D.K.T.E. Society’s Textile and Engineering Institute, Ichalkaranji.

(An Autonomous Institute, Affiliated to Shivaji University, Kolhapur)

Department of Information Technology

2019-20



**Project Report On**

**Automated Fruit Grading System**

**Under the Guidance Of**

Prof. D. M. Kulkarni

**Submitted By:**

1. Amruta V. Mahadik 16UIT11031XX
2. Shruti S. Mukkannawar 16UIT11033XX
3. Apoorva S. Patil 16UIT11038XX
4. Priya B. Taware 16UIT11054XX
5. Pooja R. Patil 16UIT51061XX

|  |  |  |
| --- | --- | --- |
| Prof. D. M. Kulkarni | Prof. (Dr).D.V. Kodavade | Prof. (Dr).P.V.Kadole |
| **(Project Guide)** | **(H.O.D)** | **(Director)** |



**YEAR 2019-2020**

**DEPARTMENT OF INFORMATION TECHNOLOGY**

**CERTIFICATE**

This is to certify that the project report entitle “**Automated Fruit Grading System**” is record of project work carried out in this college by,

Miss. Amruta V. Mahadik (16UIT11031XX)

Miss. Shruti S. Mukkannawar (16UIT11033XX)

Miss. Apoorva S. Patil (16UIT11038XX)

Miss Priya B. Taware (16UIT11054XX)

Miss. Pooja R. Patil (16UIT51061XX)

In partial fulfillment of the requirement for degree of BACHELOR OF TECHNOLOGY in INFORMATION TECHNOLOGY of SHIVAJI UNIVERSITY, KOLHAPUR. This project report is record of their own work carried out under my supervision and guidance during academic year 2019-2020.

Prof. D. M. Kulkarni Prof. (Dr.) D. V. Kodavade

**[Project Guide]** [**Head of the Department]**

Prof. (Dr.) P.V. Kadole

**[Director]**

**ACKNOWLEDGMENT**

With great pleasure we wish to express our deep sense of gratitude to Prof. D.M. Kulkarni for this valuable guidance, support and encouragement in completion of this project report. Also, we would like to take opportunity to thank our head of department Dr. D.V. Kodavade for his co-operation in preparing this project report.

We feel gratified to record our cordial hands to our staff members of our Information Technology department for their support, help and assistance which the extended as and when required.

Thank you,

Miss. Amruta V. Mahadik 16UIT11031XX

Miss. Shruti S. Mukkannawar 16UIT11033XX

Miss. Apoorva S. Patil 16UIT11038XX

Miss. Priya B. Taware 16UIT11054XX

Miss. Pooja R. Patil 16UIT51061XX

**ABSTRACT**

Agriculture sector plays a key role in the economic development of India. The task of fruit grading is vital in the agricultural industry because there is a great demand for high quality fruits in the market. However, fruit grading by human is inefficient, labour intensive and prone to error. The automated grading system not only speeds up the time of processing, but also minimizes error. There is a great demand for apples in both local and foreign markets. The apple fruit is very delicate and hence careful handling of this fruit is required during grading. Thus, this automatic and effective apple fruit grading system based is computer vision techniques. The proposed quality evaluation method consists of two phases: development of hardware and software. The hardware is developed to capture the image of the apple and move the fruit to the appropriate bins without manual intervention. The software is developed using image processing techniques to analyse the fruit for defects and ripeness. Experiments were carried out on several images of the fruit.

**INDEX**

1. **Introduction**
   1. Problem Definition
   2. Objectives of project
   3. Scope & limitations of project
   4. Timeline for project
2. **Literature Review**
   1. Technology review
   2. Literature review
3. **Requirement Analysis**
   1. **Functional Requirements**
   2. **System Requirements**
      1. Hardware Requirements(if applicable)
      2. Operating System Requirements
   3. **Tools and technologies required(if any)**
4. **System Design**
5. **Architectural Design** 
   1. Algorithmic description of each modules
   2. System Modeling
      1. Dataflow Diagram
      2. Flowchart
      3. Use case Diagram

**5. Implementation details**

5.1 Algorithm

5.2 Environmental setup for running the model

**6. Testing**

**7. Performance Analysis**

**8. Applications**

**9. Project Ethics**

**10. Reference**

1. **INTRODUCTION**

Apple grading is a challenging area because of its variation of deficiencies in type and shape. However, manual methods are still employed to handle the task of sorting and grading this fruit. It has been noticed that the use of these methods have a lot of challenges such as deficiency of standard for quality products based on personal judgment of workers in their visual perception of the fruits, poor quantities of produce, fatigue and slow decision making on the part of the workers. Therefore, if the work of classifying apples, in apple-processing industries, is done by an automated system, the work will be much faster and efficient and quality control will be made much easier. To implement this system, we used Convolutional Neural Network (CNN). Convolutional neural network is a type of artificial neural network in which the connectivity pattern between its neurons is inspired by the organization of the animal visual cortex. The visual cortex has small regions of cells that are sensitive to specific regions of visual field. Some individual neuronal cells in brain respond only in presence of edges of a certain orient action. For example, neurons fires when exposed to vertical edges and some when shown horizontal or diagonal edges

We are going to train the model to classify apple into 3 categories such as mature, pre-mature, rotten. For example, we need to train the model for the mature apple in image. To do so, we first need to teach the computer how a mature apple looks like. The more images we use to train, the better is the accuracy of the model.

Convolutional neural network (CNN) is one of the most popular techniques used in improving the accuracy of image classification. CNN has convolution layer at the beginning which breaks the image into number of tiles, the machine then tries to predict what each tile is. Finally, the computer tries to predict what's in the picture based on the prediction of all the tiles.

**1.1 Problem Statement**

To develop a fruit grading system based on color and texture of the fruit using appropriate feature extraction techniques and classification algorithms.

**1.2 Objectives**

* + Extract the relevant features from the images using feature extraction algorithm and image processing tools.
  + Classify the fruit image in the category such as premature, mature and rotten fruit to decide its grade.

**1.3 Scope**

India produces 44.04 million tons of fruits annually. A tremendous scope thus arises for grading the fruits for quality inspection tests from dispatch from farm to the consumers. Fruits must be graded for quality aspects.

**1.4 Timeline of project**

|  |  |  |
| --- | --- | --- |
| **Topic** | **Start Date** | **End Date** |
| Area Finalization | 22/07/2019 | 28/07/2019 |
| Topic Finalization | 28/07/2019 | 12/8/2019 |
| Literature review | 14/08/2019 | 05/09/2019 |
| Documentation of Synopsis | 5/10/2019 | 15/10/2019 |
| Requirement Analysis | 5/1/2020 | 15/1/2020 |
| System Requirement | 15/1/2020 | 25/1/2020 |
| Module Identification | 27/1/2020 | 7/2/2020 |
| Architecture Module | 9/2/2020 | 16/2/2020 |
| Design | 18/2/2020 | 25/2/2020 |
| Analysis Documentation | 26/2/2020 | 5/3/2020 |
| Implementation | 6/3/2020 | 10/3/2020 |
| Testing | 13/3/2020 | 20/3/2020 |
| Testing | 22/3/2020 | 28/3/2020 |
| Testing | 29/3/2020 | 3/4/2020 |
| Testing | 5/4/2020 | 8/4/2020 |
| Report Making | 9/4/2020 | 11/4/2020 |
| Performance Analysis | 12/4/2020 | 16/4/2020 |

1. **LITERATURE REVIEW**
   1. **Technology review**

Literature survey is most important step in software development process. Before developing the tool it is necessary to determine which language and operating system is used to develop the tools. In these project mainly Machine Learning and Image Processing techniques are used. Once the programmer start building the tool the programmer need lots of external support. Before building the system the above consideration is taken into account for developing the proposed system.

* 1. **Literature review**

Many researchers are working in fruit grading automation system. Following section provides summary of some of the work done in this area.

[1] Suresha M, Shilpa N.A, Soumya B. 2012 [1]. Effective automatic grading of apples is proposed. The apples RGB images is converted into HSV image and threshold based approach is used for segmentation of apples from background. Average red and green color components of apples are determined for classification of apples. With the help of support vector machines (SVMs), classification is done and found accuracy of 100%.

[2] Devrim, Unay. Bernard, Gosselin.1998.[2]. A Computer vision based system is introduced to automatically grade apple fruits. Segmentation of defected skin is done by global thresholding techniques. Statistical features are extracted from the segmented areas and then fruit is graded by a supervised classifier. Linear Discriminant, nearest neighbor, fuzzy neighbor, ad boost and support machines classifiers are tested for fruit grading and found 89% recognition rate using SVM.

[3] Manali R. Satpute, Sumati M. Jagdale. [3]. An automatic fruit quality inspection system for sorting and grading of tomato fruit and defected tomato detection discussed here. The main aim of this system is to replace the manual inspection system. This helps in speed up the process improve accuracy and efficiency and reduce time. This system collect image from camera which is placed on conveyor belt. Then image processing is done to get required features of fruits such as texture, color and size. Defected fruit is detected based on blob detection, color detection is done based on thresholding and size detection zis based on binary image of tomato. Sorting is done based on color and grading is done based on size.

[4] PL.Chithra, M.Henila. 2017. [4]. Proposed a variation in already existing algorithm to find weather an apple is defected or not. Defected part of an apples image is identified using k-means segmentation algorithm. Colour images of apples with certain defects are taken as input dataset for applying the newly proposed algorithm. Colour components of apples image are used for segmentation. K-means clustering technique- An iterative process is used to partition an apples image into K clusters. Pixels are clustered based on colour intensity values and images are generated to identify the defected part after detecting the defected part, image enhancement is done using median filter in order to enhance the defected part and which in turn will increasing the efficiency of classification process in the proposed algorithm. The experimental study clarifies the effectiveness of both already existing method and newly proposed method. Average accuracy of proposed method is 91.67%.

1. **REQUIREMENT ANALYSIS**
   1. **Functional requirements**

Our project takes image as an input and from that image machine predicts the category of fruit depending upon its features. These kind of application are part of machine learning. We need to train the machine with numerous images. In this project we use convolutional neural network (CNN) for training the machine.

Why CNN?

Image classification is part of our project and CNN is basically designed for the image classification. Before CNN, fully connected feedforward networks were used to learn features as well as classify data, it basically means that each pixel in image will be treated as the input that is later passed to neural network. It is not practical to apply this architecture to images. A very high number of neurons would be necessary, due to large input size associated with images, where each pixel is a relevant variable. The model fails because it was never taught to extract and understand higher level features from the image. Here the CNN comes to rescue and brings solution to this problem as it reduces the number of free parameters, allowing the network to be with lesser parameters. When we pass in an image as the input to the model, the model extracts features from the images. At the start these features might look like intricate edges in the images like edges, texture etc. But as we proceed to higher levels these features become more prominent.

* 1. **System requirements**
     1. **Hardware requirements**
* 8GB RAM
  + 1. **Operating system requirements**
* Windows 10
  1. **Tools and technologies required**
* Google Colab
* Python 3.7
* Python libraries

**4. SYSTEM DESIGN**

**4.1 Architectural design**

**Features collected from image dataset**

Image

**Image**

**Pre-processing**

**Classification in 3 classes**

**Feature**

**Extraction**

**Fruit grading**

Fruit grade

Fig. 4.1 System Architecture for Fruit Grading System

* + 1. **Algorithm Description for each module**

We are building three modules. In **First module** on the given input image we will perform image processing which eliminates noise from image and thereby helps in clear identification of features. The **Second module** is feature extraction where input data is transformed into a set of features. Features are distinctive properties of input patterns that help in differentiating between the categories of input patterns. The features extracted will be Color, Texture. The **Third module** is Classification where depending on the features extract we will classify them into three categories that are mature, premature and rotten.

Module 1: Image-Preprocessing

The aim of image pre-processing is an improvement of the image data that suppresses unwilling distortions or enhances some image features important for further processing. Here we perform image preprocessing on acquired image so we can get clear view of features of image.

Module 2: Feature Extraction

Performing feature extraction we extract features color, size and texture of the fruit.

* **Colour Detection**

Acquired RGB image is pre-processed then we define the list of boundaries for BGR colour. Further apply loop over the boundaries and find the colours within the specified boundaries to apply mask.

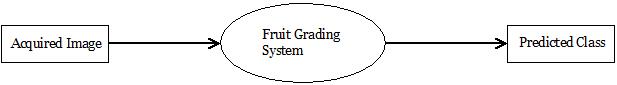
* **Texture Detection**

This is used to detect and analyze surface of fruit. Features such as entropy, mean, standard deviation of good and defected fruit is calculated and this features are used to characterize fruits surface texture.

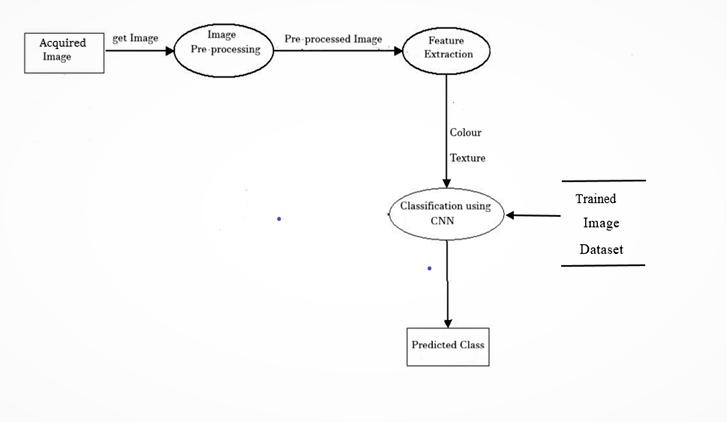
Module 3: Classification:

According to features extracted in previous step image is classified in premature, mature or rotten class by comparing acquired image with trained dataset. Accordingly results are displayed to user.

* 1. **System Modeling Design**
     1. Dataflow diagram

****

4.2.1. Fig : DFD Level 0



4.2.2. Fig : DFD Level 1

* + 1. **Use-case Diagram**

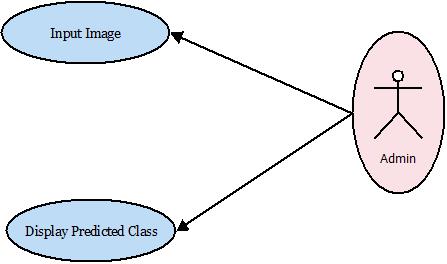
****

Fig 4.2.2 Use-case Diagram

* + 1. **Flow Chart**

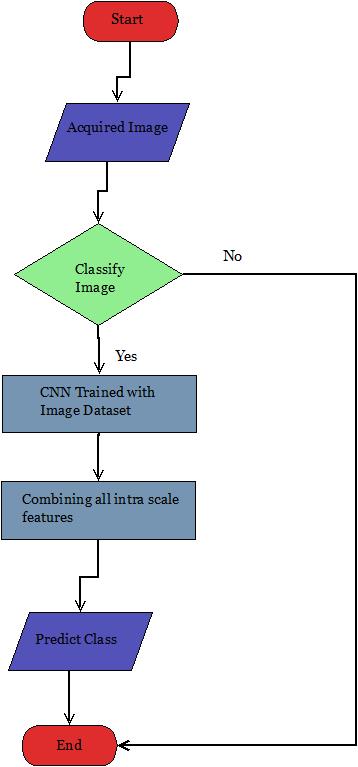
****

Fig 4.2.3 Flowchart

1. **IMPLEMENTATION DETAILS**

In neural networks, convolutional neural network (ConvNeta or CNNs) is one of the main categories to do images recognition, images classifications. Objects detections, recognition faces, etc. are some of the areas where CNNs are widely used.

CNN image classifications take an input image, process it and classify it under certain categories. Computers see an input image as array of pixels and it depends on the image resolution. Based on the image resolution, it will see h x w x d [h=height, w=width, d-dimension]. Example, an image of 6x6x3 array of matrix of RGB [3 refers to RGB values] and an images of 4x4x1 array of grayscale image.

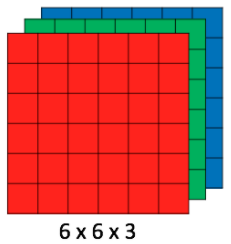
****

Fig 1: RGB matrix

Technically, machine learning CNN models to train and test, each input image will pass it through a series of convolution layers with filters (kerals), Pooling fully connected layers (FC) and apply softmax function to classify an object with probabilistic values between 0 and 1. The below figure is a complete flow of CNN to process an input image and classifies the objects based on values.

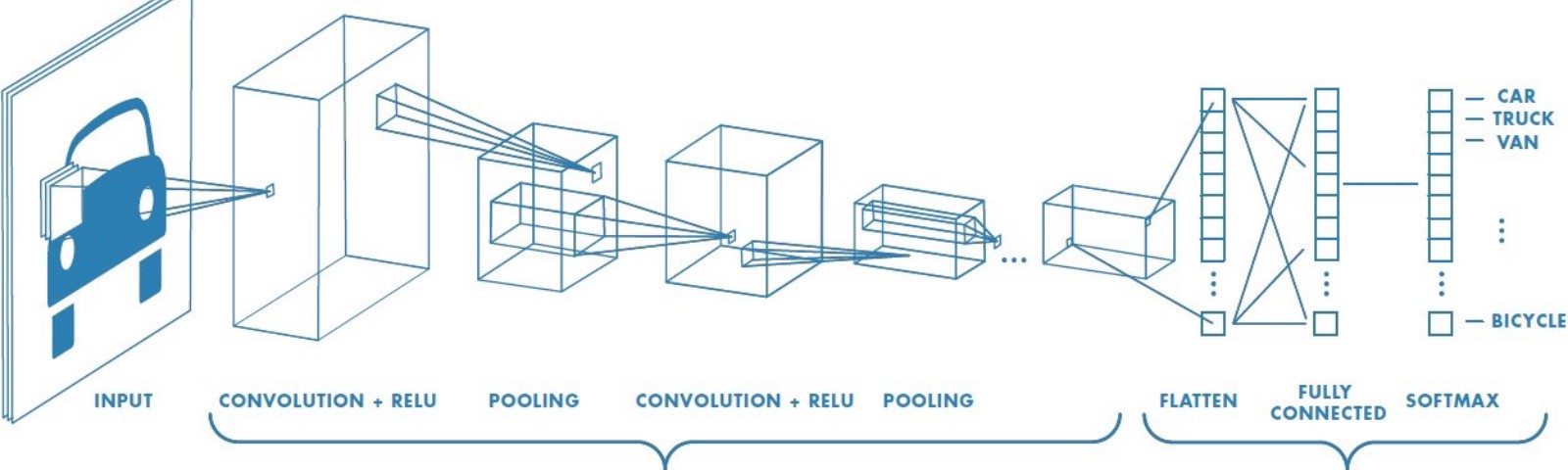


Fig 2: Neural Network with many convolution layers

Convolution Layer

Convolution is the first layer to extract feature from an input image. Covolution preserves the relationship between pixels by learning image features using small squares of input data. It is the mathematical operation that takes two inputs such as image matrix and a filter or a kernel.

* An image matrix (volume) of dimension **(h x w x d)**
* A filter **(fhxfwx d)**
* Outputs a volume dimension **(h-fh+1) x (w-fw+1) x 1**

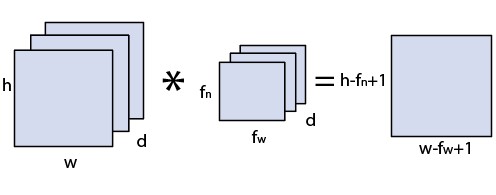
****

Fig 3: Image matrix multiplies kernel or filter matrix

Consider a 5x5 matrix whose image pixels values are 0, 1 and filter matrix 3x3 as shown in below,

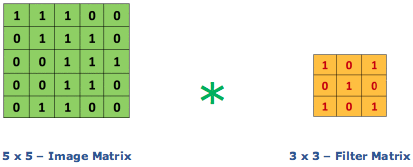


Fig 4: Image matrix multiplies kernel or filter matrix

Then the convolution of 5x5 image matrix multiplies with 3x3 filter matrix which is called “Feature Map” as output shown in below,

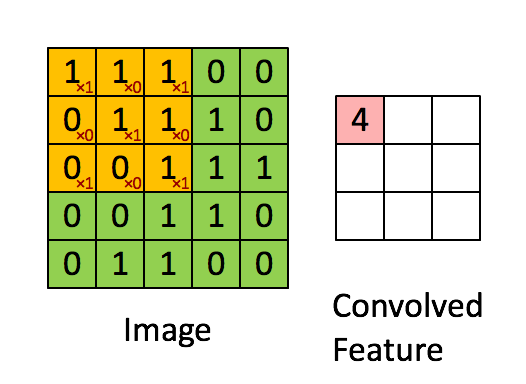


Fig 5: 3 x 3 output matrix

Convolution of an image with different filters can perform operations such as edge detection, blur and sharpen by applying filters. The below example shows various convolution image after applying different types of filters. (kernel)

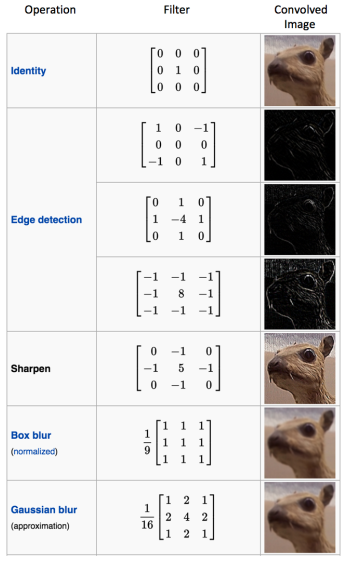


Fig 6: Some common filters

Padding

Sometimes filters does not fit perfectly fit the input image. We have two options:

* Pad the picture with zeros(zero- padding) so that it fits
* Drop the part of the image where the filter did not fit. This is called valid padding which keeps only valid part of the image.

Non-Linearity (Re-LU)

Re-LU stands for Rectified Linear Unit for non-linear operation. The output is **f(x) = max(0, x)**

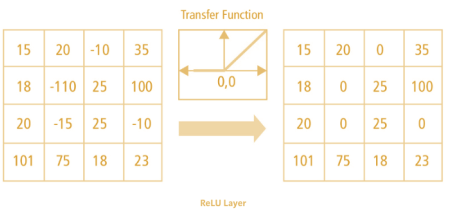


Fig 7: Re-LU operation

There are another non-linear functions such as tan(h) or sigmoid can be used instead of ReLU. Most of the data scientists uses ReLU since performance wise ReLU is better than other two.

Pooling Layer

Pooling layers’ section would reduce the number of parameters when the images are too large. Spatial pooling also called subsampling or down sampling which reduces the dimensionality of each map but retains the important information. Spatial pooling can be of different types:

* Max Pooling
* Average Pooling
* Sum Pooling

Max pooling take the largest element from the rectified feature map. Taking the largest element can also take the average pooling. Sum of all elements in the feature map call as sum pooling.



Fig 8: max-pooling

Fully Connected Layer

The layer we call as FC layer, we flattened our matrix into vector and feed it into a fully connected layer like neural network.

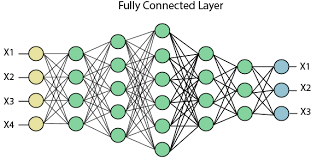


Fig 9: After max pooling flattened FC layer

In the above diagram, feature map matrix will be converted as vector (x1, x2, x3,…..) with the fully connected layers, we combined these features together to create a model. Finally, we have an activation function such as softmax or sigmoid to classify the outputs.

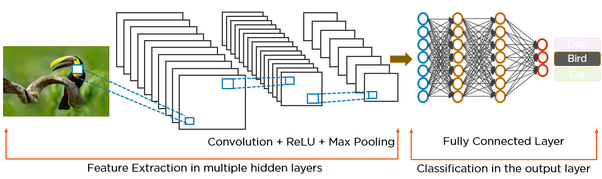


Fig 10: complete CNN architecture

Summary:

1. Provide input image into convolution layer.
2. Perform pooling to reduce dimensionality size.
3. Add as many convolution layers until satisfied.
4. Flatten the output and feed into a fully connected layer (FC layer).
5. Output the class using an activities using an activation function (Logistic Regression cost functions) and images.

**Pseudo code:**

Module 1: Whole System Step

1: Start

Step 2: Get input image from user

Step 3: The image is passed through the trainable module of convolution network.

Step 4: End.

Module 2: Train a CNN step

1: Start

Step 2: Load the dataset containing images.

Step 3: Convert the RGB images into grey scale images and set the channel to 1,

Step 4: Defining the number of classes and assign the labels to images,

Step 5: Split the dataset into training and testing dataset.

Step 6: Define the convolution neural network model.

Step 7: After building a model then start the training.

Step 8: After the training save the model into hdf5 format.

Step 9: End.

**5.1 Environmental settings for running the model**

1. Install necessary Python libraries
2. Run the Program
3. **Testing**

**6.1 Unit Testing**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Date** | **No. of Samples** | **Epochs** | **Test Accuracy** | **Validation Accuracy** |
| 13-03-2020 | 2000 | 15 | 0.8008 | 0.8010 |
| 22-03-2020 | 3000 | 20 | 0.8880 | 0.8889 |
| 29-03-2020 | 4000 | 25 | 0.9400 | 0.9455 |
| 05-04-2020 | 6340 | 30 | 0.9950 | 0.9994 |

During this testing following were occurred and solved:

1. No. of samples: 2000

Epochs: 15

Problem occurred: Memory error

Solution: Increased RAM by 4 GB

1. No. of samples: 3000

Epochs: 20

Problem occurred: Batch size

Solution: Batch size was reduced

1. No. of samples: 4000

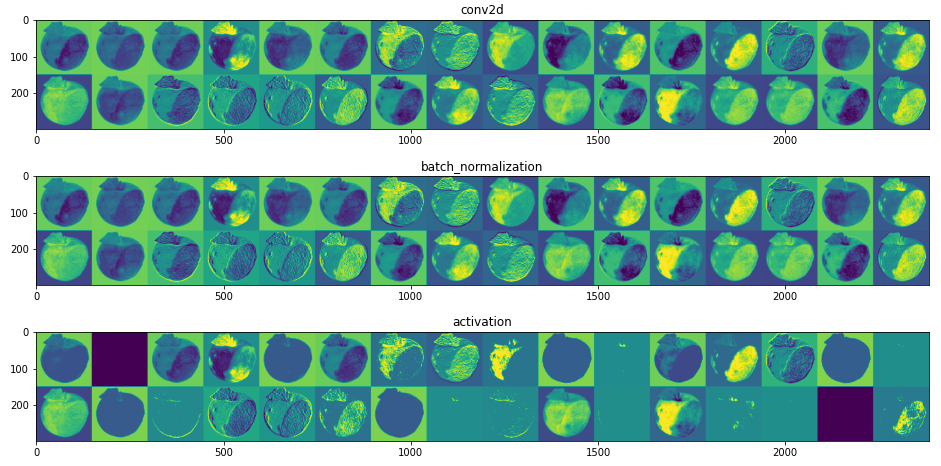
Epochs: 25

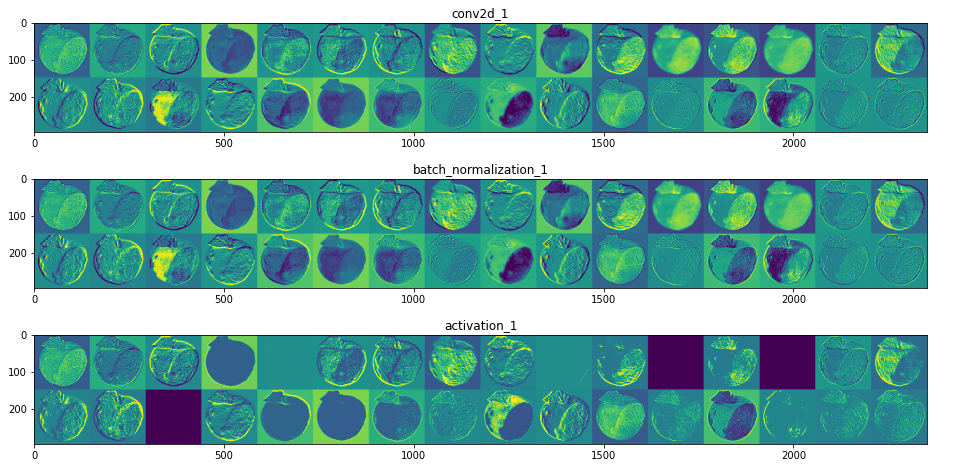
Problem occurred: Accuracy Problem

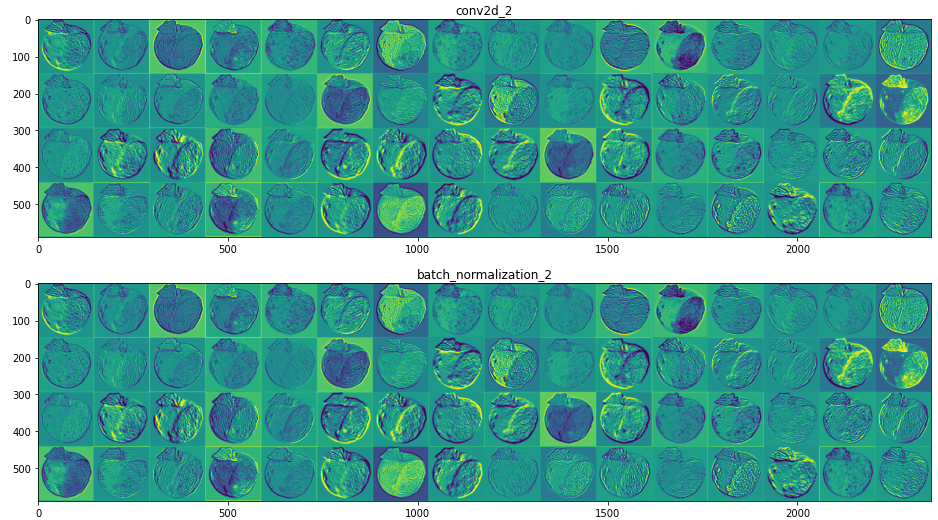
Solution: a) Learning rate was increased

* + 1. Data Augmentation

1. **Performance Analysis**

****

****

****

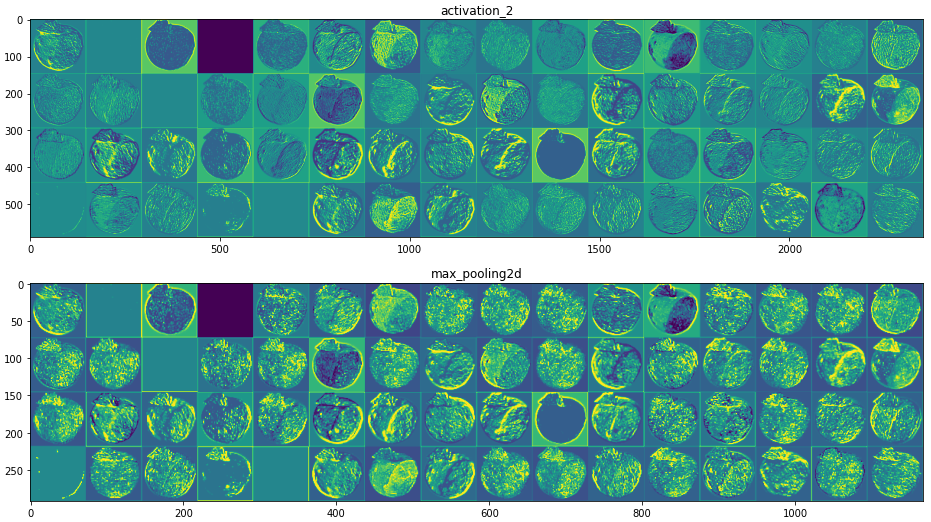
****

Fig 7.1: Feature Mapping

**The feature maps in the above activations are for a rotten input image. Using feature visualization, we can know what a neural network layer and its features are looking for. Using attribution, we can understand how the features impact the output and what regions in the image led the model to the generated output.** What we see in the above plots are the activations or the outputs of each of the 11 layers we chose. The activations or the outputs from the 1st layer (conv2d\_1) don't lose much information of the original input. They are the results of applying several edge detecting filters on the input image. With each added layer, the activations lose visual/input information and keep building on the class/output information. As the depth increases, the layers activations become less visually interpretable and more abstract. By doing so, they learn to detect more specific features of the class rather than just edges and curves. We plotted just 10 out of 314 intermediate layers. We already have in these few layers, activations which are blank/sparse (for ex: the 2 blank activations in the layer activation\_1). These blank/sparse activations are caused when any of the filters used in that layer didn't find a matching pattern in the input given to it. By plotting more layers (especially those towards the end of the network), we can observe more of these sparse activations and how the layers get more abstract.

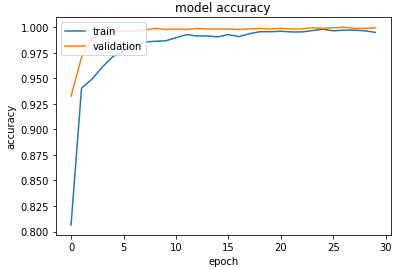
****

Fig 7.3: Training Accuracy vs Validation Accuracy Graph

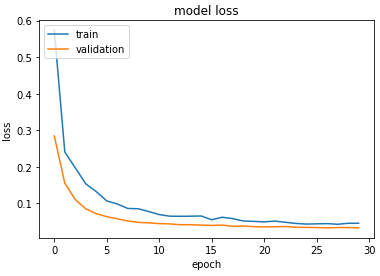
****

Fig 7.4: Training Loss vs Validation Loss Graph

**8. APPLICATIONS**

* **Food industry**

Grading of these minor fruits is considered very important as it can fetch higher price to the producer. Grading also improves packaging, handling. It is very useful for large producers because this system makes it easier to differentiate defected fruits from a bulk quantity.

* **Useful for food science students for studying**

The system can be used by food science students to make further research.

* **Useful for farmers for exporting fruits**

The graded produce fetch better price in the market. Grading helps to develop greater confidence between buyers and producers. Grading improves product uniformity within a particular grade and serves as the basis for price.

**9. PROJECT ETHICS**

As an IT student we believe it’s unethical to,

* Surf the internet for personal and non-class related purpose during classes.
* Make a copy of software for personal or commercial use.
* Make a copy of software for friend.
* Loan CDs of software to friend.
* Download pirated software from the internet.
* Distribute pirated software.
* Buy software on single user license and then install it on multiple computers.
* Share a pirated copy of software.
* Install a pirated copy of software.

**10. REFERENCES**

1. [1] Devrim, Unay. Bernard, Gosselin, “Thresholding based Segmentation and Apple Grading by Machine Vision”, Computers and Electronics in Agriculture, 117-130, January 2005.
2. [2] Suresha M. Shilpa N.A, Soumya, “Apples Grading based on SVM Classifier”, International Journal of Computer Applications(0975-8878) on National Conference on Advanced Computing and Communications – NCACC, April 2012.
3. [3] M. Satpute, S. Jagdale, “Color, Size, Volume, Shape and Texture Feature Extraction Techniques for Fruits: A Review”, International Research Journal of Engineering and Technology (IRJET), Volume: 03 Issue: 02, Feb-2016A
4. [4] PL.Chithra, M.Henila, “Defect detection in the fruit apple using K-means color image segmentation”, International Journal of advance research in computer science, Volume: 08, No: 08, September-October 2017