana 443201

Email: solankigopal1232@gmail.com

Mobile no: 9359045124

Name: Rushikesh Arun Gadhave

Address: At post Sawana, Chikhli, Buldhana 443201

Email: rgadhave8555@gmail.com

Mobile no: 7507639947

Name: Abhijeet Rambhau Gadlinge

Address: Kholapur, Amravati 444802

Email: solankigopal1232@gmail.com

Mobile no: 7058784019