

IoT based Smart Fridge Application

K.Srinivasa Rao¹, M. Bhanu Sridhar², L.Pavani³
^{1,2}Gayatri Vidya Parishad College of Engineering for Women

Abstract:- The Internet of Things (IoT) can be described as connecting everyday objects like smartphones, Internet TVs, sensors and actuators to the Internet where the devices are intelligently linked together enabling new forms of communication between things and people. IOT devices share the sensor data they collected by connecting to an IOT gateway or other edge device where data is either sent to the cloud to be analyzed or analyzed locally. IoT applications include Smart cities, Smart grids, Industrial Internet, and Connected cars. Rapid improvement in technology tend to use smarter devices in day-to-day life, one such device is refrigerator. In the modular kitchen refrigerator plays a major role in preserving food items. The smart fridge based on (IoT) system detects the shortage of fruits, by transmitting the quantity of available fruits to the users through mobile application. It gives an alert to the users to place an order if the quantity falls below the threshold value. Classification and Regression Algorithms are used in the system to classify the type of fruits, determine the quantity and predict the number of fruits required for future requirement.

Keywords: *Internet of things (IoT), Open-CV, Image Classification, VGG architecture, Linear Regression.*

1 INTRODUCTION

The size and composition of the world population has changed over the last couple of decades, and these trends are projected to continue. Such demographic trends have significant implications for almost all areas of the society, particularly in smart cities, smart house hold devices. IoT is making any objects internally connected in the recent decade and it has been considered as the next technological revolution. The IoT involves extending Internet connectivity beyond standard devices, such as desktops, laptops, smart phones and tablets, to any range of traditionally dumb or non-internet-enabled physical devices and everyday objects. It is the network of devices such as vehicles, and home appliances that contain electronics, software, sensors, actuators, and connectivity which allows these things to connect, interact and exchange data. Embedded with technology, these devices can communicate and interact over the Internet, and they can be remotely monitored and controlled. Here information is sent to the display from android application through IoT cloud. Many real-life applications such as face recognition, autonomous vehicles, object recognition and robotics rely on attempting to mimic the capabilities of the human brain in order to understand images. Computer vision is a field of mathematical analysis of visual data in terms of images of all kinds and this can be a challenging task when applied to the food industry. Visual data of fruit and vegetable expands from binary to hyperspectral images [2, 3, 4, 5, 6]. Advances in imaging techniques have resulted in more sophisticated computer vision leading to its use as an emerging standard for many agricultural applications [7]. Classification of fruit and vegetable is a relatively more complex problem due to the huge variety, for example, irregular intra class shape, colour and texture, and similar interclass shape, colour and texture. Recent state-of-the-art for fruit and vegetable classification and recognition are a combination of feature description and machine learning algorithms on visual data [1, 8, 9, 10, 11]. Significant research has been reported for representation of different characteristics of fruit and vegetable as feature vectors [6, 12]. Despite much research, many challenges need to be overcome to build an effective fruit and vegetable classification system. The technologies implemented using IOT in electrical appliances at homes made it smarter. Smart fridge is connected to the network via a protocol through which it can communicate with the user and make it interactive.

Thus, this paper provides a comprehensive method for identification and prediction of fruits as follows, the web cam is used to collect the data and send them to a host where it is intended to be processed over the internet. Fruit's identification and classification have been done using OpenCV python. And the processed data i.e., present data and predicted data is sent to firebase. Firebase can be accessed by user through MIT app. So, user can view their fruits count through MIT app and get alert notification through mail if fruits count is less than 2. Excel sheet data is used to predict the quantity of fruits required for next week. Mobile application has been developed to monitor the quantity of things in the refrigerator.

2 METHODOLOGY

The method followed in IoT Based SMART FRIDGE is as follows.

2.1 System Design:

- The Web Cam is used to collect the data and send them to a host i.e., pycharm IDE where it is intended to be processed.
- Using Open CV, python and machine learning algorithms, fruits classification and prediction will be done. Classified data will be stored in excel sheet and that data will be used as data set for prediction.
- Mobile application has been developed to monitor the quantity of things in the refrigerator.
- The processed data i.e., present data and predicted data is sent to firebase. Firebase can be accessed by user through mobile application i.e., MIT app.
- So, user can view their fruits count through MIT app and get Alert notification through mail if fruits count is less than 2.

Fig.1 shows the block diagram.

2.2 Block Diagram:

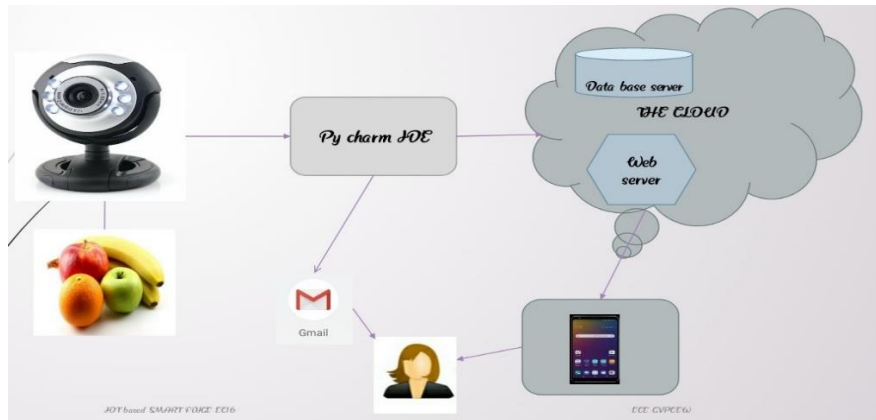


Fig. 1: Block Diagram

2.3 Software tools required: MIT APPInventor, Firebase, Programming Languages Used: Python.

2.3.1 Python: Python is a powerful high-level object-oriented programming language. It can be used to develop desktop GUI applications, and web applications. It has large and robust standard library makes python score over other programming languages.

2.3.2 Installed Packages: Firebase, OpenCV, TensorFlow, Openpyxl, Pandas, Matplotlib, Numpy.

2.3.3 OpenCV: OpenCV is a cross-platform library using which we can develop real-time computer vision applications. It mainly focuses on image processing, video capture and analysis including features like face detection and object detection. Computer Vision can be defined as a discipline that explains how to reconstruct, interrupt, and understand a 3D scene from its 2D images, in terms of the properties of the structure present in the scene. It deals with modelling and replicating human vision using computer software and hardware.

Computer Vision overlaps significantly with the following fields- Image Processing – It focuses on image manipulation. Pattern Recognition – It explains various techniques to classify patterns. Photogrammetry – It is concerned with obtaining accurate measurements from images. Robotics Application- Localization – Determine robot location automatically. Navigation- Obstacles avoidance Assembly (peg-in-hole, welding, painting), Manipulation (e.g. PUMA robot manipulator), Human Robot Interaction (HRI) – Intelligent robotics to interact with and serve people. Medicine Application- Classification and detection (e.g. lesion or cells classification and tumor detection). 2D/3D segmentation, 3D human organ reconstruction (MRI or ultrasound), Vision-guided robotics surgery. Industrial Automation Application - Industrial inspection (defect detection), Assembly, Barcode and package label, reading, Object sorting, Document understanding (e.g. OCR). Security Application - Biometrics (iris, finger print, face recognition), Surveillance – Detecting certain suspicious activities or behaviors. Transportation Application - Autonomous vehicle, Safety, e.g., driver vigilance monitoring. Features of OpenCV Library- Using OpenCV library, you can –Read and write images, Capture and save videos, Process images (filter, transform), Perform feature detection, Detect specific objects such as faces, eyes, cars, in the videos or images. Analyze the video, i.e., estimate the motion in it, subtract the background, and track objects in it. OpenCV was originally developed in C++. In addition to it, Python and Java bindings were provided. OpenCV runs on various Operating Systems such as windows, Linux, OSx, FreeBSD, Net BSD, Open BSD, etc.

2.4 Object Detection Algorithms: SSD MOBILENET V3: This is one of the famous deep learning algorithms for object detection. SSD means single shot multi-box detector. It has three stages. They are feature extraction stage, detection heads and Non-max suppression.

2.4.1 SSD Architecture:

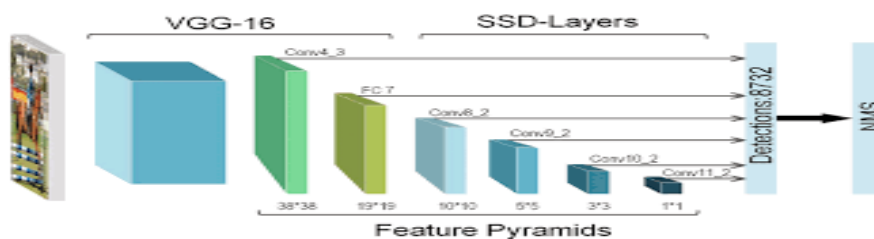


Fig. 2: SSD ARCHITECTURE

2.4.2 VGG16 (Feature extraction stage):



Fig 3: VGG extraction stage

2.4.3 VGG Architecture: VGG is one of the high performing image classification network. This is a feature extraction stage. It is purely convolutional stage. It is a stack of convolutional networks that generate feature maps. This stage in this network is to build a tower of feature maps. The goal of the feature maps is to encode semantic information that is localized in different regions inside the image which can then be used for making box proposals.

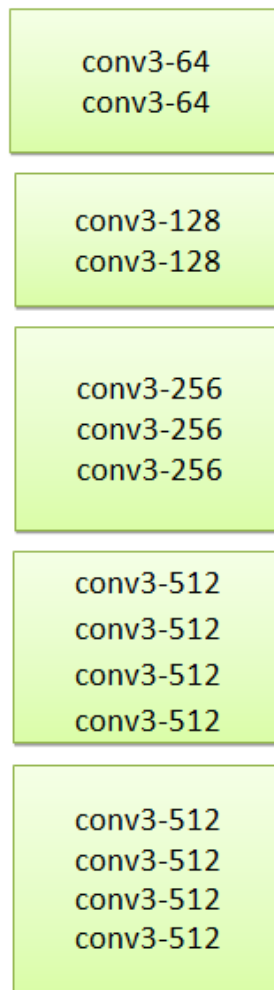


Fig 4: VGG

2.4.4 Detection stage: The detection stage consists of 6 convolution layers but their goal is not to extract useful semantic features about regions of an image. Here, the main goal of this stage is to convert the feature maps into prediction about boxes. The detection generator consists of 2 pieces. Multiway classification and box regression.

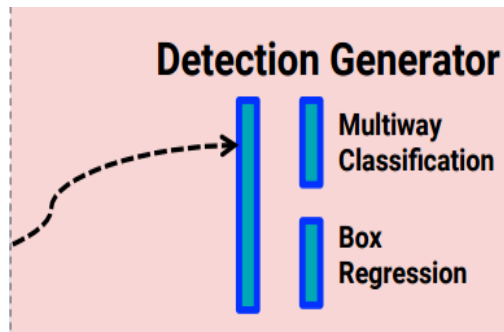


Fig 5:Detector Generator

These SSD convolution layers makes the 8732 multi box predictions for each object.The top 200 multi box predictions will be given as output of this stage based on their confidence score.The confidence score is calculated from IOU (Intersection Over Union) method.

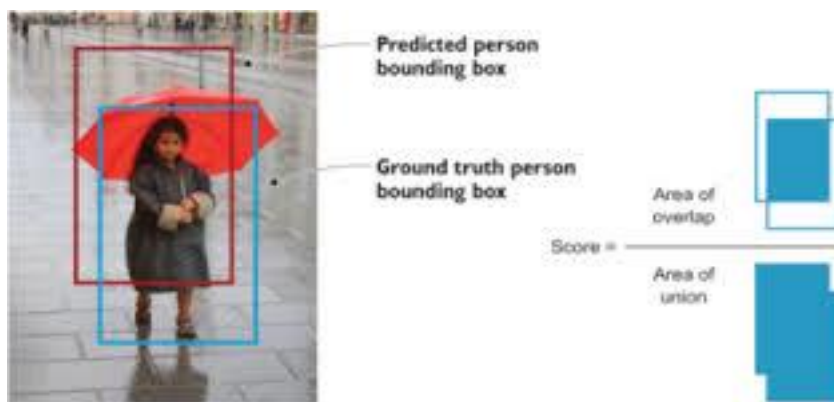


Fig 6: IOU Method

2.4.5 Non-max suppression stage: It is a common algorithm used for cleaning up when multiple boxes are predicted for the same object. To get a detector with a good performance, this algorithm is required.It discards all boxes with confidence score less than or equal to 0.6. Then, it will pick the box with the largest confidence score output as prediction.After that, it will discard the remaining box with confidence score greater than or equal to 0.5.



Fig 7: NON-Max Suppression

2.4.6 COCO Dataset:The COCO (common object in context) is a large-scale object detection, segmentation and captioning dataset. As hinted by the name, images in COCO dataset are taken from everyday scenes thus attaching “context” to the objects captured in the scenes. We can put an analogy to explain this further. Let’s say we want to detect a person object in an image. A non-contextual, isolated image will be a close-up photograph of a person. Looking at the photograph, we can only tell that it is an image of a person. However, it will be challenging to describe the environment where photograph was taken without having other supplementary images that capture not only the person but also the studio or surrounding scene.COCO was an initiative to collect natural images, the images that reflect everyday scene and provides contextual information. In everyday scene, multiple

objects can be found in the same image and each should be labeled as a different object and segmented properly. A machine learning practitioner can take advantage of the labeled and segmented images to create a better performing object detection model.

2.5 Prediction Algorithm: A Regression based on neural network is used for prediction.

2.5.1 Neural Network: A neural network is a series of algorithms that endeavors to recognize underlying relationships in a set of data through a process that mimics the way the human brain operates.

2.5.2 A Multi-Layered Perceptron (MLP): is one of the most common neural network models used in the field of deep learning. Often referred to as a “vanilla” neural network, an MLP is simpler than the complex models of today’s era. However, the techniques it introduced have paved the way for further advanced neural networks. The multilayer perceptron (MLP) is used for a variety of tasks, such as stock analysis, image identification, spam detection, and election voting predictions. The basic structure: A multi-layered perceptron consists of interconnected neurons transferring information to each other, much like the human brain. Each neuron is assigned a value. The network can be divided into three main layers. Input layer: this is the initial layer of the network which takes in an input which will be used to produce an output. Hidden layer(s): The network needs to have at least one hidden layer. The hidden layer(s) perform computations and operations on the input data to produce something meaningful. Output layer: The neurons in this layer display a meaningful output.

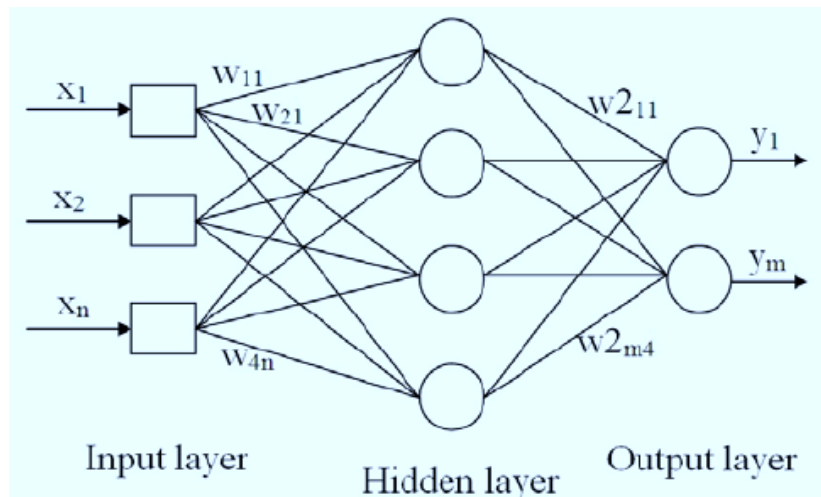


Fig 8: Multilayer perception

2.5.3 Connections: The MLP is a **feedforward neural network**, which means that the data is transmitted from the input layer to the output layer in the forward direction. The connections between the layers are assigned weights. The weight of a connection specifies its importance. This concept is the backbone of an MLP’s learning process. While the inputs take their values from the surroundings, the values of all the other neurons are calculated through a mathematical function involving the weights and values of the layer before it. For example, the value of the h_5 node could be: $h_5 = h_1.w_8 + h_2.w_9$. Backpropagation: It is a technique used to optimize the weights of an MLP using the outputs as inputs. In a conventional MLP, random weights are assigned to all the connections. These random weights propagate values through the network to produce the actual output. Naturally, this output would differ from the expected output. The difference between the two values is called the error. Backpropagation refers to the process of sending this error back through the network, readjusting the weights automatically so that eventually, the error between the actual and expected output is minimized. In this way, the output of the current iteration becomes the input and affects the next output. This is repeated until the correct output is produced. The weights at the end of the process would be the ones on which the neural network works correctly.

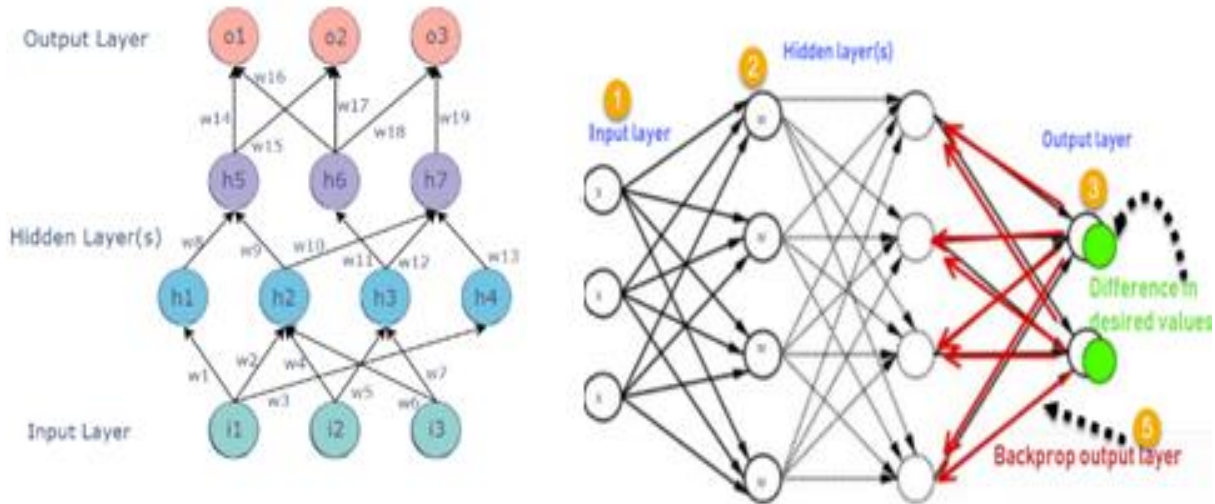


Fig 9: BACK PROPAGATION

2.5.4 Linear regression: Linear regression is one of the easiest and most popular Machine Learning algorithms. It is a statistical method that is used for predictive analysis. Linear regression makes predictions for continuous/real or numeric variables such as **sales, salary, age, product price**, etc. Linear regression algorithm shows a linear relationship between a dependent (y) and one or more independent (x) variables, hence called as linear regression. Since linear regression shows the linear relationship, which means it finds how the value of the dependent variable is changing according to the value of the independent variable. The linear regression model provides a sloped straight line representing the relationship between the variables. Consider the below image:

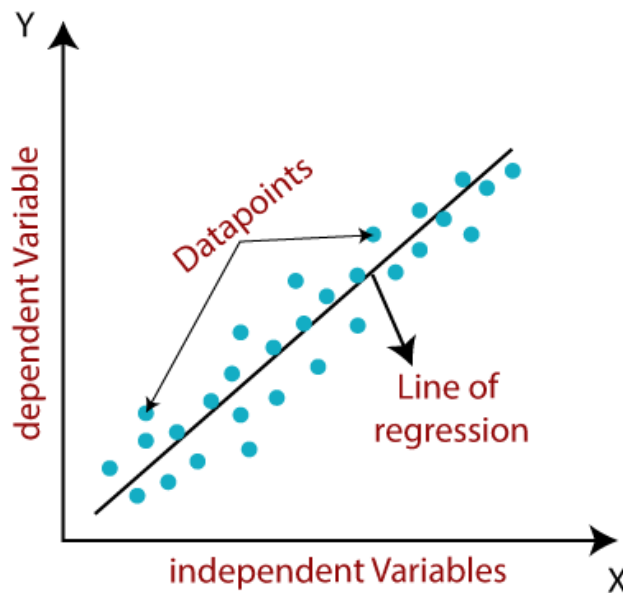


Fig 10: Graph of Linear Regression Model

Mathematically, we can represent a linear regression as:

$$y = a_0 + a_1x + \epsilon$$

Here, Y = dependent variable, X = independent variable, a_0 = intercept of the line, a_1 = Linear regression coefficient, ϵ = random error.

The values for x and y variables are training datasets for Linear Regression model representation. Types of linear regression: Linear regression can be further divided into two types of the algorithm:

2.5.5 Simple Linear Regression: If a single independent variable is used to predict the value of a numerical dependent variable,

then such a Linear Regression algorithm is called Simple Linear Regression.

2.5.6 Multiple Linear regression: If more than one independent variable is used to predict the value of a numerical dependent variable, then such a Linear Regression algorithm is called Multiple Linear Regression.

2.5.7 Linear Regression Line: A linear line showing the relationship between the dependent and independent variables is called a regression line. A regression line can show two types of relationship: Positive Linear Relationship: If the dependent variable increases on the Y-axis and independent variable increases on X-axis, then such a relationship is termed as a Positive linear relationship.

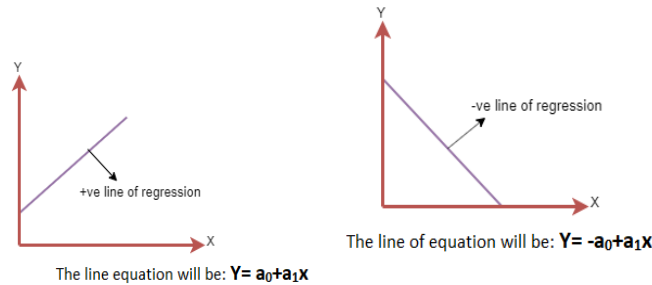


Fig 11: Positive and NEGATIVE Linear Relationship

2.5.8 Negative Linear Relationship: If the dependent variable decreases on the Y-axis and independent variable increases on the X-axis, then such a relationship is called a negative linear relationship. Finding the best fit line: When working with linear regression, our main goal is to find the best fit line that means the error between predicted values and actual values should be minimized. The best fit line will have the least error. The different values for weights or the coefficient of lines (a_0, a_1) gives a different line of regression, so we need to calculate the best values for a_0 and a_1 to find the best fit line, so to calculate this we use cost function.

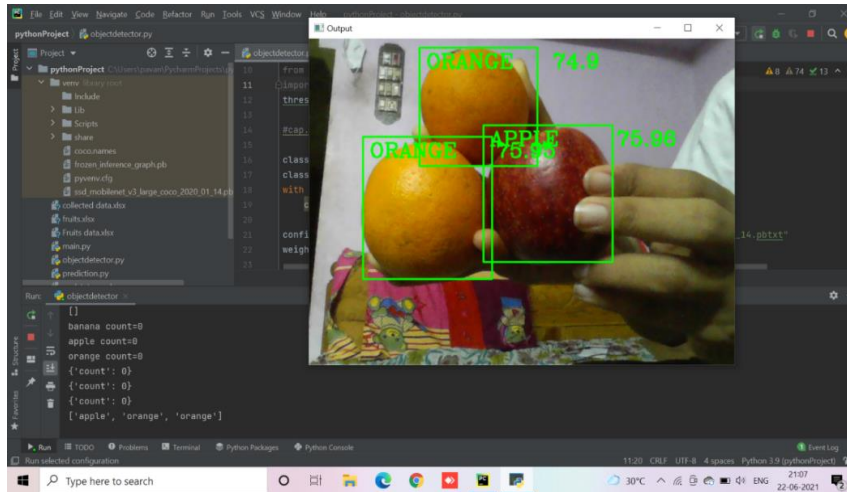
2.5.9 Cost function: The different values for weights or coefficient of lines (a_0, a_1) gives the different line of regression, and the cost function is used to estimate the values of the coefficient for the best fit line. Cost function optimizes the regression coefficients or weights. It measures how a linear regression model is performing. We can use the cost function to find the accuracy of the mapping function, which maps the input variable to the output variable.

2.6 MITAppInventor: The android app was developed using the Massachusetts Institute of Technology App Inventor that was originally provided by Google. MIT App Inventor is a block-based programming tool that allows everyone to program and design apps for Android based devices. As it is blocks-based programming, the programmer does not have to remember the coding or syntax and does not have to worry about syntax errors. This efficiently saves time in writing and debugging the code. The MIT App Inventor is highly user-friendly however it is a web-based service hence a Google account is required in order to access.

2.6.1 Firebase: The Firebase Realtime Database is a cloud-hosted database. Data is stored as JSON and synchronized in Realtime to every connected client. When you build cross platform apps with our iOS, Android, and JavaScript Store and sync data with our NoSQL cloud database. Data is synced across all client in Realtime, and SDKs, all of your clients share one Realtime Database instance and automatically receive updates with the newest data. Firebase offers two cloud-based, client-accessible database solutions that support Realtime data syncing. Realtime Database is Firebase's original database. It's an efficient, low-latency solution for mobile apps that require synced states across clients in Realtime. Cloud Fire store is Firebase's new flagship database for mobile app development. It improves on the successes of the Realtime Database with a new, more intuitive data model. Cloud Fire store also features richer, faster queries and scales better than the Realtime Database.

3 RESULTS

The following figure shows the fruit detection, localization and labelling based on their classification using webcam, Open CV, python and machine learning algorithms.



3.1 Android App Screens:

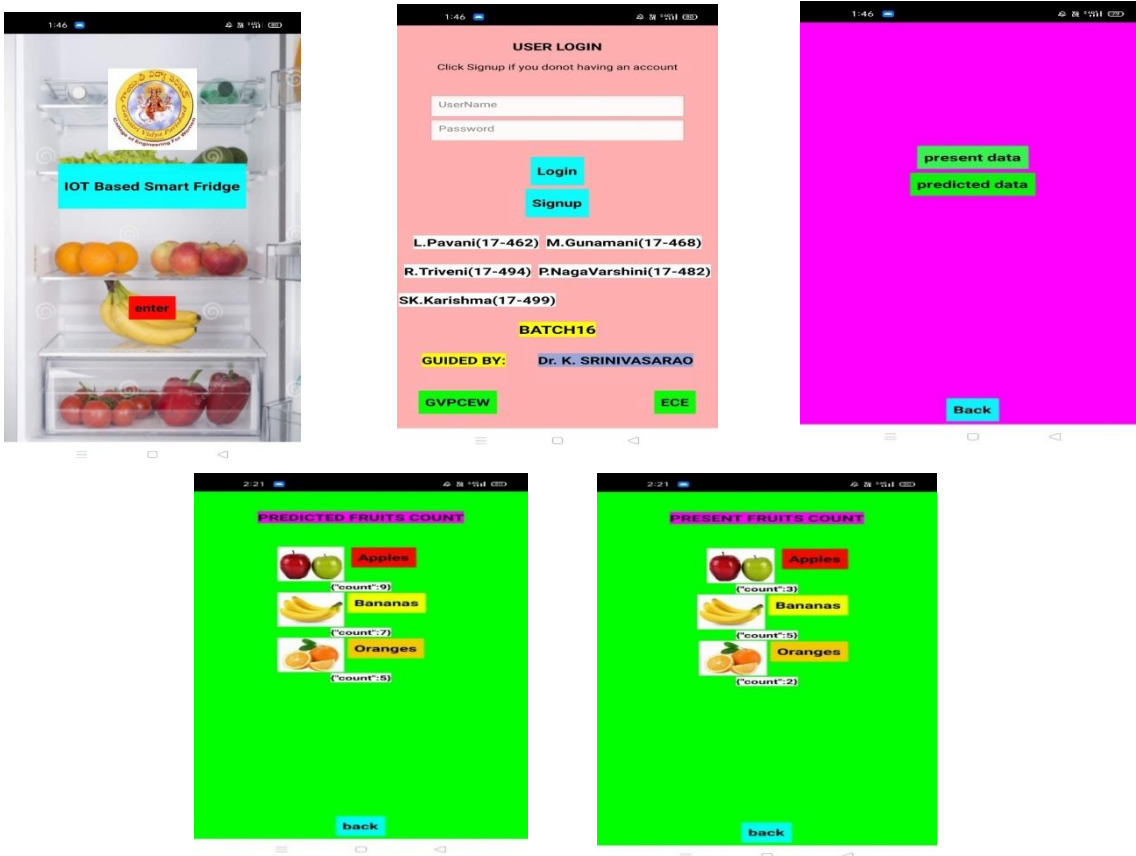


Fig 12: Pycharm platform and Android app screens

The above figures show fruits identification and labelling on Pycharm platform. Apple is identified with accuracy of 75.96% and Oranges with 75%. The designed mobile application screens are shown. First screen consists of Entry Page Of Designed APP. If we press the button enter it will open the second screen of the application. Second screen consists of the options to login and signup. If we enter the correct password and press login it enters into next screen else it shows wrong user name and password. New user must sign up for accessing the next screen. In screen 3 we have the buttons for accessing present and predicted data.

3.2 Excel Sheet: By using python code import the fruits data into excel sheet using openpyxl command in python then load the data into excel sheet. This data is used to predict fruits count required for the Future. This figure shows storage of the processed data i.e., present data and predicted data in Fire base as shown in the below Fig 13.

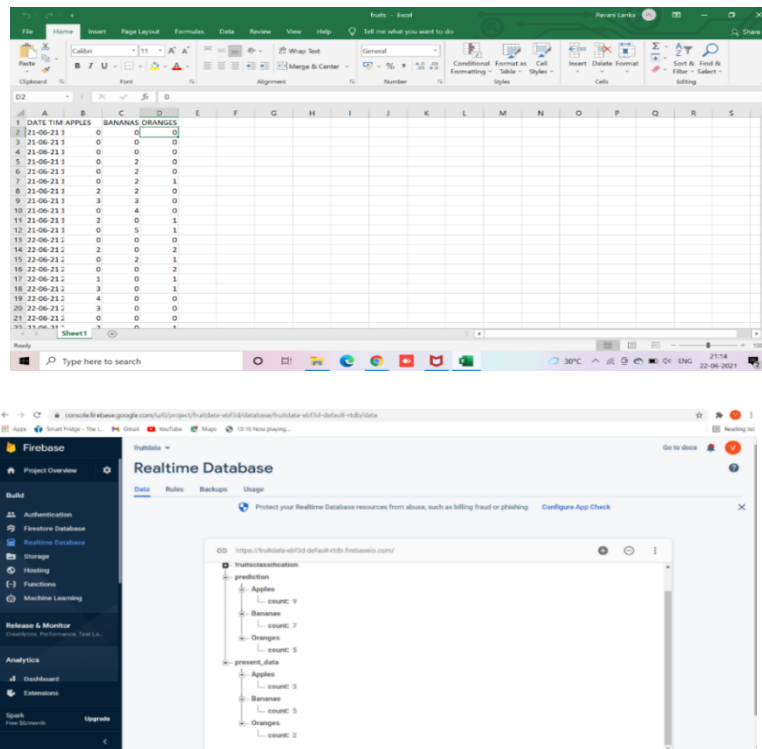


Fig 13: Excel sheet for prediction and Firebase window

User will get the alert notification through Gmail. This alerting notification is done by using the library smtplib. From smtplib, the packages MIMEMultipart, MIMEText, Header, MIMEBase have to be imported to send the alert notification through Gmail. Here, we have to give the user mail address for whom the notification has to be sent and we have to provide an email address which is used to send the mail to the user mail address. Here, the above figure shows the alert notification which has been sent to the user mail id by giving the message as the fruit count is less than the threshold value, so they must be bought.

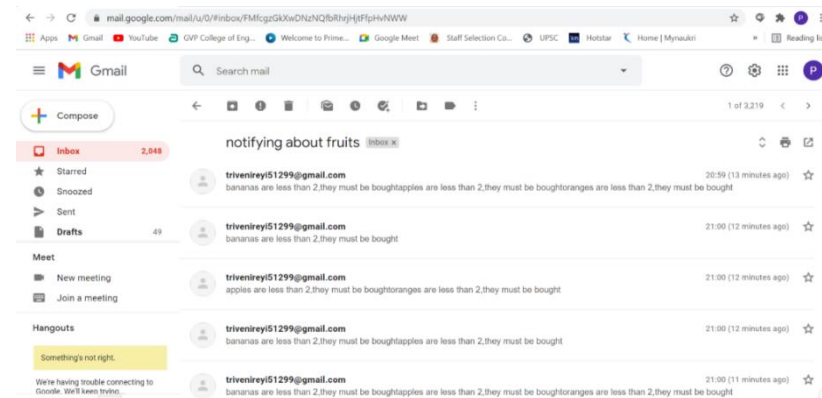


Fig 14: Mail notification

4 CONCLUSION

The dynamic rapidly changing and technology-rich digital environment enables the provision of added value applications that exploit a multitude device contributing services and information. As IOT techniques mature and become ubiquitous, emphasis is put upon approaches that allows things to become smarter, more reliable and more autonomous. The IOT based smart fridge is developed by using machine learning algorithms and software. The results show that fruits identification and labelling on Pycharm platform and Apple is identified with accuracy of 75.96% and Oranges with 75%. The Smart Refrigerator module is able to remotely notify the user and control the functions inside the refrigerator. It gives the user current availability of fruits in the fridge through the android application MIT app inventor, prediction of the fruits required and the alert notification regarding the shortage of fruits through Gmail.

4.1 Future Scope

The concept behind the IOT based smart fridge can be applied in various applications like whether the fruits are fresh or rotten, to determine whether the fruits are ripened or not in farms, to find the productivity of trees by counting the fruits on tree, to sort

the fruits, to count the number of fruits in farms and industries. Further, it can be implemented using hardware components using raspberry pi and fruits images are captured by using pi cam. This reduces the man power as well as time. More area can be converted using more sensors. Device size can be reduced effectively.

5 REFERENCES

- [1] A. Bhargava, A. Bansal, "Fruits and vegetables quality evaluation using computer vision" A review, *Journal of King Saud University - Computer and Information Sciences* doi:10.1016/j.jksuci.2018.06.002. URL <https://doi.org/10.1016/j.jksuci.2018.06.002>
- [2] F. Kurtulmus, W. S. Lee, A. Vardar, "Immature peach detection in colour images acquired in natural illumination conditions using statistical classifiers and neural network", *Precision agriculture* 15 (1) (2014) 57-79.
- [3] A. Gongal, S. Amatya, M. Karkee, Q. Zhang, K. Lewis, "Sensors and systems for fruit detection and localization" A review, *Computers and electronics in agriculture* 116 (2015) 8-19.
- [4] K. Yamamoto, W. Guo, Y. Yoshioka, S. Ninomiya, "On plant detection of intact tomato fruits using image analysis and machine learning methods, *Sensors*" 14 (7) (2014) 12191-12206.
- [5] A. Gijsenij, T. Gevers, J. Van De Weijer, "Computational color constancy: Survey and experiments, *IEEE Transactions on Image Processing*" 20 (9) (2011) 2475-2489. doi:10.1109/TIP.2011.2118224.
- [6] A. Nasirahmadi, S. H. Miraei Ashtiani, "Bag-of-Feature model for sweet and bitter almond classification, *Biosystems Engineering*" 156 (2017) 51-60. doi:10.1016/j.biosystemseng.2017.01.008. URL <http://dx.doi.org/10.1016/j.biosystemseng.2017.01.008>
- [7] H. Kuang, C. Liu, L. L. H. Chan, H. Yan, "Multi-class fruit detection based on image region selection and improved object proposals, *Neurocomputing*".
- [8] Y.-D. Zhang, Z. Dong, X. Chen, W. Jia, S. Du, K. Muhammad, S.-H. Wang, Image based fruit category classification by 13-layer deep convolutional neural network and data augmentation, *Multimedia Tools and Applications* (2017) 1-20.
- [9] P. Wan, A. Toudeshki, H. Tan, R. Ehsani, A methodology for fresh tomato maturity detection using computer vision, *Computers and Electronics in Agriculture* 146 (February 2017) (2018) 43-50. doi:10.1016/j.compag.2018.01.011. URL <https://doi.org/10.1016/j.compag.2018.01.011>.
- [10] H. Li, Bairong and Long, Yan and Song, Detection of green apples in natural scenes based on saliency theory and Gaussian curve fitting, *International Journal of Agricultural and Biological Engineering* 11 (1) (2018) 192-198. doi: 10.25165/j.ijabe.20181101.2899.
- [11] A. Wajid, N. K. Singh, P. Junjun, M. A. Mughal, Recognition of Ripe, Unripe and Scaled Condition of Orange Citrus Based on Decision Tree Classification, in: *International Conference on Computing, Mathematics and Engineering Technologies*, 2018, pp. 2-5.
- [12] T. and others Ishikawa, T and Hayashi, A and Nagamatsu, S and Kyutoku, Y and Dan, I and Wada, T and Oku, K and Saeki, Y and Uto, T and Tanabata, Classification Of Strawberry Fruit Shape By Machine Learning, *International Archives of the Photogrammetry, Remote Sensing & Spatial Information Sciences XLII* (June) (2018) 4-7.