

A
REPORT OF PROJECT
ON
**“Inverse Robot Kinematic Solution with Automated
Object Detection System”**

Submitted in partial fulfilment of the requirement for the degree of Production Engineering
course of semester-1 of Savitribai Phule Pune University

Submitted By

Jadhav Saurabhi Rajendra (B150131328)
Joshi Sumit Vinayak (B150131331)
Kulkarni Omkar Prashant (B150131340)
Marsha Anuj Vijay (B150131344)

Under the guidance of

Prof. (Dr.) Keshav N. Nandurkar
Prof. (Dr.) P. J. Pawar



Department of Production Engineering
K.K. WAGH INSTITUTE OF ENGINEERING EDUCATION & RESEARCH
NASHIK- 422003
(2020-2021)

CERTIFICATE



DEPARTMENT OF PRODUCTION ENGINEERING

K. K. WAGH INSTITUTE OF ENGINEERING EDUCATION AND RESEARCH,

NASHIK – 422003

The project report entitled

“Inverse Robot Kinematic Solution with Automated Object Detection System”

Submitted By

Jadhav Saurabhi Rajendra (B150131328)

Joshi Sumit Vinayak (B150131331)

Kulkarni Omkar Prashant (B150131340)

Marsha Anuj Vijay (B150131344)

In partial fulfilment of the requirement for the award of Degree of Bachelor of Production Engineering course semester-1 of Savitribai Phule Pune University, during the academic year 2020-2021

Dr. K.N. Nandurkar Dr. P.J. Pawar
(Project Guide)

Dr. P.J. Pawar
(H.O.D)

Dr. K.N. Nandurkar
(Principal)

1. Examiner
2. Examiner

ACKNOWLEDGEMENT

We would like to thank our department for supporting us during this semester and encouraging us to be innovative and backing us in everything we did.

We would like to thank with profound gratitude our principal of the institute **Dr. K. N. Nandurkar** and our beloved Head of Department **Dr. P. J. Pawar** for having provided facilities to carry out this project.

We deem it's our duty to place our sincere admiration and heart-felt gratitude to our guide **Dr. K. N. Nandurkar & Dr. P. J. Pawar**, for having effectively guided and supervised throughout this project by imparting his erudite knowledge and personalized guidance blended with his exemplary patience and encouragement.

We shall be failing in our duty if we do not thank all teaching, non-teaching staff of our Department. And friends for having morally supported us while doing this project.

Jadhav Saurabhi Rajendra (B150131328)

Joshi Sumit Vinayak (B150131331)

Kulkarni Omkar Prashant (B150131340)

Marsha Anuj Vijay (B150131344)

ABSTRACT

Edge detection is one of the key stages in image processing. In this paper using canny edge detection algorithm, Edges will be detected efficiently with reduction in the processing speed and reduced the memory requirement. Canny edge detection algorithm reduces the latency and increase the throughput with no loss in edge detection performance and algorithm which has the ability to compute edge of multiple blocks at the same time. Canny developed an approach to derive an optimal edge detector for clean and noisy images the canny edge detection algorithm yields better edge detection result.

Experimental results show that the proposed algorithm outperforms other colour image edge detection methods and can be widely used in colour image processing. This project we present a two-dimensional edge detector which gives the edge position in an image The method is used to get the location of the desired object from the edged image and calculate the Cartesian coordinate of the object.

Keywords— Edge Detection, Edge Detection Operator, image processing, Cartesian coordinate, Canny edge detection, hysteresis thresholding, Gaussian filter

INDEX

	Page No.
Title	I
Certificate	II
Acknowledgement	III
Abstract	IV
Table of content	V
List of figures	VI
1.0 Introduction	1
1.1 Introduction to image processing	3
1.2 Steps in image processing	4
1.3 Edge detection technique	5
1.4 Steps in edge detection	6
1.5 Edge detection operators	6
1.5.1 Roberts operator	7
1.5.2 Prewitt operator	8
1.5.3 Sobel operator	9
1.5.4 Laplacian of Gaussian operator	10
1.5.5 Canny edge operator	10
1.5.6 reasons to select Canny edge detection operator	11
2.0 Literature review	12
3.0 Problem identification	13
3.1 Canny edge detection	13
4.0 Methodology	17
5.0 Experiment and result	28
6.0 Conclusion	30
References	31

LIST OF FIGURES

Figure No.	Name of figure	Page no.
1	Flow chart of system	2
2	Image Edge Detection Operator	7
3	Mask used for roberts edge detection technique	8
4	Mask used for roberts Prewitt detection technique	8
5	Mask used for roberts Sobel detection technique	9
6	Masks used for Laplacian of gaussssian edge detection technique.	10
7	Masks used for Canny edge detection technique	11
8	Canny edge detection technique	14
9	Input & output after applying canny algorithm	17
10	Input & output after applying Guassian blur	19
11	Gradient intensity	22
12	Example	22
13	Result of non-maxima supression	25
14	Threshold result: weak pixels in grey & strong one in white	26
15	Result of Hysteresis process	27

1.0 INTRODUCTION

Robotics is an interdisciplinary field that integrates computer science and engineering. Robotics involves design, construction, operation and use of robots. The goal of robotics is to design machines that can help and assist human. Robotics integrates field of mechanical engineering, electrical engineering, information engineering, mechatronics, electronics, bioengineering, computer engineering, control engineering, software engineering among others.

A robot manipulator consists of body and arm in industries to perform repeated and dull jobs. In our project we are trying to integrate manipulator with vision system so that it can function with less human intervention. To achieve this goal, we have divided the complete project in two parts namely

- I. Dynamics and Kinematics
- II. Robot Vision System

In simple terms, a camera placed on the robot end-effector will capture the images of the robot workspace in continuous time and store it. Robot vision system consist of controller, camera and actuator as hardware and image processing algorithm as software takes the image captured and analyse it so as it finds the pre-defines object in the robot workspace. After locating object in the image co-ordinates are find out of the object which are then transferred to the first part of robot, i.e., Dynamics and Kinematics.

Using Inverse Kinematics algorithm joint parameters of robot are calculated and are used in Forward Kinematics algorithm to further achieve the position of manipulator such that the end-effector correctly lie on the head of object. Pick and place operation is being performed on the object.

This type of robot manipulator is used in industries where customization in product is moderate to high. They are also used in sorting operation for industry and commercial use.

A robot vision system consists of one or more cameras, software and a robot. The camera is mounted on the end of end-effector and takes picture of the working area or object to grip and software is used to analyse the image to identify its position, orientation, shape and environment conditions.

This high-level information then can be used to plan the robot motions such as how to grasp the part, how to avoid collisions, time required to perform the operation, etc. With effective use of robot vision system, solutions to automated assemblies can be made robust and more reliable. So, in order to capture the image and process it we have performed following work.

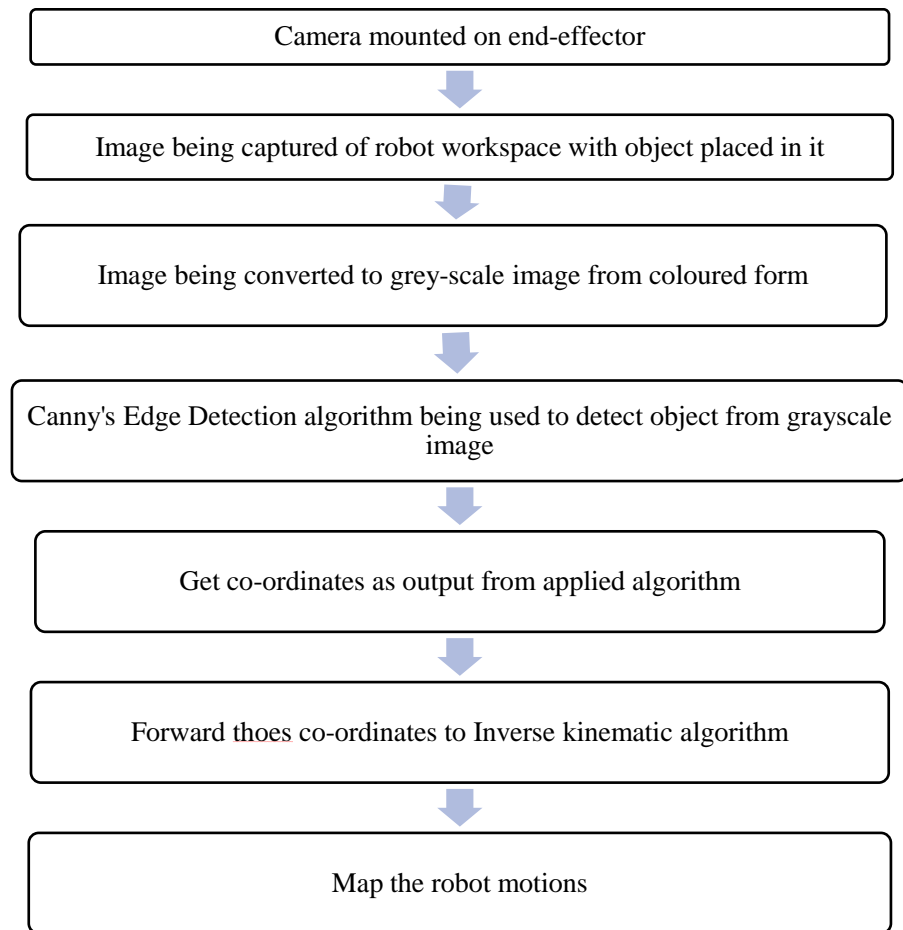


Figure – 1: Flow chart of system

1.1 INTRODUCTION TO IMAGE PROCESSING

Image processing a method of analysing and manipulating the digital images with the computer using mathematical operators. In image processing, the input is an image and outcome may be either set of characteristics or set of the parameter of image or an image. One of the important objectives of image processing is to interpret the content of image efficiently and finds the meaningful and significant information from it. The much awareness has been received from various researchers in the field of image interpretation. One of the most severe steps in image interpretation is to mine the edges information from the image appropriately.

An image comprises various information like contour of the object, its orientation, size and colour. So, as to find the shape information of the object, the edges involving in that object must be identified. Edges are the fundamental features of the image and can be formed from the outlines of the object. Edge detection is a method to detect the occurrence of edges and its locality which is created by sharp and abrupt variations in intensity (brightness or colour) of an image. The discontinuities of an image can be variation in scene illumination, discontinue in scene, surface orientation, its depth and variation in material properties. The objectives of edge detection are to detect the shape information of the object and the reflectance in the image. Edge detection is the important step in image analysing and processing, computer vision, human vision, object detection and pattern recognition.

There are various edge detection techniques for detecting the edges. The different edge detectors work differently. Means some edge detectors take more time and detects more edges with respect to others. The edge detection in an image is rest on intensity, illumination, objects, noise, blur.

1.2 STEPS IN IMAGE PROCESSING

Step 1: Image Acquisition The image is captured by a sensor (eg. Camera), and digitized if the output of the camera or sensor is not already in digital form, using analogue-to-digital convertor

Step 2: Image Enhancement The process of manipulating an image so that the result is more suitable than the original for specific applications. The idea behind enhancement techniques is to bring out details that are hidden, or simple to highlight certain features of interest in an image.

Step 3: Image Restoration - Improving the appearance of an image - Tend to be mathematical or probabilistic models. Enhancement, on the other hand, is based on human subjective preferences regarding what constitutes a “good” enhancement result.

Step 4: Colour Image Processing Use the colour of the image to extract features of interest in an image

Step 5: Wavelets are the foundation of representing images in various degrees of resolution. It is used for image data compression.

Step 6: Compression Techniques for reducing the storage required to save an image or the bandwidth required to transmit it.

Step 7: Morphological Processing Tools for extracting image components that are useful in the representation and description of shape. In this step, there would be a transition from processes that output images, to processes that output image attributes.

Step 8: Image Segmentation procedures partition an image into its constituent parts or objects.

Step 9: Representation and Description - Representation: Make a decision whether the data should be represented as a boundary or as a complete region. It almost always follows the output of a segmentation stage. - Boundary Representation: Focus on external shape characteristics, such as corners and inflections - Region Representation: Focus on internal properties, such as texture or skeleton shape

1.3 EDGE DETECTION TECHNIQUE

The edge forms between the boundaries of two regions. The main information can be mined from the edge. Edge detection is the name for a set of mathematical methods to locate the edges that having good orientation and it is an essential tool of image segmentation or we can say Edge detection is a process that detect the presence and location of edges constitute by sharp changes in colour intensity (or brightness) of an image. It also refers to the process of identifying and locating sharp discontinuities in an image. The discontinuities are rapid changes in pixel intensity which distinguish boundaries of objects in a scene. Since it is verified that the discontinuities in image brightness are likely to correspond. The edge detection process serves to simplify the analysis of images by significantly reducing the amount of data to be processed, while at the same time preserve useful structural information about object boundaries. Edge detection, segments the object while filtering the noise while preserving the structural properties of the image. Edge detection becomes difficult in case of noisy images as noise is also a high frequency content. According to John canny the following three criterions should be well taken care of while edge detection:

- High probability of marking the real edge point and low probability of marking non edge points.
- The points marked as edge points should be as close as possible to the centre of the true edge.
- There should be only one response to a single edge i.e., double line for edges should not be detected.

In the process of edge detection, the image is inputted first and converts that image into grey scale image. And then apply the edge detector to detect and extract the edges present within an image as output. Edge detection is a fundamental tool in image processing, machine vision and computer vision, particularly in the areas of feature detection and feature extraction.

1.4 STEPS IN EDGE DETECTION:

Edge detection include five steps namely- Smoothing, Filtering, Enhancement, Detection and Localization. The overviews of the stepladder in edge detection are as follow:

- **Smoothing:** It contains as much noise as possible, without destroy the true edges.
- **Filtering:** Images are frequently degraded by random variations in intensity values, called noise. Some common types of noise are salt and pepper noise, impulse noise and Gaussian noise. Salt and pepper noise contains indiscriminate occurrences of both black and white intensity values. On the other hand, there is a trade-off between edge strength and noise reduction. More filtering to reduce noise results in a loss of edge strength.
- **Enhancement:** In order to facilitate the detection of edges, it is essential to determine changes in intensity in the neighbourhood of a point. Enhancement emphasizes pixels where there is a significant change in local intensity values and is usually performed by computing the gradient magnitude.
- **Detection:** Many points in an image have a nonzero value for the gradient, and not all of these points are edges for a particular application. Hence, some method should be used to determine which points are edge points. As a rule, thresholding provides the criterion used for detection.
- **Localization:** Determine the exact location of an edge (sub-pixel resolution required for some applications, estimate the location of an edge to Better than the spacing between pixels). Edge thinning and linking are usually essential in this step.

1.5 EDGE DETECTION OPERATORS

The edge demonstration of the image decreases the amount of information to be processed, which contains vital information about the object's shape in an image. Edges are local variations in image intensity. The edge forms between the boundaries of two regions. The main information can be mined from the edge. Edge detection is a process to locate the edges that having good orientation and it is an essential tool of image segmentation. Edge detection method transforms the original image into edge image with the help of operators. It is a well-known process for identifying the dis-continuities in intensity values. In the process of edge detection, the image is inputted first and converts that image into grey scale image. And then apply the edge detector to detect and extract the edges present within an image as output. The various techniques are available for detecting the edges information such as Roberts, Prewitt, Sobel, Laplacian of Gaussian and Canny.

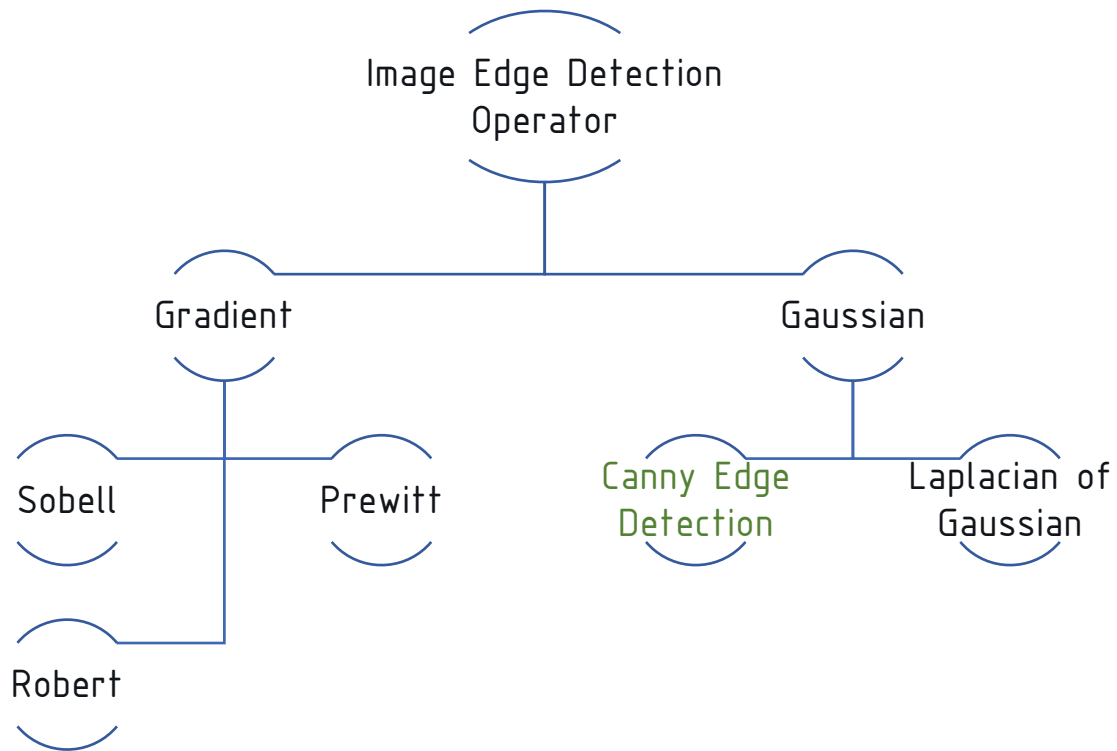


Fig 2 – Image Edge Detection Operator

These techniques are described as follows:

1.5.1 Roberts Edge Detection:

Lawrence Roberts has proposed the Roberts edge detection technique for detecting the edges within an image in 1965. It is a simple and computationally efficient approach. It measures the spatial gradient of an image. The pixel value at that point in the resultant image characterizes estimated absolute magnitude value of the spatial gradient of the inputted image at that point. It takes input image as grey scale image and produces edges involving in that image. The main disadvantages of this technique are that it can't detect that type of edges which are multiplies of 45 degrees and it is not symmetric. This operator produces the position of edges more accurately, but it has the short support of filters which causes vulnerability to noise.

<table><tr><td>+1</td><td>0</td></tr><tr><td>0</td><td>-1</td></tr></table> <div>G_x</div>	+1	0	0	-1	<table><tr><td>0</td><td>+1</td></tr><tr><td>-1</td><td>0</td></tr></table> <div>G_y</div>	0	+1	-1	0
+1	0								
0	-1								
0	+1								
-1	0								

Fig 3 – Mask used for roberts edge detection technique

The partial derivative of Robert operator is given as follows:

$$\frac{\partial f}{\partial x} = f(i, j) - f(i + 1, j + 1)$$

$$\frac{\partial f}{\partial y} = f(i + 1, j) - f(i, j + 1)$$

This operator produces the position of edges more accurately, but it has the short support of filters which causes vulnerability to noise.

1.5.2 Prewitt Edge Detection :-

Prewitt has proposed the Prewitt edge detection technique in 1970. It is a right algorithm to measure the magnitude and orientation of the edges. This technique evaluates the edge directions directly with the maximum response from the mask. It is having 8 directions. But, sometimes most direct directions approximation are not much perfect. This Prewitt operator is just like a Sobel operator and easy to implement than Sobel operator but it produces some times noisier results.

-1	0	+1
-1	0	+1
-1	0	+1

G_x

+1	+1	+1
0	0	0
-1	-1	-1

G_y

Fig 4 : Masks used for Prewitt edge detection technique.

The organization of pixels about central pixel is as follows:

$$\begin{bmatrix} a_0 & a_1 & a_2 \\ a_7 & [i-j] & a_3 \\ a_6 & a_5 & a_4 \end{bmatrix}$$

The Prewitt edge detector is less vulnerable to noise because it differentiates in one direction and make average in another direction.

$$G_x = (a_2 + ca_3 + a_4) - (a_0 + 2a_7 + a_6)$$

$$G_y = (a_6 + ca_5 + a_4) - (a_0 + 2a_1 + a_2)$$

The partial derivatives of Prewitt operator are measured as:

1.5.3 Sobel Edge Detection :-

Irwin Sobel has proposed the Sobel edge detection technique in 1970. The

Sobel kernel depends on the central difference, but while averaging it gives more weight to central pixel. One of the advantages of Sobel kernel over Prewit kernel is that it has better noise suppression characteristics. This mask can deal with the edges which are running 45 degrees to the pixel grid. This mask can be put on distinctly to the input image to give gradient components in every orientation.

-1	-2	-1
0	0	0
+1	+2	+1
G_x		

-1	0	+1
-2	0	+2
-1	0	+1
G_y		

Fig 5: Masks used for Sobel edge detection technique.

The partial derivative of Sobel operator is given as follows:

$$G_x = (a_2 + 2a_3 + a_4) - (a_0 + 2a_7 + a_6)$$

$$G_y = (a_6 + 2a_5 + a_4) - (a_0 + 2a_1 + a_2)$$

The Gradient magnitude is as follows:

$$|G| = \sqrt{G_x^2 + G_y^2}$$

The Orientation angle is measured as follows:

$$\theta = \arctan\left(\frac{G_y}{G_x}\right) - \frac{3\pi}{4}$$

1.5.4 Laplacian of Gaussian (LOG) Edge Detection :-

Marr has introduced the Laplacian of Gaussian (LOG) technique in (1982). LOG is based on second order derivative. Which stated as:

$$\nabla^2 f = \frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2}$$

LOG smoothes the image first then calculate Laplacian. This process produces the double edge image. It locates edges then search the zero crossing between the double edges. The LOG edge detection method contains the pair of 3x3 convolution mask which is illustrated in Figure:5

+1	+1	+1	-1	+2	-1
+1	-8	+1	+2	-4	+2
+1	+1	+1	-1	+2	-1
G_x			G_y		

Fig 6 : Masks used for Laplacian of gaussian edge detection technique.

1.5.5 Canny Edge Detection :-

John Canny introduced the canny edge detection technique at MIT in 1983.

It is the standard, powerful and usually used edge detection method. It separates the noise from the image before extracting edges. Canny is a better method for extracting the edges than other existing methods and produces the good result. The Canny operator can control a number of details of edge image and can suppress the noise efficiently.

-1	0	+1		+1	+2	+1
-2	0	+2		0	0	0
-1	0	+1		-1	-2	-1
G_x				G_y		

Fig 7 : Masks used in Canny edge detection technique.

1.5.6 Reason to select Canny edge detection operator

- The presence of Gaussian filter allows removing of any noise in an image
- The signal can be enhanced with respect to the noise ratio by non-maxima suppression method which results in one-pixel wide ridges as the output.
- Detects the edges in a noisy state by applying the thresholding method
- The effectiveness can be adjusted by using parameters
- It gives a good localization, response and is immune to a noisy environment.

2.0 LITERATURE REVIEW:

1. In [1], Radhika Chandwadkar et al performed a comparative study on edge detection technique. They claimed canny edge detection overcomes the drawback of Sobel edge detection and they concluded canny edge detection is better for object recognition and pattern matching.
2. Bing Wang et al proposed an improved canny algorithm in [2]. In this proposed algorithm, Gaussian filter is replaced by self-adaptive filter, morphological thinning is used to thin the edge and morphological operator is used to achieve the refining treatment of edge point's detection and the single pixel level edge.
3. In [3] Cai-Xai Deng et al proposed an improved edge detection algorithm. The proposed algorithm is mainly focused on improvising the ability of canny operator in terms of selection of variance, edge preserving and denoising effects. In this algorithm morphological filter replaces Gaussian filter and conclusion is given that proposed algorithm can filter salt and pepper noise effectively and edge detection accuracy is improved.
4. G. T Shrinivakshan et al conducted an experiment to understand fundamental concept of various filter in [4]. They implemented experiment on a image using matlab. The International Journal of Computer Applications (0975 – 8887) Volume 116 – No. 9, April 2015 39 difference between gradient and laplacian operator is identified. The advantages and disadvantages of filter are studied
5. In [5] Weibei Rong et al proposed a new canny edge detection algorithm to overcome the drawback of traditional canny edge detection algorithm like sensitive to noise and losing weak edges. Here they replaced image gradient with concept of gravitational field intensity. Two adaptive thresholds are used, one is based on image gradient magnitude and another on standard deviation. The concluded proposed algorithm holds good for all the kind of images, simple and easy to implement.
6. In [6] Poonam Dhankhar, Neha Sahu et al proposed through study has been done on most commonly used edge detection techniques such as Sobel, Prewitt, Roberts, Canny, Laplacian Gaussian(LoG).

3.0 PROBLEM DEFINATION

The current work proposes a model for Edge detection using the Canny Edge Detection method. Edge Detection is all about identifying the object present in the camera frame and check its size for the given size by locating its edge.

In this example, the image captured is converted to grey-scale image which is of nut with different sizes and the canny algorithm is applied over it. Edge of the nut is mapped through the Canny edge detection and area inside the edge is calculated which gives the approximate estimate of the size of the nut. This led to the sorting operation which is our application to place the right size of nut in the respective container.

We have prepared a python program using OpenCV library to perform above operation.

Following is an image captured from the camera mounted on end-effector. The results are shown in Experiment and Result section.

3.1 Canny Edge Detection:

The Canny edge detection algorithm is known to many as the optimal edge detector. Canny's intentions were to enhance the many edge detectors already out at the time he started his work. He was very successful in achieving his goal and his ideas and methods can be found in his paper, "A Computational Approach to Edge Detection". In this project followed a list of criteria to improve current methods of edge detection. The first and most obvious is low error rate. It is important that edges occurring in images should not be missed and that there be NO responses to non-edges. The second criterion is that the edge points be well localized. In other words, the distance between the edge pixels as found by the detector and the actual edge is to be at a minimum. A third criterion is to have only one response to a single edge.

Based on these criteria, the canny edge detector first smoothes the image to eliminate and noise. It then finds the image gradient to highlight regions with high spatial derivatives. The algorithm then tracks along these regions and suppresses any pixel that is not at the maximum (non-maximum suppression).

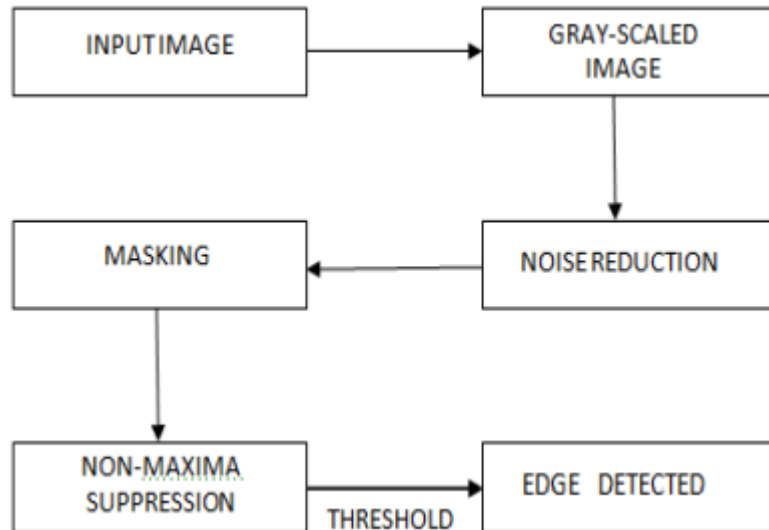


Fig 8 : Canny edge detection technique.

The Canny algorithm runs in five separate steps:

➤ **Noise detection by Smoothing :-**

The outcomes of edge detection are extremely sensitive to image noise because the mathematics involved behind the scenes is essentially based on derivatives. One method to obtain rid of the noise on the image is by implementing a Gaussian blur to smooth it. To accomplish this, the image concentration method with a Gaussian kernel (3x3, 5x5, 7x7, etc.) is used. The size of the core depends on the trace of the expected blur. Fundamentally, the less visible is the blur for the smallest kernel. In this step, first, the input image needs to be converted into a grayscale by adjusting contrast and brightness so that the image is blurred to eliminate noise. Therefore, for making the location and detection edge effective, the first step is to filter for removing noise in the main image. Commonly, it is used as a Gaussian filter for noise elimination.

$$G(x, y) = \frac{1}{2\pi\sigma^2} \exp\left(-\frac{x^2 + y^2}{2\sigma^2}\right)$$

Where y is the distance from the origin in the vertical axis, x is the distance from the origin in the horizontal axis and σ is the standard deviation of the Gaussian distribution.

➤ Calculate image gradient :-

The gradient computation step by calculating the image gradient using the Canny edge detection operator detects the edge and direction intensities. By calculating the image gradient, the edge pixels with drastic changes in gray area values are identified. Since points in the direction of most intensity variation represent so the gradient is a unit vector. At this stage, first, the vertical and horizontal ingredients of the gradient are calculated and then it is computed the magnitude and orientation of the gradient. The gradient magnitude G and gradient angle Θ are calculated as follows

$$\text{Gradient Magnitude} = G = \sqrt{(G_x^2 + G_y^2)}$$

$$\text{Gradient Angle} = \theta = \arctan\left(\frac{G_y}{G_x}\right)$$

G_x and G_y represent the horizontal and vertical gradient respectively.

Canny edge detection algorithm utilizes four filters to calculate diagonal, vertical and horizontal edges in the blurred image. Also, the edge direction angle is rounded to one of the four angles demonstrating vertical, horizontal and the two diagonals (0° , 45° , 90° , 135° degrees). It results to estimate the first derivative in the vertical direction G_y and horizontal G_x . Then, the edges where the intensity of the gray level changes the most are detected by the Canny algorithm.

➤ Non-maxima suppression (NMS) :

This method is based on one of the two techniques generally employed for edge detection, the first one is to consider edges as the zero-crossings of the Laplacian of image intensity The second one is to suppress the local non-maxima of the magnitude of the gradient of image intensity in the direction of this gradient also this method called NMS. Ideally, the ultimate image should possess slim edges. Thus, to thin the edges, we need to fulfil NMS. Also, NMS can effectively find the edge and repress the occurrence of incorrect edges.

NMS is also based on the gradient magnitudes that the detector converts the thick edges of the image, to nearly thin and sharp edges which can be more utilized for identification purposes. It is mainly performed in NMS for thinning the edge. In this process, the image is scanned along the edge direction and rejects any pixel value that is not considered to be an edge which will result in a thin line in the output image.

➤ **Double Thresholding :-**

The threshold value based on 2 parts, $T1$ = high threshold, $T2$ = low threshold. If the pixels possessing values of a grayscale level higher than $T1$ are strong edge pixels, the result is edge area. The result is depending on the neighbouring pixels. If the pixels possess values of the grayscale level among $T1$ and $T2$ the stage goal is at identifying three types of pixels: strong, weak, and non-relevant. In this paper, we used this step as follows:

- Pixels which they have an intensity value and not enough to be considered, are weak pixel, but they are small enough to be considered as irrelevant edge detection.
- Pixels with high intensity are strong ones in which they participate in the final edge.
- Other pixels contribute as non-relevant for the edge.
- Also, we can see what the double thresholds maintain for:

The low threshold is applied to detect unrelated pixels (intensity less than the low threshold).

- Pixels which they have an intensity value and not enough to be considered, are weak pixel, but they are small enough to be considered as irrelevant edge detection.
- The high threshold is utilized to recognize the strong pixels (intensity higher than the high threshold)
- All pixels possessing intensity among both thresholds are flagged as weak and the Hysteresis system (next stage) will aid us to identify the ones that could be regarded as strong and the ones that are regarded as irrelevant.

➤ **Track edge by hysteresis :-**

Edges will remove in the output image if they do not link to a very definitive edge. If weak edges that are connected with strong edges, they will contain in the ultimate image. Strong edges are represented as certain edges and are contained in the final edge image. A pixel is labelled as a strong edge pixel if its gradient magnitude is bigger than the upper threshold. Whereas if the value of the gradient magnitude of the pixel falls among the lower and higher threshold, a pixel is classified as a weak edge pixel. Strong edges are the ones which can be included immediately as edges in the final edge image. On the other hand, if weak edges are linked to the strong edges, they can be marked.

4.0 METHODOLOGY

The idea in this project is to build an algorithm that can sketch the edges of any object present on a picture, using the Canny edge detection algorithm. First of all, let's describe what is the Canny Edge Detector:

The Canny edge detector is an edge detection operator that uses a multi-stage algorithm to detect a wide range of edges in images. It was developed by John F. Canny in 1986. Canny also produced a computational theory of edge detection explaining why the technique works.

The Canny edge detection algorithm is composed of 5 steps:

- Noise reduction;
- Gradient calculation;
- Non-maximum suppression;
- Double threshold;
- Edge Tracking by Hysteresis.

After applying these steps, you will be able to get the following result:

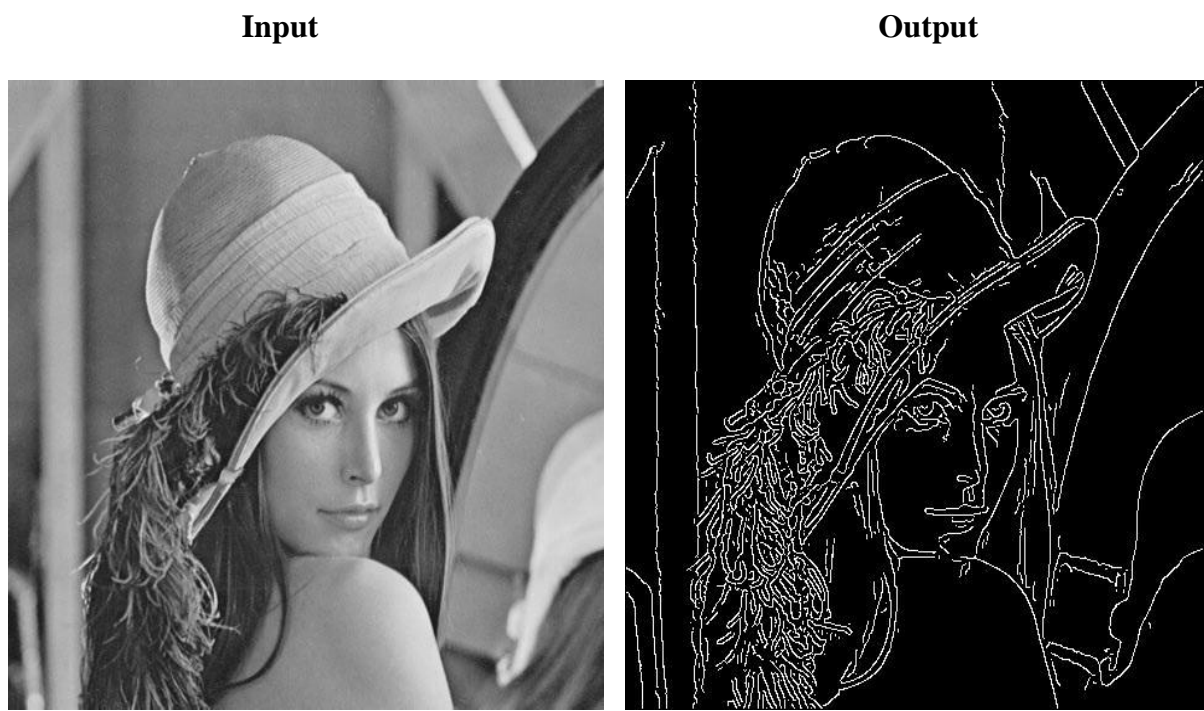


Fig 9 : Input and output after applying Canny algorithm.

The canny algorithm is based on grey-scaled pictures. So the pre-requisite is to convert the original image to grey-scale image.

Noise reduction

An image is a picture, photograph or any other form of 2D representation of any scene. Image noise is random variation of brightness or colour information in images, and is usually an aspect of electronic noise. It can be produced by the image sensor and circuitry of a scanner or digital camera. Image noise can also originate in film grain and in the unavoidable shot noise of an ideal photon detector.

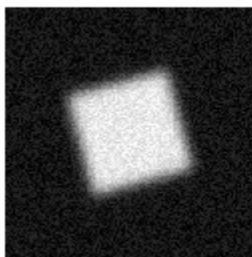
Most algorithms for converting image sensor data to an image, whether in-camera or on a computer, involve some form of noise reduction. There are many procedures for this, but all attempt to determine whether the actual differences in pixel values constitute noise or real photographic detail, and average out the former while attempting to preserve the latter. However, no algorithm can make this judgment perfectly (for all cases), so there is often a trade-off made between noise removal and preservation of fine, low-contrast detail that may have characteristics similar to noise.

One way to get rid of the noise on the image, is by applying Gaussian blur to smooth it. To do so, image convolution technique is applied with a Gaussian Kernel (3x3, 5x5, 7x7 etc...). The kernel size depends on the expected blurring effect. Basically, the smallest the kernel, the less visible is the blur. In our example, we will use a 5 by 5 Gaussian kernel.

The equation for a Gaussian filter kernel of size $(2k+1) \times (2k+1)$ is given by:

$$H_{ij} = \frac{1}{2\pi\sigma^2} \exp\left(-\frac{(i - (k+1))^2 + (j - (k+1))^2}{2\sigma^2}\right); 1 \leq i, j \leq (2k+1)$$

noisy image



Canny filter, $\sigma = 1$ Canny filter, $\sigma = 3$



After applying the Gaussian blur, we get the following result:



Fig 10 : Input and output after applying Gaussian blur.

Gaussian Filter

In electronics and signal processing, a Gaussian filter is a filter whose impulse response is a Gaussian function (or an approximation to it, since a true Gaussian response is physically unrealizable as it has infinite support). Gaussian filters have the properties of having no overshoot to a step function input while minimizing the rise and fall time. This behaviour is closely connected to the fact that the Gaussian filter has the minimum possible group delay. It is considered the ideal time domain filter, just as the sinc is the ideal frequency domain filter. These properties are important in areas such as oscilloscopes and digital telecommunication systems.

Mathematically, a Gaussian filter modifies the input signal by convolution with a Gaussian function; this transformation is also known as the Weierstrass transform.

The one-dimensional Gaussian filter has an impulse response given by

$$g(x) = \sqrt{\frac{a}{\pi}} e^{-ax^2}$$

and the frequency response is given by the Fourier transform

$$g(x) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{x^2}{2\sigma^2}}$$

with f the ordinary frequency. These equations can also be expressed with the standard deviation as parameter

$$g(x) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{x^2}{2\sigma^2}}$$

and the frequency response is given by

$$\hat{g}(f) = e^{-\frac{f^2}{2\sigma_f^2}}$$

By writing α as a function of σ with the two equations for $g(x)$ and as a function of σ_f with the two equations for $\hat{g}(f)$ it can be shown that the product of the standard deviation and the standard deviation in the frequency domain is given by

$$\sigma\sigma_f = \frac{1}{2\pi},$$

where the standard deviations are expressed in their physical units, e.g., in the case of time and frequency in seconds and hertz, respectively.

In two dimensions, it is the product of two such Gaussians, one per direction:

$$g(x, y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}}$$

where x is the distance from the origin in the horizontal axis, y is the distance from the origin in the vertical axis, and σ is the standard deviation of the Gaussian distribution.

Gradient Calculation

Image gradients are very strong cue for detecting edges within image. This is because pixels belonging to one object are likely to have similar pixel intensities, resulting low gradient magnitudes among those pixels. Boundaries between objects, On the other hands, often have sharp transition in pixel intensities and this results high gradient magnitudes, and the boundaries correspond to edges we want to find. Therefore, by finding pixels within image whose gradient magnitude has a high value, we can locate edges with high probability.

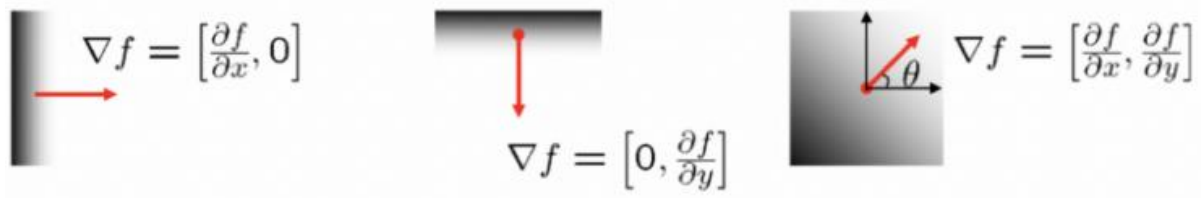
The Gradient calculation step detects the edge intensity and direction by calculating the gradient of the image using edge detection operators.

In order to compute the image gradients for each pixel, x-derivative of Gaussian and y-derivative of Gaussian filters are used. Then the magnitude and orientation maps are built based on computed derivatives with respect to x and y for each pixel, as follows.

- The gradient of an image

$$\nabla f = \left[\frac{\partial f}{\partial x}, \frac{\partial f}{\partial y} \right]$$

- The gradient points in the direction of most rapid intensity change



- The gradient direction (orientation of edge normal) is given by:

$$\theta = \tan^{-1} \left(\frac{\partial f}{\partial y} / \frac{\partial f}{\partial x} \right)$$

- The edge strength is given by the gradient magnitude

$$\|\nabla f\| = \sqrt{\left(\frac{\partial f}{\partial x} \right)^2 + \left(\frac{\partial f}{\partial y} \right)^2}$$



Fig 11 : Gradient intensity

Non-maximum Suppression

Ideally, the final image should have thin edges. Thus, we must perform non-maximum suppression to thin out the edges.

The principle is simple: the algorithm goes through all the points on the gradient intensity matrix and finds the pixels with the maximum value in the edge directions.

Let's take an example:

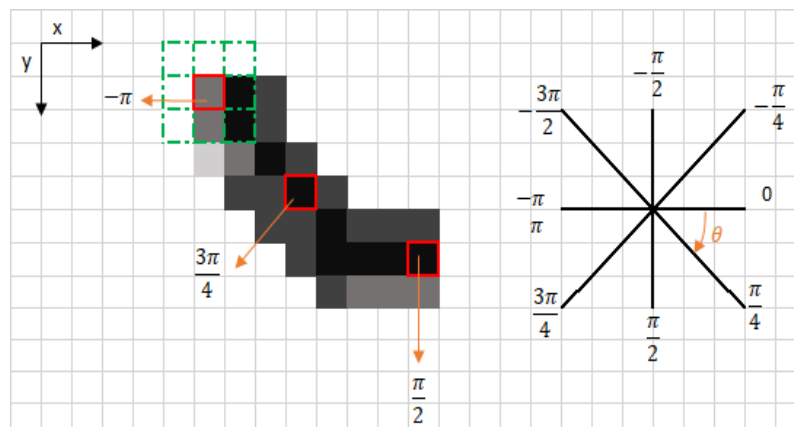
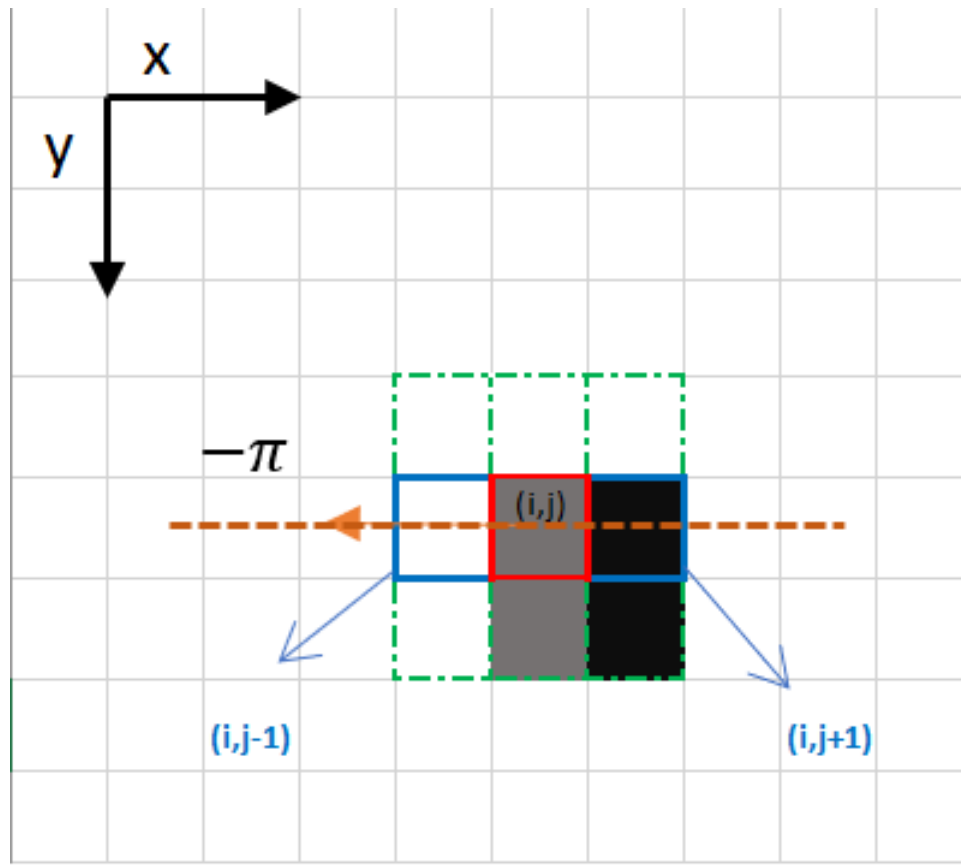


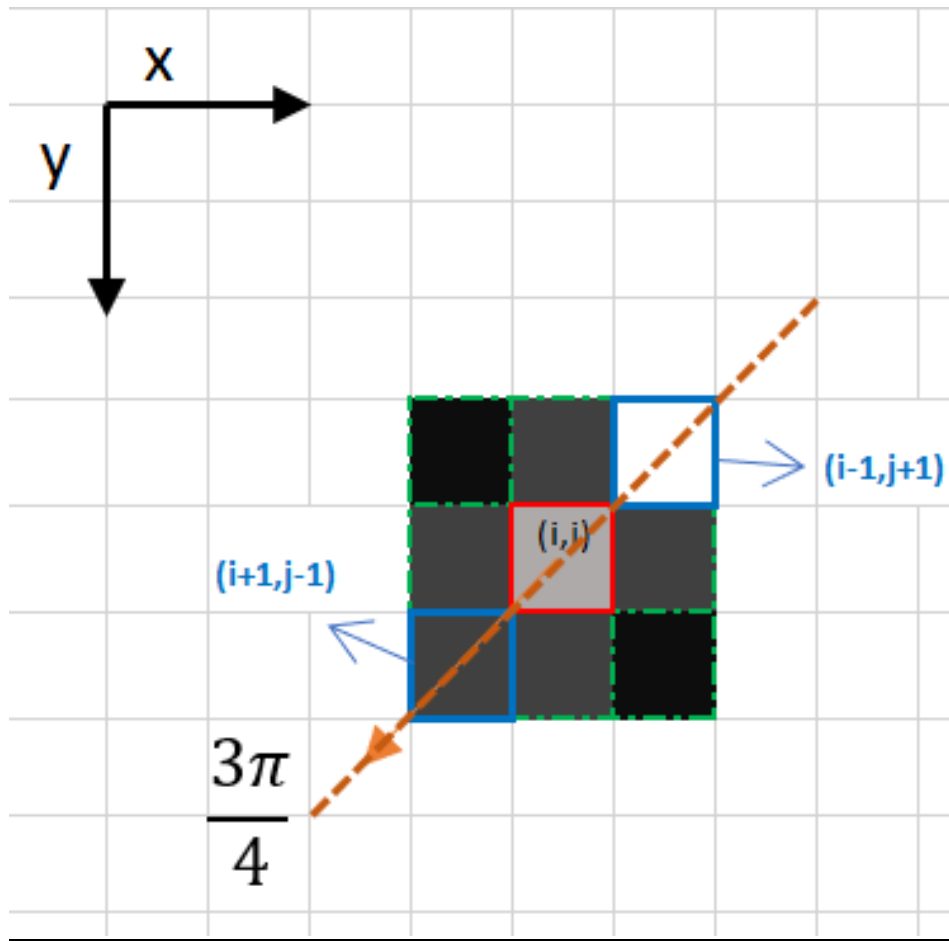
Fig 12 : Example.

The upper left corner red box present on the above image, represents an intensity pixel of the Gradient Intensity matrix being processed. The corresponding edge direction is represented by the orange arrow with an angle of $-\pi$ radians (± 180 degrees).



The edge direction is the orange dotted line (horizontal from left to right). The purpose of the algorithm is to check if the pixels on the same direction are more or less intense than the ones being processed. In the example above, the pixel (i, j) is being processed, and the pixels on the same direction are highlighted in blue $(i, j-1)$ and $(i, j+1)$. If one of those two pixels is more intense than the one being processed, then only the more intense one is kept. Pixel $(i, j-1)$ seems to be more intense, because it is white (value of 255). Hence, the intensity value of the current pixel (i, j) is set to 0. If there are no pixels in the edge direction having more intense values, then the value of the current pixel is kept.

Let's now focus on another example:



In this case the direction is the orange dotted diagonal line. Therefore, the most intense pixel in this direction is the pixel $(i-1, j+1)$.

Let's sum this up. Each pixel has 2 main criteria (edge direction in radians, and pixel intensity (between 0–255)). Based on these inputs the non-max-suppression steps are:

- Create a matrix initialized to 0 of the same size of the original gradient intensity matrix;
- Identify the edge direction based on the angle value from the angle matrix;
- Check if the pixel in the same direction has a higher intensity than the pixel that is currently processed;
- Return the image processed with the non-max suppression algorithm.

The result is the same image with thinner edges. We can however still notice some variation regarding the edges' intensity: some pixels seem to be brighter than others, and we will try to cover this shortcoming with the two final steps.



Fig 13 : Result of non-maximum suppression.

Double Threshold

The double threshold step aims at identifying 3 kinds of pixels: strong, weak, and non-relevant:

- Strong pixels are pixels that have an intensity so high that we are sure they contribute to the final edge.
- Weak pixels are pixels that have an intensity value that is not enough to be considered as strong ones, but yet not small enough to be considered as non-relevant for the edge detection.
- Other pixels are considered as non-relevant for the edge.

Now you can see what the double thresholds holds for:

- High threshold is used to identify the strong pixels (intensity higher than the high threshold)
- Low threshold is used to identify the non-relevant pixels (intensity lower than the low threshold).
- All pixels having intensity between both thresholds are flagged as weak and the Hysteresis mechanism (next step) will help us identify the ones that could be considered as strong and the ones that are considered as non-relevant.

The result of this step is an image with only 2-pixel intensity values (strong and weak):



Fig 14 : Threshold result: weak pixels in grey and strong one in white.

Edge tracking by Hysteresis

Based on the threshold results, the hysteresis consists of transforming weak pixels into strong ones, if and only if at least one of the pixels around the one being processed is a strong one, as described below:

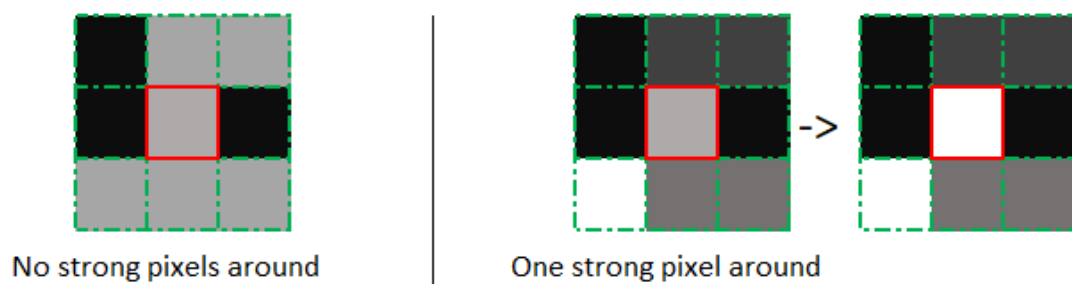




Fig 15 : results of Hysteresis process.

5.0 EXPERIMENT AND RESULT

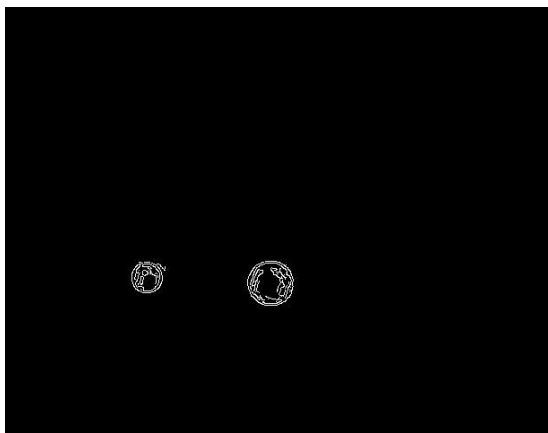
In the experimental setup we have considered sorting of nuts of different size .In other words we are try to sort out the oversize nut .The image shown below have two nuts of size 5 mm and 10 mm .we have carried out experiment to sort the nut of size greater than 5mm .The results are shown below.



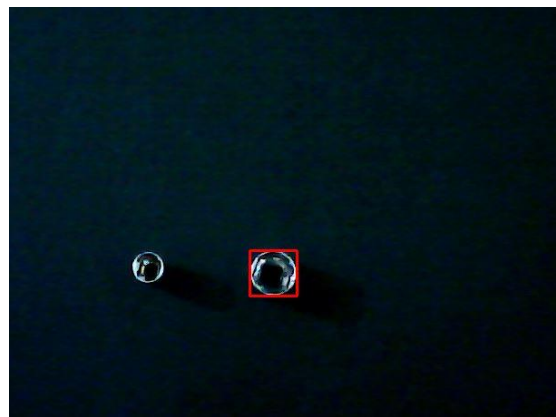
Input Image



Gray Scale Image



Edged Image



Result Image

To calculate x and y location

We have pixel coordinates of top left and bottom right corner of the bounding rectangle shown in red colour.

Top left = 289,286

Bottom right = 341,336

Centre of rectangle = 315,311

The total pixel in vertical direction is 480 which corresponds to 160 mm length of workplace

Therefore, x co-ordinate = $315 \times (16/480)$

$$= 10.5 \text{ cm}$$

The total pixel in horizontal direction is 640 which corresponds to 200 mm length of workplace

y co-ordinate = $311 \times (20/640)$

$$= 9.7 \text{ cm}$$

These coordinates are with respect to camera frame to convert it to robot base frame multiply it with homogenous transformation matrix between robot base frame and camera frame.

We have assumed $\Theta = 180$ and $\alpha = 90$ and translation along x and y to be -1.5 and -0.5 respectively. The obtained results are shown below

```
C:\Users\Lenovo\Desktop\Project>final1.py
Enter the Size to be sort: 10
top left (289, 286) bottom right (341, 336)
X_cordinate is [8.86666667] Y_cordinate is [9.34375]
```

6.0 CONCLUSION

The vision system for a robot does the task to find out the Cartesian coordinates of the desired object.

The canny edge detector technique was used to find out the edges in the image

Using the detected edges, a contour is drawn to calculate its area.

In the vision system the following operation were performed.

- Capture the image of the workplace
- Convert it to grayscale
- Find out the edges using canny edge detection technique
- Using the external boundaries, we found out the area of the object
- By comparing the area with the desired area, the oversized nut is detected
- By converting pixel to cm the Cartesian coordinates of the object are found

References

- [1] Radhika Chandwadkar, Saurabh Dhole, Vaibhav Gadewar, Deepika Raut, Prof. S. A. Tiwaskar “Comparison Of Edge Detection Techniques” Proceedings Of Sixth Iraj International Conference, 6th October 2013, Pune.
- [2] Bing Wang, Shaosheng Fan “An Improved canny Edge Detection Algorithm” Second International Workshop On Computer Science And Engineering 2009
- [3] Cai-Xia Deng, Gui-Bin Wang , Xin-Rui Yang, “Image Edge Detection Algorithm Based on Improved Canny Operator”, Proceedings of the 2013 International Conference on Wavelet Analysis and Pattern Recognition, Tianjin, July, 2013.
- [4] G.T. Shrivakshan, Dr.C.Chandrasekar , “A Comparison of various Edge Detection Techniques used in Image Processing”, IJCSI International Journal of Computer Science Issues, Vol. 9, Issue 5, No 1, September 2012..
- [5] Weibin Rong, Zhanjing Li, Wei Zhang and Lining Sun, “An Improved Canny Edge Detection Algorithm”, Proceedings of 2014 IEEE International Conference on Mechatronics and Automation, Tianjin, China August 3 - 6, 2014.
- [6] Poonam Dhankhar¹ , Neha Sahu² ¹ Student, ITM University, Gurgaon-Haryana, India ² Faculty of CSE, ITM University, Gurgaon-Haryana, India