**KIRANA PRODUCT BILLING**

* EDGE COMPUTING

**Team Name:**

DP\_Group

|  |  |  |
| --- | --- | --- |
| **CT Reference No** | **Name** | **Position** |
| CT20172257624 | Deeban N | Team Leader |
| CT20172255284 | Prashanth P | Team Member |

**Department:**

Computer Science and Engineering (B.E - CSE)

**College Name:**

M. Kumarasamy College of Engineering (Autonomous), Karur

(2016 - 2020 Batch)

**INTRODUCTION:**

Shopping is easy but waiting on a bill counter after shopping is too boring and tedious task. Huge amount of rush plus cashier preparing the bill with barcode scanner is too time consuming and results in long ques. So here we have made an innovative project called Kirana Product Billing.

The main objective of this project is, that retail store looking for more digitalized way of expanding their business. They want to use a more systematic way of checkout system and reduce their human work force at billing counters.

We need a system that allows automatic detection of product using camera. The detection of the product will be the size of the product, type of product and automatically take the cost of product to make a bill of materials at checkout by using Machine Learning concepts.

For an example, if toothpaste is placed in front of the camera, it should detect the product, according to the size and weight of toothpaste the system will understand the price of the product. The details of each product scanned are then presented at the bill of materials for payment.

**DETAILS OF TECHNOLOGY USED:**

* Machine Learning Algorithms to measure the size of the object
* **Optical Character Recognition (OCR)**
* **Image Pre Processing**
* **Object Segmentation**
* **Picture per metric algorithm**

**MACHINE LEARNING ALGORITHMS TO MEASURE THE SIZE OF THE OBJECT:**

The project provides a script to read an image based on the dimensions of the object.

**ALGORITHMS:**

1. **IMAGE PRE-PROCESSING:**

* Read an image and convert it no grayscale
* Blur the image using Gaussian Kernel to remove un-necessary edges
* Edge detection using Canny edge detector
* Perform morphological closing operation to remove noisy contours.

1. **OBJECT SEGMENTATION:**

* Find contours
* Remove small contours by calculating its area (threshold used here is 100)
* Sort contours from left to right to find the reference objects

1. **REFERENCE OBJECT:**

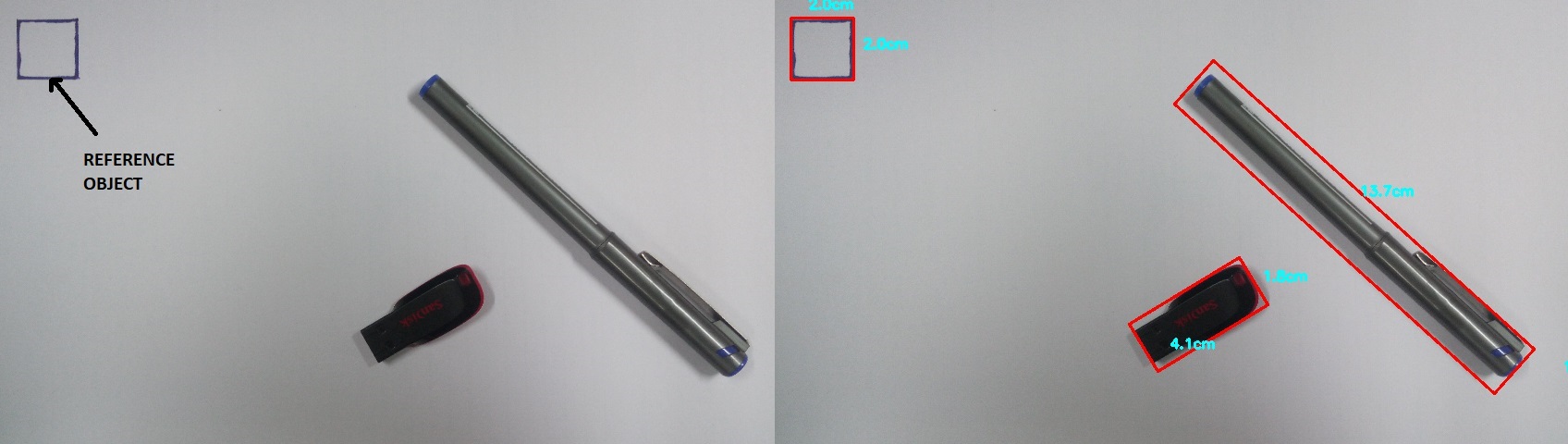
* Calculate how many pixels are there per metric (centimeter is used here)

1. **COMPUTE RESULTS:**

* Draw bounding boxes around each object and calculate its breadth and length.

1. **OPTICAL CHARACTER RECOGNITION (OCR):**

Optical Character Recognition (OCR) is the process of electronically extracting text from images or any documents like PDF and reusing it in a variety of ways such as full text searches. Pytesseract will recognize and read the text present in images. It can read all image types - png, jpeg, gif, tiff, bmp etc. It’s widely used to process everything from scanned documents.



**SOFTWARE AND HARDWARE REQUIRED:**

* + Camera
  + Weighting Machine
  + Python 3 Pip OpenCV Numpy
  + Objects in the Supermarkets

**CAMERA:**

It is used to capture the image of the product and give it to the OCR.

**WEIGHT MACHINE:**

Weight machine is used to calculate the weight of the product and gives the size of the product.

**ARCHITECTURE:**

**Product**

**Capture the image of the product and measure the weight.**

**Measure the length and breadth of the product in the image**

**Find the match of the product in the database.**

**Finally display the total cost of the products**

**SOLUTION DESCRIPTION:**

1. First take the photo of the product.
2. Get the dimensions of the product by using the picture per metric algorithm.
3. Mark the weight of the product and get the quantity of the product.
4. By using the OCR, extract the features from the image.
5. Then Match the dimensions of the product from the database.
6. If the product is matches more than one item, then use the OCR and the weight of the product, so that we can match the right product.
7. If the same product is added again and again we can add the count and make the bill to count the total at the last.
8. After that the final bill will be displayed.
9. Take the receipt and pay the amount to cashier.

**CONCLUSION:**

This project will be very useful to calculate the bill of the product automatically. This System is used to reduce the time of the customer and also the work of cashier in the supermarkets. This project is very useful for the markets in the crowd areas.