DESIGN AND FABRICATION OF SORTING MACHINE

MEB334 – DESIGN PROJECT

Submitted by

P.V.N.S. SATHWIK	**(0015127001)**
D. MAHESH	**(0015127064)**
K. SANDEEP	**(0015127065)**
VISHNU VARDHAN DADI	**(0015127066)**

Under the guidance of

DR. M.M. RAMYA

In partial fulfilment for the award of the degree

Of

BACHELOR OF TECHNOLOGY

IN

MECHANICAL ENGINEERING

HINDUSTAN
INSTITUTE OF TECHNOLOGY & SCIENCE
(DEEMED TO BE UNIVERSITY)

DEPARTMENT OF MECHANICAL ENGINEERING

SCHOOL OF MECHANICAL SCIENCES

HINDUSTAN INSTITUTE OF TECHNOLOGY AND SCIENCE

PADUR 603103

MAY 2018

DESIGN AND FABRICATION OF SORTING MACHINE

MEB334 – DESIGN PROJECT

Submitted by

P.V.N.S. SATHWIK	**(0015127001)**
D. MAHESH	**(0015127064)**
K. SANDEEP	**(0015127065)**
VISHNU VARDHAN DADI	**(0015127066)**

Under the guidance of

DR. M.M. RAMYA

In partial fulfilment for the award of the degree

Of

BACHELOR OF TECHNOLOGY

IN

MECHANICAL ENGINEERING

HINDUSTAN
INSTITUTE OF TECHNOLOGY & SCIENCE
(DEEMED TO BE UNIVERSITY)

DEPARTMENT OF MECHANICAL ENGINEERING

SCHOOL OF MECHANICAL SCIENCES

HINDUSTAN INSTITUTE OF TECHNOLOGY AND SCIENCE

PADUR 603103

MAY 2018

HINDUSTAN

INSTITUTE OF TECHNOLOGY & SCIENCE

(DEEMED TO BE UNIVERSITY)

BONAFIDE CERTIFICATE

Certified that this project report **"DESIGN AND FABRICATION OF SORTING MACHINE"** is the bonafide work of **P.V.N.S. SATHWIK (0015127001), D. MAHESH (0015127064), K. SANDEEP (0015127065) and VISHNU VARDHAN DADI (0015127066),** who carried out the project work under my supervision during the academic year 2017-2018**.**

HEAD OF THE DEPARTMENT	**SUPERVISOR**
Dr. G. RAVIKUMAR SOLOMON	**Dr. M.M RAMYA**
PROFESSOR	PROFESSOR
Department of Mechanical Engineering	Center for Automation and Robotics
Hindustan Institute of Technology	Hindustan Institute of Technology
and Science, Padur.	and Science, Padur**.**

INTERNAL EXAMINER	**EXTERNAL EXAMINER**
Name: ____________________	Name: __________________
Designation: _______________	Designation: ______________

Project Viva- Voce conducted on _____________

ACKNOWLEDGEMENT

We would like to place on record our sincere thanks and gratitude to **Dr. ELIZABETH VERGHESE**, Chancellor of Hindustan University for her endeavor in educating us in her esteemed institute which has helped us work towards our goal.

We also express our sincere thanks to **Dr. K.P. ISAAC,** Vice Chancellor of Hindustan University and to **Mr. ASHOK VERGHESE**, Director of Hindustan University.

We would like to thank **Dr. G. RAVIKUMAR SOLOMON**, Head of the Department of Mechanical Engineering, Hindustan Institute of Technology and Science for their guidance and for the opportunity provided to do this project.

I wholeheartedly thank **Dr. M.M. Ramya,** Professor, CENTRE FOR AUTOMATION AND ROBOTICS, Hindustan Institute of Technology and Science for her constant support, encouragement and guidance during every stage of the experimental work that brought this project to a successful completion.

We are also pleased to express our kind thanks to all faculties and the lab assistants, lab in-charge faculty for their constant help and advice for the completion of our project.

ABSTRACT

This project mainly focuses on the design and fabrication of Sorting Machine. As the development and customer expectations in the high-quality foods are increasing day by day, it becomes very essential for the food industries to maintain the quality of the product. Therefore, it is necessary to have the quality inspection system for the product before packaging. Automation in the industry gives better inspection speed as compared to the human vision. The automation based on the computer vision is cost effective, flexible and provides one of the best alternatives for more accurate, fast inspection system. Image processing and image analysis are the vital part of the computer vision system. In this project, we discuss the real time quality inspection of the biscuits of premium class using computer vision. It contains the designing of the system, implementing, verifying it and installation of the complete system at the biscuit industry. Overall system contains Image acquisition, Preprocessing, Important feature extraction using segmentation, Color variations, Interpretation and system hardware.

TABLE OF CONTENTS

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

1.1 PROBLEM DEFINITION

With the advances in Machine vision technology, signal processing techniques are widely applied to many food quality applications. In this project we developed two methods to solve problems in the biscuit quality while packing using Image processing techniques. First problem is the detection of broken biscuits that are moving on a conveyor for packing. This problem is quite common in many food processing industries and various methods have been in practice as a solution for this problem. Second problem is the detection of burnt biscuits that are overly baked inside the oven and are moving on the conveyor for packing. With the help of image processing methods, we can find these burnt biscuits and separate them. Further we can do data logging for this method which will help us to implement in predicting the actual fault among the number of machines used in the biscuit production.

In our project we are using a mini prototype of the industrial conveyor along with the camera and the biscuit sorting mechanism. The camera captures the biscuit image which is processed using the image processing techniques and the Boolean logic generated will trigger the sorting mechanism. Here we are using a Rack and Pinion mechanism for the sorting of biscuits.

1.2 MACHINE VISION

Recent developments in machine vision and supporting technologies have resulted in general acceptance of the feasibility and profitability of implementing visual inspecting systems in quality assurance operations of food producing lines. Machine vision benefited the most from the increase in processing and storage powers of modern chips, and from the emergence of megapixel sensing and imaging devices.

Machine vision technology utilizes image processing techniques for the purpose of extracting visual features about an object for a variety of qualitative, quantitative and control applications. During the last few years, the trend in designing and building

machine-based inspection systems have almost always followed one of two patterns. These are the traditional sensor (camera) – computer – interface pattern, and a newer pattern– the so called smart camera pattern.

1.3 OPENCV FOR IMAGE PROCESSING

OPENCV (Open Source Computer Vision Library) Windows, Mac OS, Linux, iOS and Android. It has C++, Python and Java interfaces. OPENCV is designed for computational efficiency and with a strong focus on real time applications. Its usage ranges from interactive art, to mines inspection, image processing, stitching maps on the web or through advanced robotics.

In this project we are using OPENCV version 3.2.0 with Python 2.7. The operating system we are using is UBUNTU 16.0.4 which is Linux distro.

ADVANTAGES OF OPENCV:

- OPENCV is a free open source library.
- It is compatible with C/C++/PYTHON.
- OPENCV is very fast.
- Low RAM usage (approx. 60-70 mb).
- It is portable as OPENCV can run on any device that can run C.

1.4 ARDUINO FOR COMMUNICATION

Arduino is an open source computer hardware and software company, project and user community that designs and manufactures single-board microcontrollers and microcontroller kits for building digital devices and projects.

The boards feature serial communications interfaces, including Universal Serial Bus (USB) on some models, which are also used for loading programs from personal computers.

In this project we are using Arduino Mega 2560 Micro controller for the Serial Communication and also for the control of motor which drives the Pinion wheel in Rack and pinion Mechanism.

Arduino Mega 2560:

Fig 1.1 Arduino Mega 2560

1.5 COMPONENTS OF SYSTEM

Conveyor Belt:

A conveyor belt is the carrying medium of a belt conveyor system (often shortened to belt conveyor). A belt conveyor system consists of two or more pulleys (sometimes referred to as drums), with an endless loop of carrying medium—the conveyor belt—that rotates about them. One or both of the pulleys are powered, moving the belt and the material on the belt forward. The powered pulley is called the drive pulley while the unpowered pulley is called the idler pulley.

Fig 1.2 Conveyor Belt

Pillow Block Bearings:

A pillow block usually refers to a housing with an included anti-friction bearing. A pillow block refers to any mounted bearing wherein the mounted shaft is in a parallel plane to the mounting surface, and perpendicular to the centre line of the mounting holes, as contrasted with various types of flange blocks or flange units. A pillow block may contain a bearing with one of several types of rolling elements, including ball, cylindrical roller, spherical roller, tapered roller, or metallic or synthetic bushing. The type of rolling element defines the type of pillow block

Fig 1.3 Pillow Block Bearing

Shaft:

A shaft is a rotating machine element, usually circular in cross section, which is used to transmit power from one part to another, or from a machine which produces power to a machine which absorbs power. The various members such as pulleys and gears are mounted on it.

Fig 1.4 Shaft

DC Gear Motor:

A DC motor is any of a class of rotary electrical machines that converts direct current (DC) electrical energy into mechanical energy. The most common types rely on the forces produced by magnetic fields. Nearly all types of DC motors have same internal mechanism, either electromechanical or electronic, to periodically change the direction of current flow in part of the motor.

Fig 1.5 DC Gear Motor

Rack and Pinion Gears:

A rack and pinion is a type of linear actuator that comprises a pair of gears which convert rotational motion into linear motion. A circular gear called "the pinion" engages teeth on a linear "gear" bar called "the rack"; rotational motion applied to the pinion causes the rack to move relative to the pinion, thereby translating the rotational motion of the pinion into linear motion.

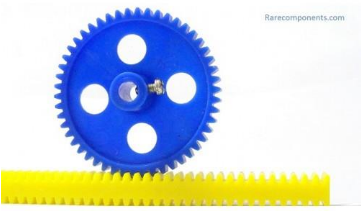


Fig 1.6 Rack and Pinion

Webcam:

In this project we are using a webcam, in which Image Resolution is Interpolated to 25 Mega Pixels with 6 Light sensors. Image Control Colour saturation, brightness, sharpness and brightness is adjustable; Snap shot switch is available for taking still pictures.

Fig 1.7 Webcam

1.6 TYPES OF DEFECTS IN BISCUITS

There are many types of defects in the biscuits that are due to the malfunctioning of different components of the biscuit manufacturing unit. The common defects in the biscuit manufacturing are:

SPREADING OF BISCUITS:

The distortion in the shape of the biscuits, that is increase in breadth or height of biscuit than the standard.

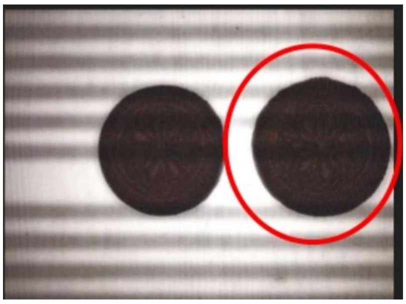


Fig 1.8 shape distortion

CREAM OOZING AND REVERSE SHELL:

Cream Oozing-Cream comes out of the Sandwiched biscuits. Reverse Shell-Cream sandwich biscuits when one of the biscuits is sandwiched in the reverse manner.

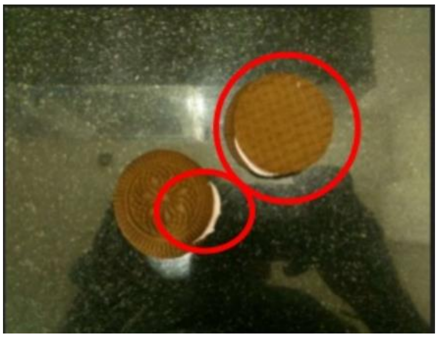


Fig 1.9 Cream oozing and Reverse shell

BREAKAGE:

Breaking of the biscuit to physical abrasion or pressure.

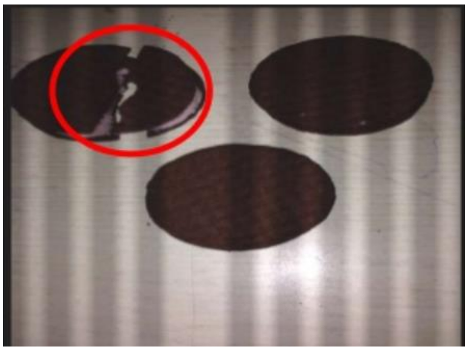


Fig 1.10 Broken biscuits

BLISTERS:

Dark patches at the centre of the biscuit due to overcooking in the oven.

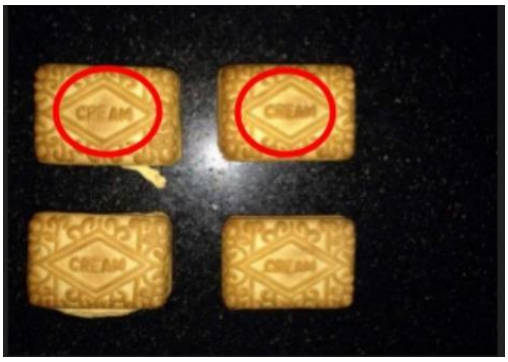


Fig 1.11 Blisters

1.7 IMPORTANCE OF QUALITY INSPECTION

The incredible importance of food safety and quality inspections is largely based on not only the well-being and safety of the public consumer, but also the relationship between food quality and brand recognition. While contemporary inspection technology very effectively attends to the health and safety concerns of the public, proper and successful inspections are also key components in building and improving upon a company’s brand image. Due to the incredibly accurate inspection applications of modern Machine – vision technology, companies are able to more consistently deliver on quality guarantees, as these technologies can catch what the human eye and conventional inspection mediums cannot; this eliminates the threat of public relations disasters following flawed or potentially unsafe product distribution**.**

These technologies also greatly limit the risk of major recalls and widespread consumer dissatisfaction, which can be incredibly damaging for a brand. Almost all of today's machine vision inspection technologies are HACCP (Hazard Analysis and Critical Control Point) compliant, and adhere to Global Food Safety Initiative (GFSI) quality standards.

In addition to incredibly accurate safety inspection, quality assurance and the protection of company's brand, today's highly effective inspection technologies can also your company countless manufacturing dollars and labour time. Companies utilizing these methods can spend less resources physically inspecting their products and working to verify their safety, and more on other measures to improve business.

CHAPTER 2

LITERATURE SURVEY

2.1 IMAGE PROCESSING

Image processing is one of the most increasing areas now a day. By converting analog image into the digital system and using digital image processing technique it is possible to extract various features from the image [2]. In our project we are resizing the image to reduce computation time in further image processing techniques [1]. Effective resizing of images should not only use geometric constraints but consider the image content as well [3].

Denoising of the image is to be done as the camera is working for a long time as a result thermal noise is added [5]. The capture of the image takes very less time which gives dark noise in the image. This can be overcome by using Gaussian method of blurring [4].

The results on an RGB image may give some pleasing results but to attain faster computation colour image is to be converted to Gray scale image. Even this is a single channel the image sharpness, contrast is not lost. The major part of the information for the inspection will be still there in the case of edge detections and contours finding [6].

For defect detection Otsu's thresholding method gives better results [7]. The Otsu method, provides satisfactory results for thresholding an image with a histogram of bimodal distribution. This method, however, fails if the histogram is unimodal or close to unimodal. For defect detection applications, defects can range from no defect to small or large defects, which means that the gray-level distributions range from unimodal to bimodal [7].

In real time usage of machine vision, the vision sensors and other vision-based equipment are controlled by a single computer system for better communication between the sensors. This is achieved by serial communication, which is mostly prescribed in many inspection cases [8].

2.2 MACHINE COMPONENTS

For the transport of biscuits and other food products in a manufacturing unit a variable speed conveyor is preferable to use as the inspection needs a slow conveyor speed and for transport a faster conveyor speed [9]. The rollers used for the movement of the conveyor will provide a smoother operation during speed variation and will with the vibrations which is best suited for food transportation [10].

In food inspection the sorting of the brittle food materials is done using a linear push rod which can be easily attained by a rack and pinion mechanism used in car steering. This will give required force on the target and separate it to the requirement [11].

The conveyor works on microcontroller logic during the food inspection. This conveyor can be controlled properly with a variable speed by microcontroller only when a DC geared motor is implemented [12].

In any rigid conveyor system, a shaft has to be used to transmit the motion between two rollers at one end. The belt takes the support of the shaft to rotate[13].

CHAPTER 3

PROBLEM DESIGN

3.1 IMAGE PROGESSING

3.1.1 Image Acquisition

In image processing, it is defined as the action of retrieving an image from some source, usually a hardware-based source for image processing. The image acquired is completely unprocessed. In our project we as using a portable USB webcam for image acquisition. In the code we have defined the port for webcam and defined a value for FPS (Frames per second).

Code:

```
#*** Setting up camera port and FPS***
Cam_port=1
fps = 30
camera=cv2.VideoCapture(Cam_port)
def get_img():
    retval,img=camera.read()
    return img
for i in xrange(fps):
 temp=get_img()
print("=======>>>Taking image...=======<<<")
cam_cap=get_img()
#giving file name for the captured image
file="bisk_img.png"
cv2.imwrite(file, cam_cap)
#destroying the camera loop
del(camera)
cv2.waitKey(100)
#reading the saved image
img=cv2.imread('bisk_img.png')
```

3.1.2 Image Resizing

The image read will be very large in size which takes more computation time in further image processing steps. So, we are resizing the image to reduce the time of computation. Here the resize is done by taking care that most of the information is not lost. This is attained by trial and error method.

We will be using the "**cv2.resize**" function to resize our images. In this line of code, we define our new image width to be 150 pixels. In order to compute the ratio of the new height to the old height, we simply define our ratio **r** to be the new width (150 pixels) divided by the old width, which we access using **image.shape[1]**. The width of the new image will be 150 pixels. The height is then computed by multiplying the old height by our ratio and converting it to an integer.

The first argument is the image we wish to resize and the second is our computed dimensions for the new image. Last parameter is our interpolation method, which is the algorithm working behind the scenes to handle how the actual image is resized. In general, we find that using cv2. INTER_AREA obtains the best results when resizing; however, other appropriate choices include cv2.INTER_LINEAR, cv2.INTER_CUBIC, and cv2.INTER_NEAREST.

Code:

```
#resizing the image
r=150.0/img.shape[1]
dim=(150, int(img.shape[0] * r))
rsimg=cv2.resize(img,dim, interpolation = cv2.INTER_AREA)
```

3.1.3 Image denoising

The image obtained may have some noise in it due the factors like light, temperature of sensor(Webcam). So, we are reducing this noise for better results by smoothening the image. There are many smoothing operations but we are using gaussian blur because of the shadows and light factors involved.

The first argument to the function the image we want to blur. Then, similar to cv2.Gaussianblur, we provide a tuple representing our kernel size. Again, we start with a small kernel size of 21×21 and start to increase it.

The last parameter is our σ, the standard deviation in the x-axis direction. By setting this value to 0, we are instructing OpenCV to automatically compute them based on our kernel size.

Code:

```
#smoothening the image to remove noise
blur=cv2.GaussianBlur(rsimg,(21,21),0)
```

3.1.4 Colour Conversion

The image obtained is a BGR image which contains three channels Blue, Green, Red. This will increase the computation time of image to be processed. So, we are converting this BGR image to GRAY SCALE image which will have only one channel. This will reduce our computation time and gives better results. To achieve this, we use **cv2.cvtColor** function of OPENCV.

The first parameter is the BGR image and the second parameter is the function to convert BGR to GRAY SCALE.

Code:

```
#converting the image to gray scale from BGR
gray=cv2.cvtColor(blur,cv2.COLOR_BGR2GRAY)
```

3.1.5 Thresholding

Thresholding is the simplest method of image segmentation. From a gray scale image thresholding can be used to create binary images. Thresholding replaces each pixel in an image with a black pixel if the image intensity $I_{i,j}$ less than some fixed constant T (that is, $I_{i,j} < T$), or a white pixel if the image intensity is greater than the constant.

There are many thresholding methods but we are using Otsu's Thresholding. We are using **cv2.threshold** function in OPENCV to convert GRAY SCALE image to BINARY IMAGE.

The first parameter is the gray scale image, second is the fixed constant for thresholding, third is the value to be attained, fourth is the binarization methods in Otsu's method.

Code:

```
#Thresholding the image using otsu thresholding
ret, otsu =
cv2.threshold(gray,0,255,cv2.THRESH_BINARY_INV+cv2.THRESH_OT
SU)
```

3.1.6 Area Calculation

In this step we are calculating the number of pixels using the OPENCV function **np.sum** and we define pixel values to be counted. Here 255 represents white pixels. We are printing this pixel value on the python console.

Code:

```
#counting number of white pixels in the image
n_white_pix = np.sum(otsu == 255)
#printing the number of pixels
print('Number of white pixels:', n_white_pix)
```

3.2 SERIAL COMMUNICATION

Serial communication is the process of sending data one bit at a time, sequentially, over a communication channel or computer bus. This is in contrast to parallel communication, where several bits are sent as a whole, on a link with several parallel channels.

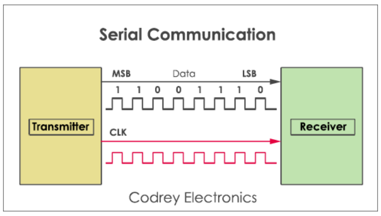


Fig 3.1 Serial Communication

We have selected serial communication for our project because python 2.7 have a library called **Serial** which contains the functions for the serial communication. The type of serial communication we are using is UART (Universal Asynchronous Receiver Transmitter). This type of communication won't use a clock but uses a Baud rate for communication.

3.2.1 Baud Rate

It is the number of bits transmitted by the transmitter per second. It is measured in bps (bits per second). The transmitter and receiver should always be set to the same baud rate. Standard baud rate settings are – 2400, 4800, 9600, 14400, 19200, 38400, 57600, 115200, etc...

In our project we specify the baud rate in Arduino program as **Serial.begin(19200).** The function Serial.begin, starts the serial communication.

In the project we have set our Baud rate as 19200. We selected so because the webcam we are using is a portable USB type, which also uses serial communication for the data transfer. This serial communication generally uses 9600 baud rate. To make our baud rate different from this we have defined a baud rate of 19200.

3.2.1 Serial Port

In computing, a Serial port is a serial communication interface through which information transfers in or out one bit a time contrast to parallel port.

For serial communication to happen we have to define the port through which we are sending the data in the code. We are using **serial.Serial** function of python 2.7 to define the port and the baud rate in the code.

The first parameter is the Port "**/dev/ttyACM0"** which is the serial port for the UBUNTU 16.04 LTS operating system. If it is a WINDOWS operating system, this will be like “**COM1”**. Here in both the cases 0 and 1 means the port number. This number may vary for different systems.

We are using **set.write** function to write the information to be send to the Receiver. In our case we are sending ‘Y’. If this ‘Y’ matches with the Receiver script the loop executes.

Code:

```
ser = serial.Serial(“/dev/ttyACM0”, 19200)
ser.write(‘Y’)
```

3.2.3 Arduino Motor Shield

Arduino works at an input voltage of 5v, but our Motors require 12v for their effective functioning. The battery we are using in our project is 12v. So, to accommodate the same battery for both Arduino and the Motors we are using a Motor driver shield for the Arduino. This Arduino Motor shield contains three L293D ICs which shares the required voltages for motor and Arduino. This motor shield support 4 Dc motors.

Code:

```
#include <AFMotor.h>
char incomingBit;
AF_DCMotor motor1(1, MOTOR12_64KHZ);
void setup () {
  Serial.begin(19200);
  motor1.setSpeed(225);
}
void loop(){
  if (Serial.available() > 0){
    incomingBit = Serial.read();
     if(incomingBit == 'Y' || incomingBit == 'y') {
      motor1.run(FORWARD);
      delay(2000);
      motor1.run(BACKWARD);
      delay (2000);
```

```
    }
    else {
     Serial.println("no defects");
     delay (15);
    }
  }
}
```

3.3 DESIGN OF CAD MODEL:

SOLIDWORKS 2015 was used as a software design tool to design and simulate our project.

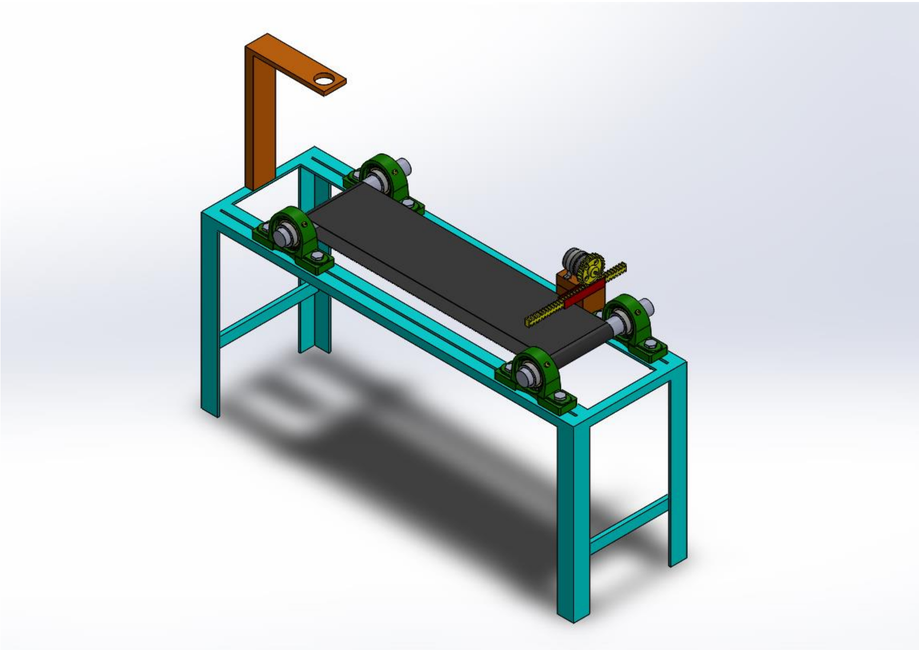


Fig 3.2 Sorting Machine

Frame:

Length = 72 cm

Width = 28 cm

Height = 40 cm

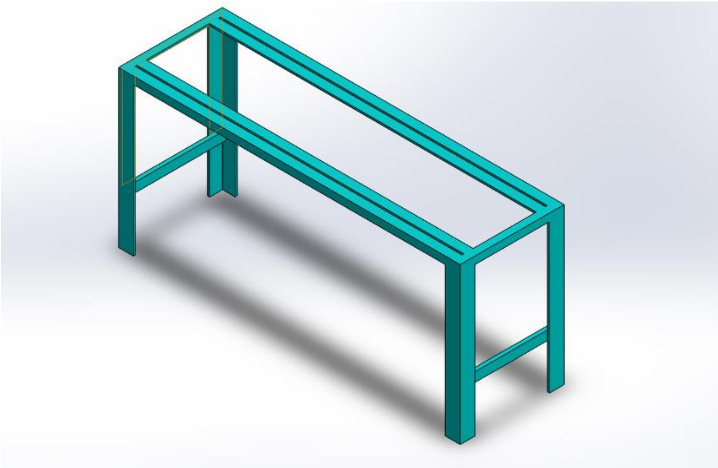


Fig 3.3 Frame

Ball Bearing:

Internal Diameter = 20 cm

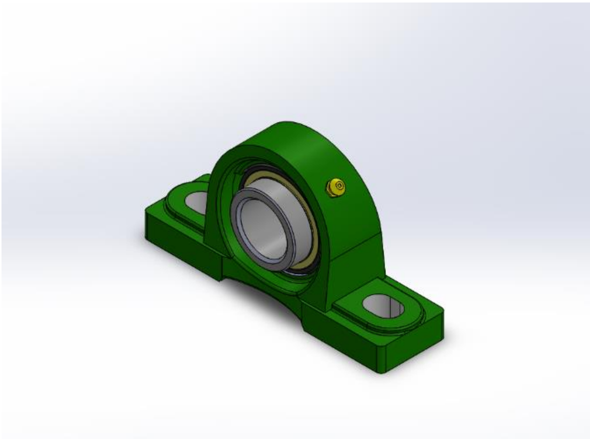


Fig 3.4 Bearing

Shaft:

Diameter of the shaft(ds) = 2cm

Length of the shaft = 30cm

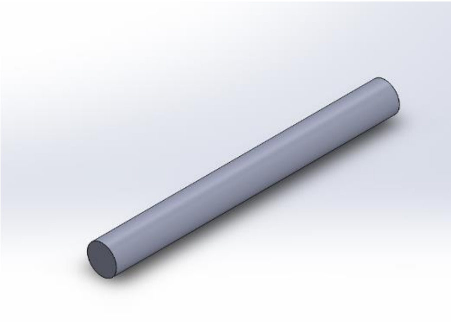


Fig 3.5 Shaft

Belt specifications:

Length of the belt (l)	=	118cm
Centre distance (c)	=	58 cm
Torque required (T)	=	4 N-m

Motor specifications:

Speed	=	10 rpm
Torque	=	4 N-m
Power	=	4 watts

Pinion specifications:

No. of teeth (Z1) = 36 teeth

Diameter of pinion (d1) = 6cm

Module (m) = $\left(\frac{d_1}{z_1}\right)$ = 1.669

Speed of pinion (n) = 100 rpm

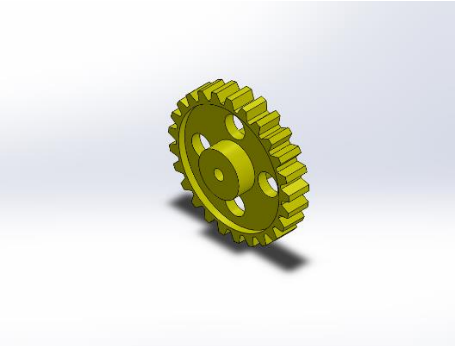


Fig 3.6 Pinion

Rack specifications:

No. of teeth (Z2) = 50

Length of the Rack (L) = 23 cm

Width of the rack (W) = 1.2 cm

Speed of Rack (N) = 72.46 rpm

Speed ratio (i)= $\left(\frac{n}{N}\right)$ = 1.38

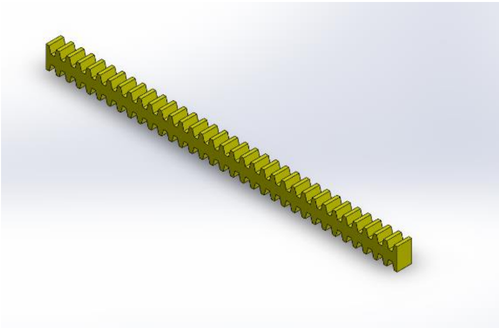


Fig 3.7 Rack

CHAPTER 4

RESULT AND DISCUSSION

Result:

Performance metrics:

Number of images taken = 20

Number of broken biscuits in images = 5

Number of biscuits in good shape =15

Before doing the fabrication work we designed the model in “SolidWorks” software and we simulated that.

In this project we have developed a conveyor system in which quality inspection of the biscuits is being carried out using certain image processing techniques with the help of OpenCV and python.

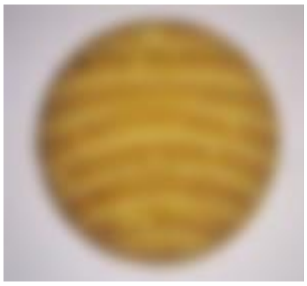
Arduino Mega 2560 is used as a microcontroller to actuate the mechanisms. A Serial communication is done between Python and Arduino, to specify the Boolean obtained in image processing.

All the image processing techniques are done by capturing the images through a webcam. Then, converting the raw image to a grayscale image and performing threshold operations to convert the image to binary. Therefore, we calculate the number of pixels in the binary image to create a Boolean for the serial communication.

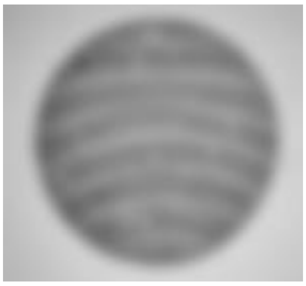
Arduino code is written to control the motor, i.e. to sort the defective biscuits from the good biscuits.

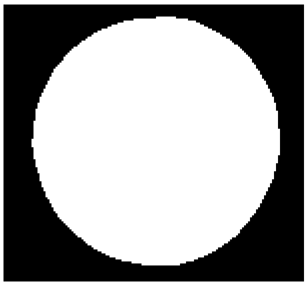
After completing the fabrication, we tested the project module by supplying the power to the motor. After that we calculated the delay of the motor and included that in the program. The Processing through python is little slow but the output was successful.

Normal Biscuit:



a) Original img b) Blurred img



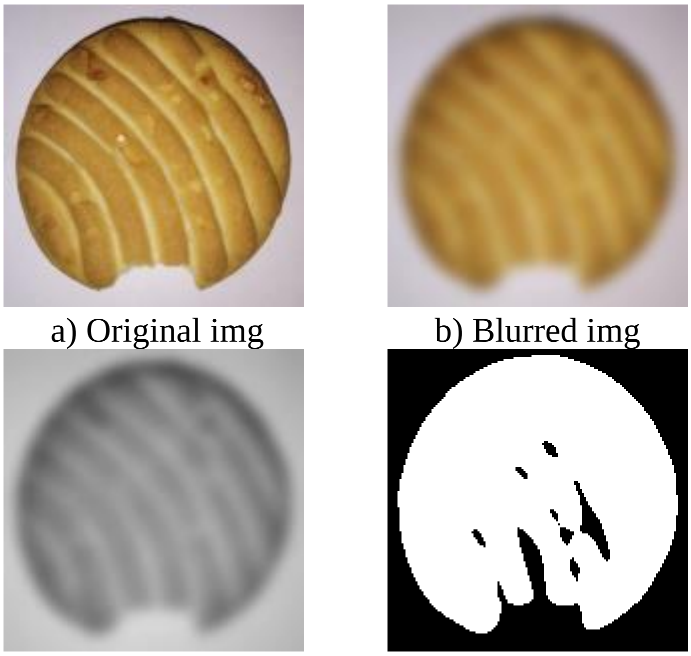


c) Grayscale img d) Threshold img

Fig 4.1 Normal Biscuit

The above pictures are results for the perfect shaped biscuit during every step of the image processing algorithm. These results are obtained under proper lighting conditions. Results may vary during high light exposure and during very low light conditions.

Broken Biscuit:



a) Original img b) Blurred img

c) Grayscale img d) Threshold img

Fig 4.2 Broken Biscuit

The above pictures are results for the broken biscuits during every step of the image processing algorithm. These results are obtained under proper lighting conditions. Results may vary during high light exposure and during very low light conditions.

Image Taken	Image Processed	Result
Good biscuits (15)	Good biscuits (14)	93.33%
Good biscuits (15)	Bad biscuits (1)	-6%
Bad biscuits (5)	Good biscuits (0)	0%
Bad biscuits (5)	Bad biscuits (5)	100%

Overall efficiency of machine = 93%

Due to very small broken region in the biscuit some of the biscuits were actually not detected. This can be further increased by multiple inspection in the code.

CHAPTER 5

CONCLUSION

5.1 Conclusion

Although there is much advancement in the field of Automation in industries, Image processing techniques help a lot in the quality inspection of the products being manufactured in the industries. The Project was an exposure to the world of practical working knowledge. It gives us an opportunity to apply the theories studied in the Machine vision in a practical situation. It also gave an opportunity to know the practical difficulties that arise in the process of designing and fabrication. Right now the accuracy of the machine is 93%. The accuracy of the machine can be further increased by replacing the camera with higher precision and also proper calculation of conveyor speed and ejector speed.

5.2 FUTURE SCOPE OF THE PROJECT

The future scope of the project is to improve the quality of inspection, by improving the image processing techniques based on the colour segmentation. So, that the burnt biscuits or extra fried biscuits can also be sorted. The log of these colour segmentation of the biscuits can also be stored in a CSV file. This logged data can be used as a input data set for the Linear Regression model in Machine Learning. With help of this technique we can actually predict which machine is malfunctioning in the entire biscuit manufacturing unit.

5.3 REFERENCE

[1] Cheng-JinDuDa-WenSun1. computer vision for food quality inspection, *Science Direct* ,11 February 2004.

[2] TadhgBrosnanDa-WenSun. Image processing in food inspection, *Science Direct*, 22 July 2003.

[3] Shai Avidan. Effective image resizing for faster computation, *Ariel Shamir Proceeding SIGGRAPH '07 ACM SIGGRAPH 2007 papers*, Article No. 10.

[4] Richard Szeliski, Yuan, Jian Sun, Long Quan, Heung-Yeung Shum Lu. Image deblurring with blurred/noisy image pairs, *Science Direct*, Article No.: 1 doi>10.1145/1275808.1276379

[5] J. Portilla; V. Strela; M.J. Wainwright; E.P. Simoncelli. IEEE Transactions on Image Processing, *IEEE papers* ,(Volume: 12, Issue: 11, Nov. 2003)

[6] C. Saravanan, A detailed report on colour conversion, *2010 Second International Conference* on 19-21 March 2010. DOI: 10.1109/ICCEA.2010.192.

[7] Hui-FuangNg, Thresholding in information extraction, *Science Direct*, August 2008.

[8] Jung-Hun Kim; Suk Ahn; Jae Wook Jeon; Jong-Eun Byun, Serial communication of industry equipment using a computer,*ISIE 2001. IEEE International Symposium* on12-16 June 2001.

[9] E.W.T.Ngai F.F.C.Suk S.Y.Y.Lo, Report on conveyor application in different places, *IEEE papers*,16 June 2007.

[10] T.OhjiaS.IchiyamaaK.AmeiaM.SakuiaS.Yamadab, Design of roller for different conveyors , *US-Patent* 14 January 2004.

[11] John Baxter, "Analysis of Stiffness and feel for a Power-Assisted Rack and Pinion Steering Gear," *SAE Technical Paper 880706,1988*.

[12] K. T. Chau; Dong Zhang; J. Z. Jiang; Chunhua Liu; Yuejin Zhang, DC geared motors for speed variation, *IEEE Transactions on Magnetics* (Volume: 43, Issue: 6, June 2007).

[13] A.M.Crawford∗J.H.Curran, Report on types of shafts and application, *Science Direct* ,21 February 2003.

[14] https://docs.opencv.org/3.0-beta/doc/py_tutorials/py_tutorials.html

[15] https://pythonhosted.org/pyserial/

[16] http://www.numpy.org/

[17] https://www.arduino.cc/

[18] OpenCV Documentation for Python/Image processing http://opencv-python-tutroals.readthedocs.io/en/latest/py_tutorials/py_imgproc/py_table_of_contents_imgproc/py_table_of_contents_imgproc.html

[19] Python extension packages https://www.lfd.uci.edu/~gohlke/pythonlibs/