Server-Based Vending Machine in the context of Nepal

Karan K.C.
Electronics and Communication Engineering
Kantipur Engineering College
Tribhuwan University
Lalitpur, Nepal
kirajkc10@gmail.com

Riken Maharjan
Electronics and Communication Engineering
Kantipur Engineering College
Tribhuwan University
Lalitpur, Nepal
rikenm44@gmail.com

Rudra Narayan Jha
Electronics and Communication Engineering
Kantipur Engineering College
Tribhuwan University
Lalitpur, Nepal
rudrajha56@gmail.com

Sanjeev Jung Chaudhary
Electronics and Communication Engineering
Kantipur Engineering College
Tribhuwan University
Lalitpur, Nepal
sanjeevjungchaudhary@gmail.com

***Abstract*—This paper aims to propose the concept of a "Server-Based Vending Machine" in the context of Nepal. As in Nepal, vending machines are just available in recent years and also there are only a few in numbers as all are mostly traditional types of a vending machines. Currently, the world is becoming more and more digital and development of smart technologies traditional vending machine is becoming less and less usable or efficient. To improve and reinforce the services of traditional vending machines, this system provides an intelligent solution to facilitate their functions. In this approach, we put forward the design of server-based (remote) monitoring of system and product's data. The server and vending machine are connected via NodeMCU and duplex communication is established between them. All the data of ordered items are sent to the server while updated data of product price by owner are sent to a vending machine. Before sending data to the server it is temporarily stored in Arduino DUE and sent only if the transaction is confirmed. According to product data, the product's cost, quantity, and quality are maintained. When items in the vending machine are going to be out of stock i.e. less than 2 in quantity in the vending machine, it alerts the owner through the server so that owner can restock before time. A customer has the availability of both cash or card as payment methods. As for cash payment, the system uses an IR sensor to detect cash so that cash is pulled inside, and once cash is completely pulled inside Raspberry PI and PI camera performs image processing to detect and validate the amount of cash. If cash is not sufficient system asks to insert more cash. The system showed an confidence level of about 95% to 98% for cash detection while the fps of Raspberry PI was approximate 0.8. And for card payment, an RFID card is used. The system verifies the card tag id and balance in it and deducts the amount from it. There is serial communication between Arduino DUE and Raspberry PI at a baud rate of 9600 as well as Arduino DUE and NodeMCU at a baud rate of 115200. It makes it easier for a customer by providing a facility to vend multiple items at a time. Once the whole transaction is completed items are dropped in a container via controlled motorized action. The ultimate goal is to produce a user-friendly vending machine to enhance the customer purchasing experience as well as easily manageable by an owner.**

***Index Terms*—Server-based Monitoring, Cash Payment, Card Payment, Image Processing, Serial Communication, Non-Maximum Suppression(NMS).**

I. Introduction

A. Background and Problem Statement

A vending machine is an automated machine that provides items such as snacks, beverages, and lottery tickets to consumers after money, a credit card, or a specially designed card is inserted into the machine. The traditional vending machine works as when sufficient coin is inserted then customer selects an item by using keypads or buttons and the item is dropped down to a receiving window.

Vending machines exist in many countries, and in more recent times, specialized vending machines that provide less common products compared to traditional vending machine items have been created. Today's vending machines are evolving to be more than just vending machines [1]. With high-definition displays running rich graphics, the ability to interact with the customer, and more, enables continuing innovation to benefit both the customer and virtually everyone in the vending industry. Along with the coming of the Internet of Things (IoT) age and its increasing prevalence in our daily lives [2].

In Nepal, only a few vending machines can be seen in recent years and they are of a traditional type. Lalitpur city unveils Nepal's first seed vending machine in Dec 2019. Through the vending machine, farmers can get 36 different varieties of seeds of vegetables and flowers. There are other vending machines for a sanitary pad, beverages, tea, coffee, and so on in Nepal.

As for the common or traditional vending machine, running out of products is bound to happen in vending machines, especially if you have an infrequent restocking schedule. There is the difficulty of changing product prices if needed by the owner. The traditional vending machine uses only cash or coin payment system.

B. Objectives

The main objective is to produce a system that will have server-based monitoring of the system and a user-friendly environment. The following targets must be fulfilled to reach this goal:

- To develop server end.
- To detect and validate cash amount.
- To send and receive data from the server.
- To alert when item quantity is less.
- To scan and verify the card.
- To order multiple items at a time.

II. Related Works

These days numerous research and literature survey has been carried out on vending machine and its advancement. A vending machine that operates in response to a coin or card inserted by a user is designed [3], [4]. Likewise, the electronic chip, capable of identifying the fingerprints of each user of the automatic vending machine was utilized to memorize them in memory cells and subsequently comparing certain electronic scanning points of their memorized image for electronic identification, when checking the residual credit and/or selection of the desired product [5]. In the field of IoT, a system is designed to monitor the health where data gathered is stored in the cloud and can be viewed in LCD in real-time data [6]. There are technologies that have been used previously for object detection using SSD: Single Shot multibox Detector [7], convolution neural networks for mobile vision applications [8] and recognition and classification of the objects in the images, as well as locating of the objects within it, drawing a bounding box around the appropriate way [9].

III. Methodology

The overall block diagram of the system is shown in Fig.1. The system is divided into three major parts i.e Vending Machine, Cash Recognition, and Server. Vending Machine part consists of Arduino Due as the central microprocessor that is connected to various devices i.e. LCD display to display available items as well as ordered items list, Keypad as an input source for a user, RFID module to detect the card, and L298N motor driver to control DC motor so to drop items. DC motor is operated only for certain interval which is controlled by motor driver using PWM signal. Arduino Due has 12 PWM pins. And L298N controls one DC motor using three pins i.e. input 1 and input 2 to control the direction of the motor and enable the pin to control speed. PWM is provided to enable the pin. PWM allows adjusting the average value of the voltage that is going to the DC motor by turning on and off the power at a fast rate. The average voltage depends on the duty cycle, so when analog output is 0 then the duty cycle is 0% means no power is supplied and when analog output is 255 then the duty cycle is 100% means full voltage is supplied.

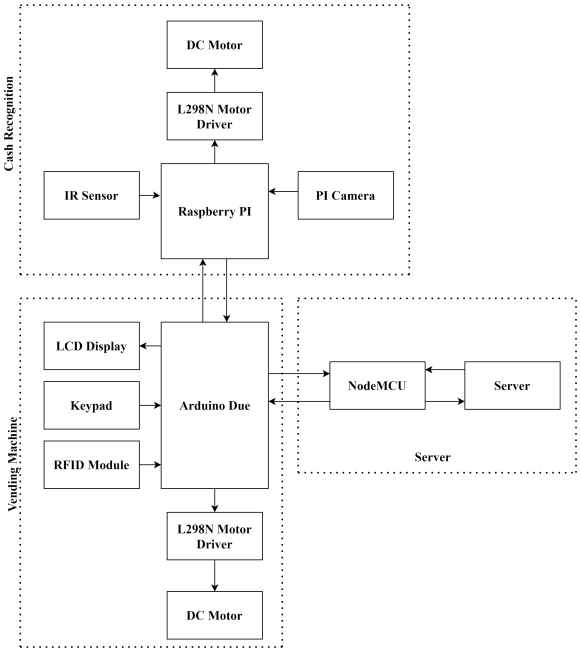

Fig. 1. Block Diagram of System.

Cash Recognition part includes Raspberry PI as the central processor that takes IR sensor's signal as input to control DC motor via L298N motor driver in order to pull cash inside, and PI camera as to take images of cash to detect, locate and validate its amount. On the server-side, NodeMCU acts as a bridge between the vending machine and the server. Bidirectional data communication occurs between the vending machine and the server through NodeMCU.There is serial duplex communication between Raspberry PI and Arduino Due at 9600 baud rate along with NodeMCU and Arduino Due at 115200 baud rate.

The system comprises various software. Arduino Due's controlling program was developed in Arduino IDE, version 1.8.9. In Raspberry PI, python 3.6.8, TensorFlow 1.14, openCV 3.4.2, and NumPy 1.16.4 were used for cash detection. The backend of the server was developed using Django 2.1.5, MySQL client 1.4.6, and XAMPP 3.2.4, whereas the frontend

using Html, CSS, and Javascript.

A. Displaying and Ordering Items

Flowchart 2, shows the process for ordering items. On the LCD, items list with the latest updated price and quantity are displayed. A customer orders by selecting an item using a keypad and inputs the item amount. There are choices to add, view, or cancel orders. If the customer cancels the order then the whole transaction is terminated and LCD, as well as saved data, is reset. In case the customer wants to view the record of the reserved product, he/she can select the view button which displays every item's name and total price. When they confirm their order, the details of ordered items are saved within Arduino Due and moves towards the billing process.

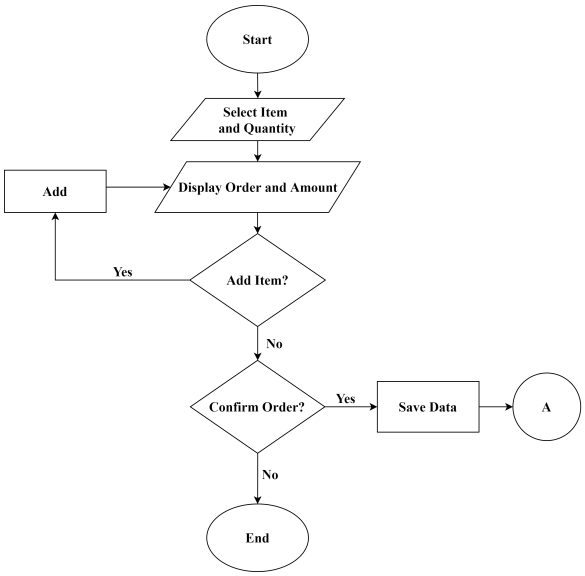

Fig. 2. Flowchart of Ordering System.

B. Billing Process

Flowchart 3, shows the billing process after the order is confirmed. Once an order is confirmed, the system displays the total amount to be paid and provides options for payment i.e. cash or card.

1) Cash Payment

If cash is selected for payment then the system asks to enter cash in the cash slot. The vending machine checks presence of the cash implementing IR sensor. Once cash is inserted, Raspberry PI operates the motor to pull the cash inside. The Raspberry PI camera captures real-time images of cash for detection and validation. It checks whether the amount is enough or not. If it isn't enough then the system asks to insert more cash until provided cash is sufficient.

2) Card Payment

In the case of a card, the RFID module reads the card and verifies its authenticity and amount in it [10]. Provided that there isn't enough balance, the entire transaction is exited and the system is reinitialized. However, if there is a sufficient amount on the card then the system deducts the total bill amount from it.

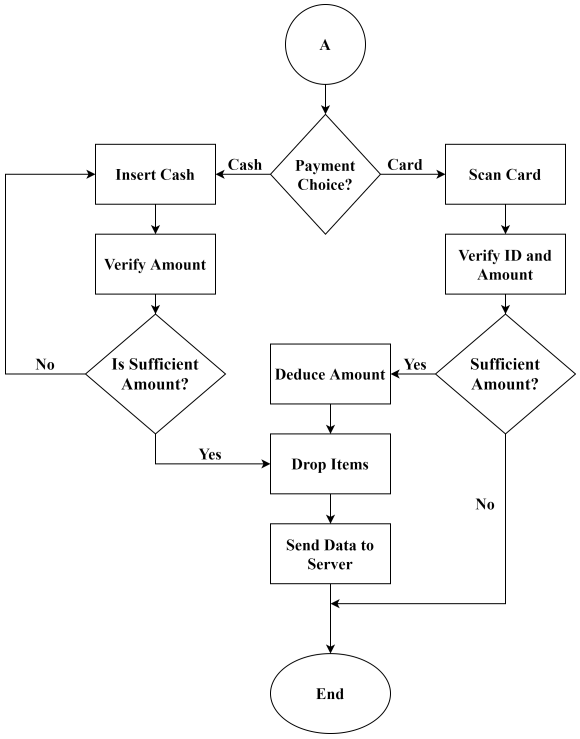

Fig. 3. Flowchart of Payment System.

C. Dropping Items

Once payment is done either by cash or card, the system drops ordered items by motorized action into the outlet window so that the customer can collect products.

D. Server

Once the whole ordering and billing process is completed, then item's data stored in Arduino Due is sent to NodeMCU via serial communication and then into the server. The owner can monitor the whole transaction remotely and update the price of items remotely through the server [11]. In the vending machine, NodeMCU continuously provides data from the server to be displayed on the home screen of LCD. Also, NodeMCU checks the number of items in a vending machine non stop, if the number is less than 2 it alerts the owner via server. Once the item is restocked and the owner updates quantity in server alert message vanishes.

E. Cash Detection

1) Preparing the Training Data

Flowchart 4, shows the process of custom trained models. Out of the total 819 images, 273 of each cash (i.e. Rs.5, Rs.10, and Rs.20), 655 were used

as training images and 164 as test images. After images are labeled, it will have a respective .xml file for each image. The .xml format is not suitable directly for training our model in Tensorflow, we have to convert the .xml files into .csv and then convert the .csv files into TFRecords.

- convert the .xml files into .csv files.
- convert the .csv files into TFRecords.
- generate the label map file for the training.

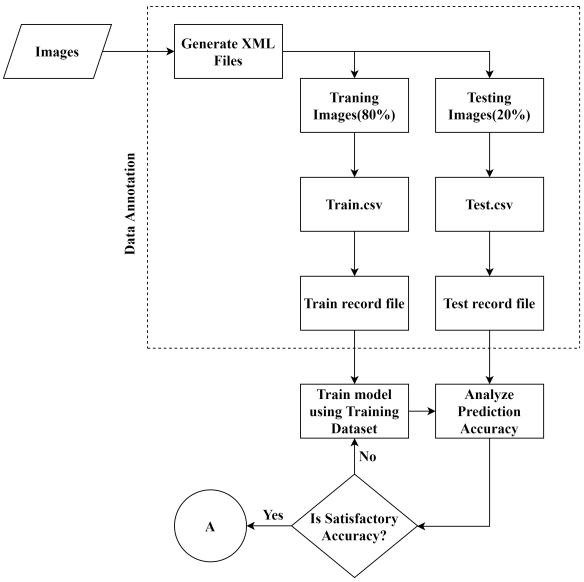

Fig. 4. Flowchart of Custom Trained Model.

2) Choosing our Pre-Trained Model and Training

Here we choose "SSD MobileNetv2 quantized" which was previously trained in the MS COCO dataset . The data pool is initially divided into training and testing groups (ratio of 8:2). Then for training, the model individually studies all training data, extracts their correspondingly unique features, and uses them as a reference to classify the test images. Here MobileNetV2 does all the feature extracting .

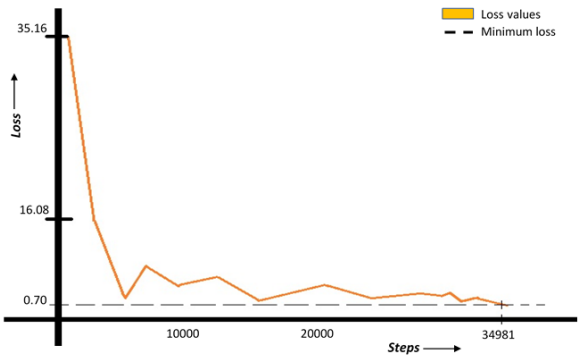

Fig. 5. Loss Graph of Trained Model.

3) Evaluation of the Trained Model

The evaluation of the trained model is done using loss value during its training [i.e. loss is a number indicating how bad the model's prediction was on a single example]. The loss graph model is shown in Fig.5. If the model's prediction is perfect, the loss is zero, otherwise, the loss is greater. The goal of training a model is to find a set of weights and biases that have low loss, on average, across all examples. Hence comparing the loss value for every step of the training phase, training the model is stopped at a step value 34981 and satisfactory loss value of 0.7. An epoch indicates the number of passes of the entire training dataset the machine learning algorithm has completed. 120 epoch is used as a save point during the training phase.

4) Implementation

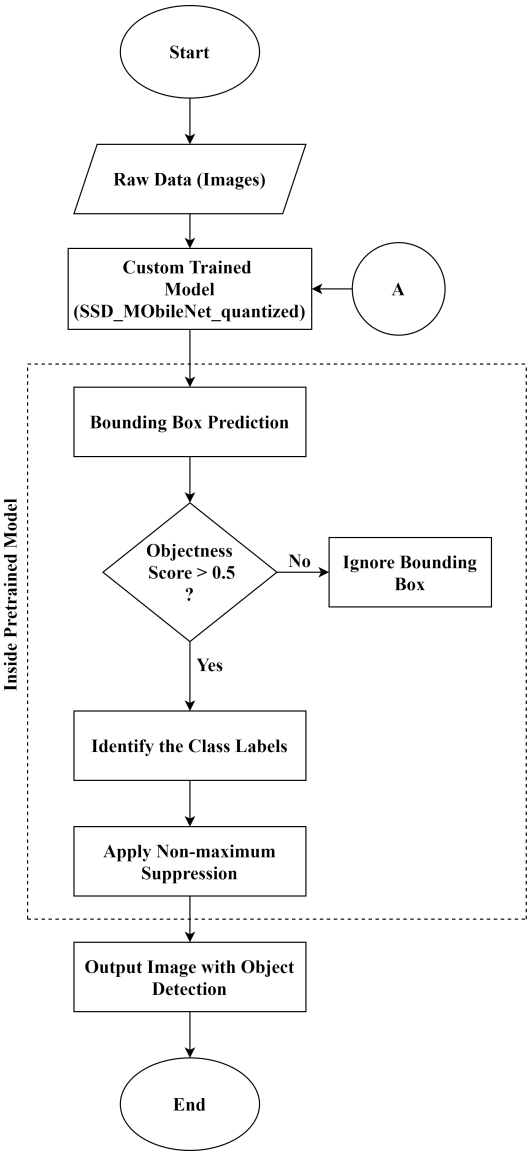

Fig. 6. Flowchart of Cash Recognition.

The trained model is exported to be applied to a raw image for classification. The model then examines,

forms bounding boxes, and predicts its labels along with a certainty level. Flowchart 6, shows cash recognition operation of the system.

The cash detection process is initiated with the help of an IR sensor indicating the presence of money. Then the Pi camera is turned ON and the live camera video is connected using the OpenCV library the live video is converted to images(frames) with conversion speed as per Raspberry Pi's processing power i.e. approximate 0.8 fps. These captured images are then fed into a customized trained model(SSD_MobileNet_V2) after being resized into 300x300 pixels. Then the trained model starts to predict and identify the class ID for any bounding box created in those images. If class ID is identified with a confidence level less than 50%(i.e. 0.5) then the bounding box is discarded else the identified class ID is stored in memory. The "class-label" corresponding to identified "class-ID" with a confidence level greater than 0.5(i.e. 50%) is then displayed in a graphical window.

As the bounding box starts to form as per the model's prediction, there may appear multiple bounding boxes in a single image. This can create confusion in further processes, so applying the Non-Maximum Suppression algorithm to identify and select a single bounding box using many overlapping bounding boxes [12]. The criteria for discarding any boxes are usually their overlap measure along with their probability (i.e. confidence score). And finally, a well-filtered and relatively accurate class is identified. Here the identified class is among Rs 5 , Rs 10, or RS 20. Then this identified class is used for further processing.

IV. Result and Discussion

In this vending machine, an updated list of items with the price is displayed on the LCD, and ordered is placed by using the keypad. By pressing a button on the keypad as that of the item number shown on the screen, the item was selected and the screen to supply quantity was displayed. Once quantity was selected, the user was able to press a predefined key for further process i.e. view, cancel, add or confirm. At the time of pressing a cancel button, the initial home screen was shown. While add button was clicked, the home screen was shown to input the order. When a view key was pressed after an item is ordered or a further item was added, all the ordered items with their total price were presented on the LCD. But pressing a view button following the cancellation showed a blank screen. After pressing confirm button, the total amount, as well as the payment option, was shown.

Two options were provided for payment i.e. cash and card. When key 2 was pressed, the "Enter Card" message was shown on display. After inserting an RFID card in the card slot, the RFID module detected the card and Arduino Due checked the card id as well as its balance. After validation was true, the bill amount was deducted and items was dropped by motorized action controlled by Arduino Due. Also, for the second transaction process cash payment was used. By pressing key 1 after confirming the order, the "Enter Cash" message was displayed. At first, Rs.5 was inserted. As Rs.5 was inserted IR sensor detected cash and Raspberry Pi operated the motor to pull cash inside, motorized action was performed till the IR sensor couldn't detect cash means the whole cash was pulled inside. Once the whole cash was inserted motor stopped and the PI camera opened for cash detection and validation. The fps of the Raspberry Pi was approximately 0.8 so it took some time to detect cash. A bounding box was formed around cash showing its confidence level and amount. The cash was only validated if it was detected 10 times. After validating cash, the amount was deducted from the total bill and cash was again pulled further inside by motor for 10 seconds. It asked to enter more cash so we inserted Rs.10 and Rs.20 one after another to pay the total bill. Once a sufficient amount was inserted Raspberry PI sent a signal to Arduino Due which dropped ordered items by running motor.

TABLE I
Confidence Level for Rs.5

	Cash Visible Portion		
Camera Height	***50%***	***75%***	***100%***
5 cm	0	0	0
8 cm	0%	90%	94%
10 cm	87%	92%	99%
12 cm	0%	0%	96%
15 cm	0	0	0

TABLE II
Confidence Level for Rs.10

	Cash Visible Portion		
Camera Height	***50%***	***75%***	***100%***
5 cm	0	0	0
8 cm	87%	90%	96%
10 cm	92%	96%	99%
12 cm	90%	92%	98%
15 cm	0	0	0

TABLE III
Confidence Level for Rs.20

	Cash Visible Portion		
Camera Height	***50%***	***75%***	***100%***
5 cm	0	0	0
8 cm	90%	92%	96%
10 cm	94%	96%	99%
12 cm	91%	94%	97%
15 cm	0	0	0

The confidence level of cash was varying due to camera height and lighting condition. The different confidence level for different amount is shown is

Fig. 7. Confidence Level of cash at Camera's height of 10cm.

listed in tables. Table I, Table II, and Table III show confidence levels of Rs.5, Rs.10, and Rs.20 correspondingly. From comparing values of these tables, the Pi camera's height was selected to be 10cm and mounted 10cm above the cash tray. Figure 7 shows confidence levels of all cash at 10cm height. The confidence level 0 means no bounding box was formed resulting no cash detected. Also, cash was not detected when there was poor lighting, in such a case flashlight was used.

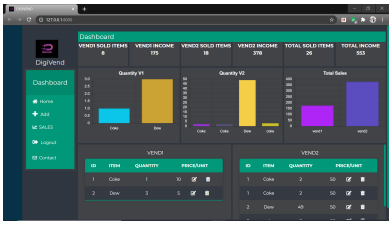

Fig. 8. Item Lists and Charts in Server.

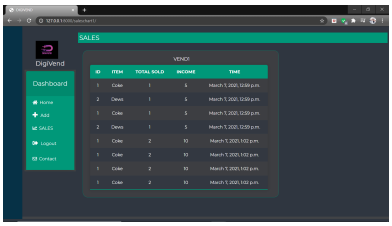

Fig. 9. Sales Data in Server.

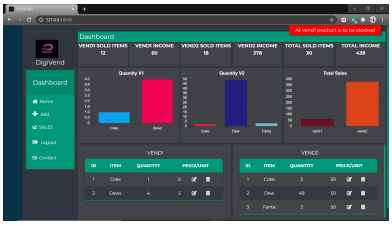

Fig. 10. Alert Notification.

After completion of the whole transaction, ordered data was sent to the server via NodeMCU. On the home screen of the server, tables of a list of items of a vending machine and various charts of present items as well as sold items are displayed as in Figure 8. And sold items with time-stamps were listed in the sales section of the server as in Figure 9. When an item's quantity was less than 2, an alert notification popped up in the server and remained until the quantity's value was updated. Figure 10 shows popped up alert notification in the server.

V. Conclusion and Future Enhancement

This paper put forth the key to prodigious exposure to the customer and make them have a completely different outlook towards vending machines back in the days. This Smart Vending Machine can transact both cash and cashless. Cash of Rs.5, Rs.10, and Rs.20 are only recognized by this vending machine so the customer has to enter only one of this cash. The confidence level was about 99%. Sometimes cash was not detected due to lack of light so we have to use a flashlight as a light source. When the number of items is less than 2, vending machine alerts the owner via server to restock it. Once quantity was updated alert message disappeared. A customer can buy more than one item at one transaction. Also, an owner can view the sales data of items. The fps of Raspberry Pi was low that is approximate 0.8 fps so cash detection takes some time. DC motor used for dropping items ran for 1400 milliseconds and stopped to drop one item. Since NodeMCU operates on a high baud rate, the serial communication was established at a 115200 baud rate with Arduino Due. Both laptop running server and NodeMCU must be connected on the same network for communication.

As this system is limited to only three types of cash, more types of cash can be used by creating a trained model of them. The system doesn't know itself when items are restocked, quantity must be updated manually to inform the system that it has been restocked. This can be solved using an IR sensor or image processing to detect items in a tray and when an item is restocked it can update data automatically. So as for further more advancement, the system can be made more user-friendly, completely IoT-based cloud server and recommendation systems can be incorporated.

Acknowledgment

The authors of this paper would like to express our sincere thanks to Kantipur Engineering College for providing us an environment and all the resources we needed to achieve this research content. Also, we would like to express our sincere thanks to our teachers, professors, and seniors for inspiration and generous help and idea in this research. We would like to thank Er. Manoj Gyawali, who help us throughout the project by providing proper guidance and advises as a mentor and adviser. Finally, we would like to express our heartfelt gratitude to all the people who have directly and indirectly helped to initiate this research and for their support up to the completion of the research.

REFERENCES

[1] Yokouchi, Toshio."Today and tomorrow of vending machine and its services in Japan." 2010 7th International Conference on Service Systems and Service Management. IEEE, 2010.

[2] Karthik, B. N., et al."Survey on IOT & Arduino based patient health monitoring system." International Journal of Scientific Research in Computer Science, Engineering and Information Technology 3.1 (2018): 1414-1417.

[3] E. P. Abad, "Design and implementation of paper vending machine for retail of common usable papers for unstop students," International Journal for Research in Applied Science & Engineering Technology, vol. 5, pp. 289–294, 2017.

[4] K. Ushikubo, "Automatic vending machine," Oct. 18 1988, uS Patent 4,778,983.

[5] A. Nicolini, "Payment system for automatic vending machines," Apr. 18 2002, uS Patent App. 09/977,924.

[6] J. Wan, M. A. Al-awlaqi, M. Li, M. O'Grady, X. Gu, J. Wang, and N. Cao, "Wearable iot enabled real-time health monitoring system," EURASIP Journal on Wireless Communications and Networking, vol. 2018, no.

[7] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg, "Ssd: Single shot multibox detector," in European conference on computer vision. Springer, 2016, pp. 21–37.

[8] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, and H. Adam, "Mobilenets: Efficient convolutional neural networks for mobile vision applications," arXiv preprint arXiv:1704.04861, 2017.

[9] S. Sanchez, H. Romero, and A. Morales, "A review: Comparison of performance metrics of pretrained models for object detection using the tensorflow framework," in IOP Conference Series: Materials Science and Engineering, vol. 844, no. 1. IOP Publishing, 2020, p. 012024.

[10] Ramzan, Aneeqa, Saad Rehman, and Aqib Perwaiz. "RFID technology: Beyond cash-based methods in vending machine." 2017 2nd International Conference on Control and Robotics Engineering (ICCRE). IEEE, 2017.

[11] Kumar, Santosh. "A Review On Client-Server Based Applications And Research Opportunity." International Journal of Recent Scientific Research 10 (2019).

[12] Sambasivarao, K."Non-maximum suppression (nms)." Online unter https://towardsdatascience. com/non-maximum-suppression-nms-93ce178e177c [Stand 16.10. 2019] (2020).Accessed 5 Aug, 2020.