IFERP
connecting engineers... developing research

ISSN (Online) 2456-1290

International Journal of Engineering Research in Mechanical and Civil Engineering (IJERMCE)
Vol 2, Issue 3, March 2017

Detection and Tracking of Wheeled Mobile Robot by Image Processing

[1] Jennifer Jacob, [2] Robins Mathew, [3] Somashekhar S Hiremath*, and [4] Chandrashekhar Bhat
[1], [4] Department of Mechatronics Engineering, Manipal Institute of Technology, Manipal-576104
[2], [3] Department of Mechanical Engineering, IIT Madras, Chennai-600036

***Abstract*- Mobile robotics is a fast developing area due to their wide applications in exploration on-ground, under-ground, on-water, under-water and space. Detection and tracking of the paths followed by these robots are required to analyze their motion and to guide them through optimal path. This paper focuses on the application of image-processing for estimating the position and velocity of mobile robot on an indoor workspace. The motion of the robot on the workspace was captured using monocular vision system. In order to detect the path of the mobile robot, the obtained image was processed using a full featured high-level programming language-MATLAB. This work is fully dependent on how well the feature of the robot in the image plane was extracted and tracked from the initial frame to the subsequent frames. Thus, color-based feature extraction and blob analysis are used to serve the purpose of detecting the path of the mobile robot. The resulting data was further used to obtain path coordinates which in turn gives the position and velocity errors of the robot. A calibration phase was carried out to analyze the relationship between the measured coordinates (pixels) with the world coordinates. Some of the challenges faced during implementation of the system are also mentioned in the paper.**

Index Terms: **Blob analysis, Feature-extraction, Image-processing, Position estimation, Velocity estimation, Wheeled mobile robot**

I. INTRODUCTION

With the advent of human civilization and tremendous development in technology, man is trying to establish a natural interaction with different machines and gadgets to have more automation. Any gadget is made to act upon the human instructions through exchange of data between the gadget and its user. This data can be in any form such as an electrical signal, speech, visual images, numerical values and so forth. The recent years have shown the importance of data in the form of images or video captured by camera in numerous fields including video conferencing, medical image processing, surveillance, movement-recognition, object recognition, object tracking, etc.

Object tracking refers to act of archiving the position of an object at any instant of time and tracking changes of its position over sequence of images. An effective tracking algorithm should overcome many real circumstances such as background clutters, occlusions, and different illuminations. There are numerous techniques for detecting and tracking object/s of interest in the captured scene/s using the platform of image processing. Models of image background-scene and foreground can be used for the detection of moving objects as described by [1], [2]. Ellipsoidal targets in an image can be tracked by kernel-based visual tracker which uses Bhattacharya coefficient to find similarity between the target ellipsoidal model and the candidate [3]. This approach has complex computation, though it provided good performance in most of the image sequencing. Another technique was put forward which includes using tunable kernels with mean-shift algorithm for tracking human motion as detailed [4]. Gaussian Mixture Model (GMM) and background modeling are also emerging techniques for the detection of moving objects. This technique can track and segment multicolored objects [5]. Boundary-based and region-based information can be utilized to detect several non-rigid objects [6]. Shape, contour, edges and color are some of the important spatial features of any object. Among them, color acts as a powerful descriptor to identify and extract objects of interest from an image. Tracking of object using color information has been very effective with less processing-time as discussed [7]. The distinctive color of the target object can serve as an appropriate means of locating the object in the captured scene. This technique was adopted in the current work.

The fundamental part of image-based detection and tracking starts with the study of image-formation from 3D plane to the image-plane. An image capturing device is composed of a single microchip in which an array of photosensitive elements is arranged which acts as image sensor. When the light waves (photons) hit these photosensitive elements, the state of the elements change due to the photoelectric effect [8]. This change gives rise to image formation. The basic element of this image is called pixel and the number of pixels of the image is known as its resolution. Fig.1 shows the block diagram of an image acquisition system.

ISSN (Online) 2456-1290
International Journal of Engineering Research in Mechanical and Civil Engineering (IJERMCE)
Vol 2, Issue 3, March 2017

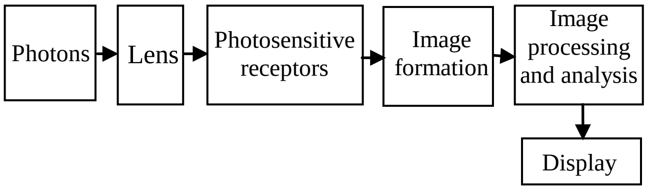


Fig.1 Block diagram of an image processing system

Literatures focuses on many techniques and approaches for detection and tracking of stationary as well as mobile objects based on distinctive features of the objects and motion observed which are of great importance depending upon the application. This paper discusses the application of image-processing to detect and track wheeled mobile robot moving over a two-dimensional platform. The robot in the image was detected and its motion was tracked continuously using the developed vision-based measurement system. The detection of the robot in the image was based on RGB (Red, Green, Blue) color-model. Blob analysis was used for discerning the robot from the background. After further processing, the position and velocity of the robot in motion were obtained along with its trajectory. The programming platform, MATLAB v8.6 (R2015b) was used as a tool for image processing and analysis.

The remainder of the paper is organized as: section II talks about the detection and tracking procedure in detail. Section III explains the calibration phase. Section IV explains the results of the performed experiments while section V contains conclusion and future work.

II. ROBOT DETECTION AND TRACKING

The experimental setup which was developed is shown in Fig. 2. The mobile robot used for the experiment was e-puck which is a mini mobile robot/ research platform [10].This robot has a diameter of 70 mm and the two-dimensional platform is of size 1600 x 800 mm. To obtain the relevant data within less time is an important factor to be taken care of. Any data or information is converted from its raw form to a useful one after going through different stages of processing. In the case of an image-data, more resolution means more number of pixels which in turn gives more quality to the image. But as the number of pixels increases, the time taken to process an image or sequence of images also increases. Thus, considering the quality of the image and the processing-time, an optimal resolution of 640 x 360 with a frame rate of 30 frames per second (fps) was chosen.

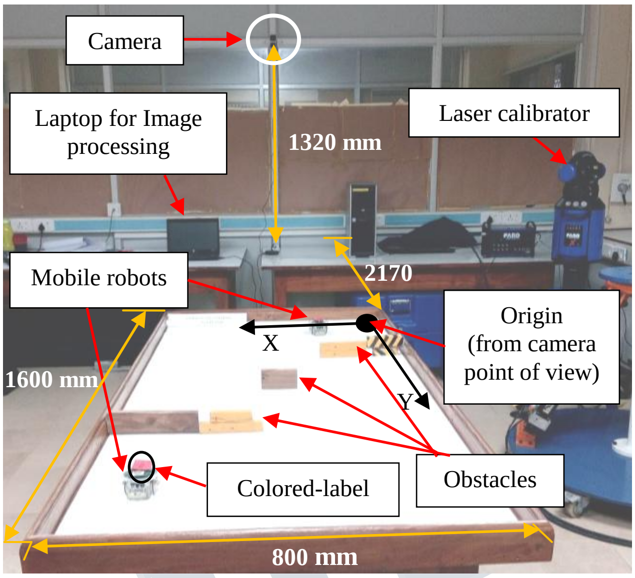


Fig. 2 Vision-based measurement system

There are various tracking techniques using mechanical tracker, ultrasonic trackers, magnetic trackers, radio and microwave trackers, etc that can be exploited to detect and track the position and orientation of an object. These trackers involve placing of devices (such as sensors or transmitters) on the object to be tracked as mentioned [9]. In the current work, there was no involvement of any such hardware.

The movement of the mobile robot on the platform was captured continuously using a camera mounted on the wall. A colored-label stuck on top of the mobile robot acts as a distinctive feature which had to be detected by the image-processing algorithm. The obtained-image was processed through different stages of image-processing as mentioned in Fig. 3. The centroid of the colored-label in pixel coordinates was obtained through image processing as the robot moved in a pre-defined path from the source point to the destination point. The obtained pixel coordinates was further transformed into world coordinates to obtain actual position of the robot. Each stage of processing phase is explained below in detail.

A. *Image acquisition*

When the robot was in motion, a video was acquired using a high definition camera which was connected to the image-processing system (laptop) via a USB cable. Each frame of this video acts as an image during the processing. The properties such as brightness, exposure, focus-mode, etc. of the camera were set ahead of capturing. The camera facing the platform, was mounted on the wall at a height of 1320 mm (1.32 m) and at a linear distance of 2170 mm (2.17 m) from the two-dimensional platform as shown in Fig. 3.

IFERP
connecting engineers... developing research
ISSN (Online) 2456-1290
International Journal of Engineering Research in Mechanical and Civil Engineering (IJERMCE)
Vol 2, Issue 3, March 2017

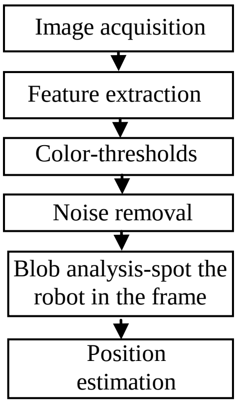


Fig.3 Different stages of processing the captured frame

B. Feature extraction

Extraction of relevant information from an image is a crucial step in order to classify the object of interest from the other objects in the same image. Selection of a satisfactory and acceptable feature extraction technique according to the input should be done with extreme care [11]. This extraction can be based on motion classification, object classification or simply on features like color, shape, edge, and so forth. Color is widely used as a distinct feature for object detection and tracking by many researchers [5], [12], [13]. Present approach utilizes color as the feature to be extracted from the captured-image.

There are many color models available out of which the popular ones are RGB (Red, Green, Blue), HSV (Hue, Saturation, Value), HLS (Hue, Lightness, Saturation), HSI (Hue, Saturation, Intensity) and NCC (Normalized Color Components). Any color is a combination of the primary colors (Red, Green, Blue) in specific amount/ proportion. The RGB color space was considered as a convenient one out of the many color-models for detection of the robot from the other objects. A color-component in a pixel consists of numeric values between 0-255 [14]. RGB model has 8 bits per color component i.e., 24 bits per pixel. Fig. 4 (a) shows the e-puck with a pink colored-label, while Fig. 4 (b), (c) and (d) show the individual red, green, and blue channels respectively separated from the original image. These separated channels/components were given threshold limits to extract the required colored-label while eliminating the other objects in the scene.

C. Color-thresholding

Color threshold is a method of segmenting the desired portion of an image. Upper and lower thresholds applied on the color components select portions with a particular color while eliminating the remaining portion of the image. So, thresholds were applied in such a way that only the pink-label on the robot was selected. Fig. 5 shows the intensity values of each primary color of the pink label which was on top of the robot.

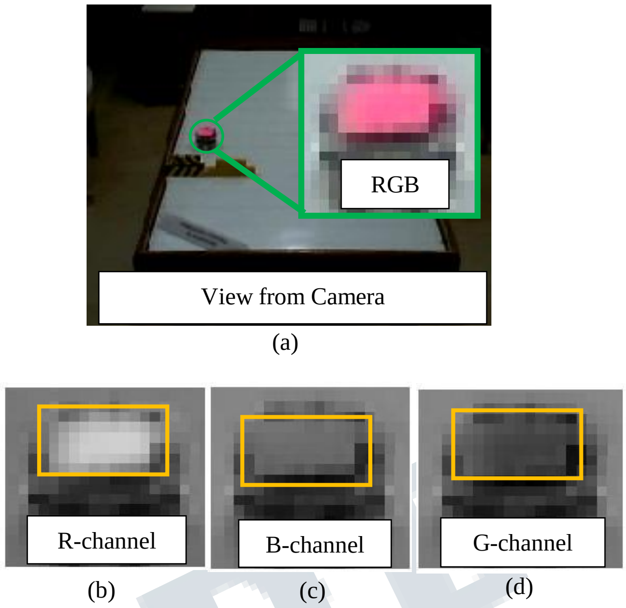
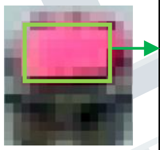


Fig. 4 Separated color channels of the colored-label



R: 204 G: 69 B: 110	R: 204 G: 69 B: 110	R: 204 G: 69 B: 110	R: 204 G: 69 B: 110	R: 204 G: 69 B: 110	R: 204 G: 69 B: 110
R: 204 G: 69 B: 110	R: 204 G: 69 B: 110	R: 204 G: 69 B: 110	R: 204 G: 69 B: 110	R: 204 G: 69 B: 110	R: 204 G: 69 B: 110
R: 204 G: 69 B: 110	R: 204 G: 69 B: 110	R: 204 G: 69 B: 110	R: 204 G: 69 B: 110	R: 204 G: 69 B: 110	R: 204 G: 69 B: 110
R: 204 G: 69 B: 110	R: 204 G: 69 B: 110	R: 204 G: 69 B: 110	R: 204 G: 69 B: 110	R: 204 G: 69 B: 110	R: 204 G: 69 B: 110
R: 205 G: 64 B: 107	R: 205 G: 64 B: 107	R: 205 G: 64 B: 107	R: 205 G: 64 B: 107	R: 205 G: 64 B: 107	R: 205 G: 64 B: 107

Fig. 5 RGB values of each pixel of the pink-label

A. Noise removal

Noise is some form of unwanted signal which can be reduced up to a great extend. These noises can damage the original data in the raw signal. Thus, filtering is required in every signal processing system. An image is a 2D signal which can have noise caused by circuit damage of the capturing device, weather, illumination, etc.

There are many filters such as median filter, adaptive filter, mean filter and so forth for removing noises in an image. Median filter removes the noise by sorting the elements in a given matrix and assigning the middle value to the central pixel of the original matrix [15]. Median filter was preferred since it simultaneously preserves the edges and removes noise of the detected region.

ISSN (Online) 2456-1290
International Journal of Engineering Research in Mechanical and Civil Engineering (IJERMCE)
Vol 2, Issue 3, March 2017

B. *Blob analysis*

Blob analysis is a technique that detects a blob/region in an image having consistent properties satisfying a given criteria. In other words, all the pixels or points in a blob can be considered to be similar to its neighbor. Some of its applications include detection of head and hands of a human body [16], identification of machine-printed characters [17], detecting flaws on silicon wafer, detecting soldering defects on electronic boards, and so forth.

In the conducted experiment, blob analysis technique was performed on the image after applying thresholds to extract the pink-colored label which was the target-blob. The algorithm places a bounding box enclosing the detected region and centre of this box was considered as the centroid of the blob. The center of the bounding box was assumed as the center of the blob. These centroids of the extracted blobs of consecutive frames are used for further processing to obtain the path tracked and velocity. Fig. 6 shows computed centroid (+) in the extracted-blob in binary image and original color-space respectively.

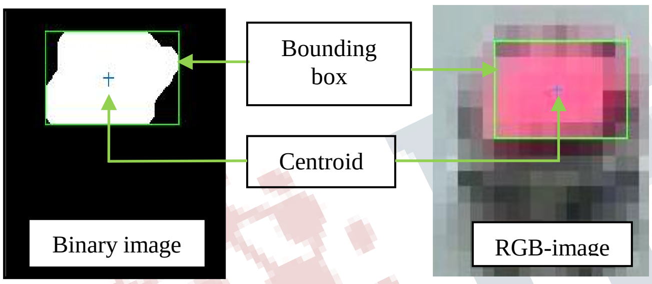


Fig.6 Isolated pink-label along with its centroid

III. CALIBRATION

The position of any detected object in an image can be specified in pixel coordinate system. This position information should be transformed to the actual coordinates (mm). In order to find the coordinate transformation equation which converts pixel coordinate (pixels) to world coordinate (mm), few sets of experiments were to be conducted. The robot was kept at various locations on the platform in such a manner that the actual x-coordinate (mm) remains same and only y-coordinate (mm) would change.

Image in the selected resolution were captured using the camera. These images were processed to find the location of the robot in pixel coordinates (pixels). Fig. 7 (a) shows the relation between measured y-coordinate (pixels) of robot in pixels with the actual y-coordinate (mm). Similarly, to calculate the relation between actual x-coordinate (mm) and measured x-coordinate (pixels), another set of experiments were conducted by keeping the robot in such a manner that the actual y-coordinate (mm) remained constant and x-coordinates (mm) only changed. Fig.7 (b) shows relation between measured x-coordinate (pixels) and actual x-coordinate (mm) while keeping the y-coordinate (mm) constant. The experiments were conducted in such a manner that the entire workspace was covered. The obtained-coordinates showed that depending upon the orientation of the two-dimensional platform, the relationship between pixel coordinates (pixels) and world coordinate system (mm) varied tremendously. It was observed that the measured y-coordinate (pixels) was dependent on actual y-coordinate (mm) and independent of actual x-coordinate (mm), while measured x-coordinate (pixels) depended on actual x-coordinate (mm) as well as actual y-coordinate (mm) for the current experimental setup.

Basic curve-fitting was done to find the relation between actual coordinates (mm) and the measured coordinates (pixels). The combination of equations (1), (2) and (3) obtained through this calibration phase gave the transformation of the measured pixel coordinates (x_m, y_m) to the actual coordinate /world coordinate (x_a, y_a) and x_e is the error in measured x coordinates (pixels).

$$x_a = x_m - (p_5 * x_e + p_6) * \left| \frac{(y_a - 10)}{140} \right| \quad (1)$$

$$y_a = p_1 * (y_m)^3 + p_2 * (y_m)^2 + p_3 * (y_m) + p_4 \quad (2)$$

$$x_e = p_7 * (x_m) + p_8 \quad (3)$$

where

$$p_1 = -0.000019663,\ p_2 = 0.017074,\ p_3 = -5.7855,$$
$$p_4 = 706.16,\ p_5 = 0.43749,\ p_6 = -96.373$$
$$p_7 = -0.51973,\ p_8 = 20.666$$

These equations are incorporated into the processing stage to obtain the measured values in world coordinate. A verification of the measurement technique was done by keeping the robot at some random location on the workspace. During this stage, the coordinates obtained by the measurement system was compared with the coordinates obtained from a standard laser calibration system (Faro laser-calibrator). The results showed that the above developed imaging system was efficient for calculating the position of the robot. The results showed that the above mentioned system was efficient for predicting the position of the robot

ISSN (Online) 2456-1290
International Journal of Engineering Research in Mechanical and Civil Engineering (IJERMCE)
Vol 2, Issue 3, March 2017

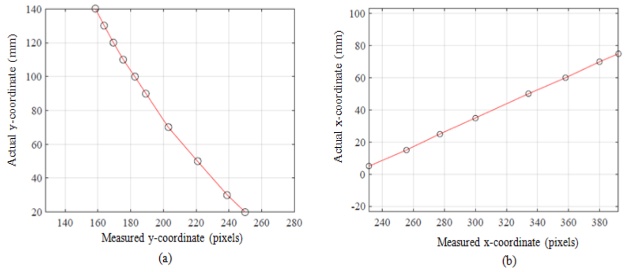


Fig.7 Calibration graphs illustrating the variation between actual coordinates (mm) and measured coordinates (pixels)

IV. RESULT AND DISCUSSION

Experiments were conducted to check the performance of the developed system for finding out the position, velocity and trajectory tracked by the robot on the platform.

1. Position measurement

The robot was placed at different locations along the diagonal connecting the origin of the platform to the corner which is diametrically opposite to the origin. Position measurement error was calculated by finding the difference in the distance from the origin as given by the proposed method and the distance which was obtained from the laser-calibration system. Table .I contains the measurement error (%) obtained. Fig. 8 shows the variation of error (%) with the actual distance (mm). It was found that the max-error of the system was ±10 mm. From the error-graph, one can conclude that there occurred very less variation in the measurement-error throughout the workspace. An attempt was also made to obtain the trajectory tracked by the robot with the help of the system. This was then compared with the actual trajectory tracked by the robot. Fig. 9 illustrates the known trajectory (thin) and the measured trajectory (thick). This comparison between the trajectories proved that the proposed algorithm was effective and worked well and it could track trajectory from the beginning till the end with less error.

2. Velocity measurement

The robot was moved over the platform in a particular known velocity, its motion was captured as sequence of frames and stored as a video file. Each frame of this video was further processed by the measurement algorithm and the bounding box enclosing the extracted-feature in each frame was obtained. Fig. 10 shows the processed frame after placing the bounding box on the detected region corresponding to the position of the robot. Incremental distance moved by the robot from one frame to other was differentiated with respect to the time interval between the frames to obtain velocity of the robot. Equation (4) calculates the distance d using the Euclidean distance formula.

Here x_i, y_i are the position coordinates of the detected object in the current frame and x_{i-1}, y_{i-1} are the coordinates in previous frame. Equation (6) calculates time interval between two adjacent frames. The frame rate used in the experiments for recording the video was 30 frames per second. Using (6) the time between each frame was calculated as 0.0333 s. For this experiment, the actual velocity of the robot was set at 50 mm/s.

$$d = \sqrt{(x_{i-1}-x_i)^2+(y_{i-1}-y_i)^2} \quad (4)$$

$$v = \frac{d}{\Delta t} \quad (5)$$

$$\Delta t = \frac{1}{frame\ rate} \quad (6)$$

TABLE I. Tabulation of distance measurement

Actual distance (mm)	Measured distance (mm)	Difference (Measured - Actual)	Error (%)
205.87	235.97	30.1	14.6
439.76	466.97	27.21	6.2
650.31	686.15	35.84	5.5
887.87	902.61	14.74	1.7
1104.4	1118.7	14.3	1.3
1330.59	1340.2	9.61	0.7
1549.4	1549.6	0.2	0.0

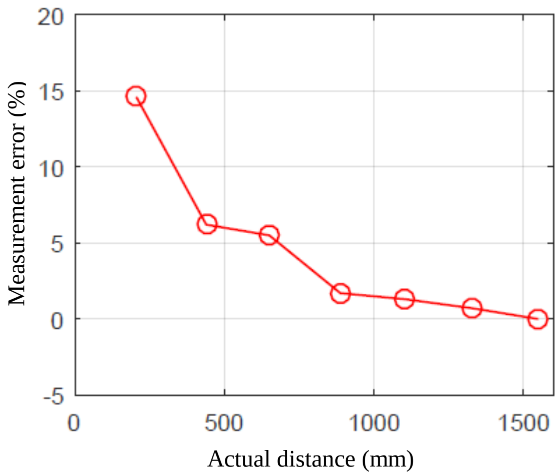


Fig.8 Distance measurement error

Fig. 11 shows the velocity measured by the current measurement system when the robot was continuously moved at 50 mm/s from one end of the platform to the other end. The plot illustrates that the measured velocity fluctuates around

IFERP
ISSN (Online) 2456-1290
International Journal of Engineering Research in Mechanical and Civil Engineering (IJERMCE)
Vol 2, Issue 3, March 2017

the actual velocity with a maximum error of ±10 mm/s. It was also observed that there was an increasing trend in the error as the robot moved away from the camera.

The fluctuation in measurement was due to the inaccuracies of the image-processing algorithm in finding the exact position of the robot. The increasing trend in error was corresponding to the increase in distance of the robot and the camera and, partially due to illumination-effects. The current work could be modified to have accurate detection and measurement by combining color-based feature extraction with subsequent frame-differencing algorithm.

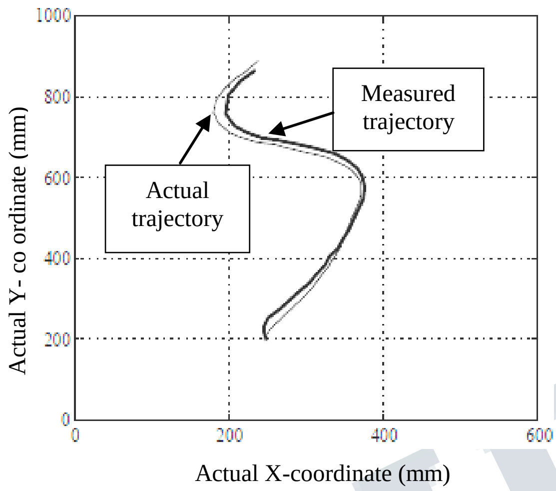


Fig. 9 Measured- trajectory versus actual-trajectory

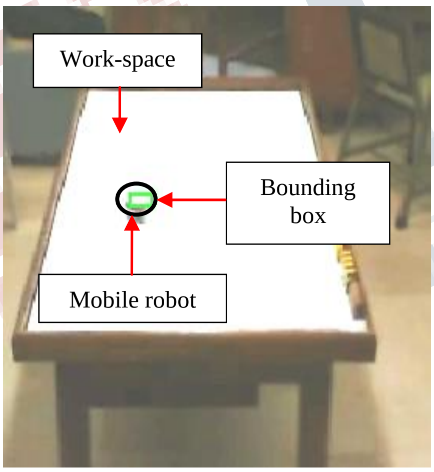


Fig. 10 Processed image of the robot on the platform after detection

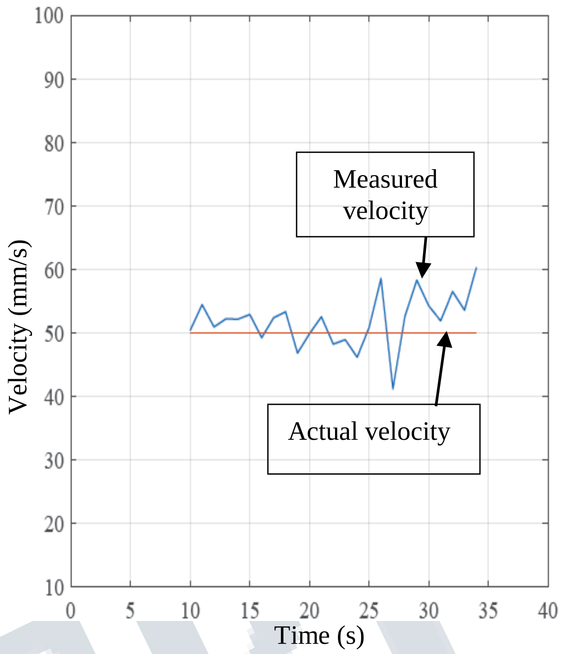


Fig. 11 Measured velocity versus actual velocity

V. CONCLUSION

The importance of image processing and analysis has brought out a massive outcome in various spheres of researches such as object detection, tracking, robot navigation, biometric applications, defense and security, hand-writing recognition, currency inspection, medical domain, etc. A novel implementation of this topic for mobile robot detection and tracking was applied in the paper and some of the highlights are mentioned below: Measurement of position and velocity of a mobile robot using monocular vision was presented. The following are the scanning of the paper:

- The algorithm that involved feature extraction and blob analysis could estimate the static and dynamic quantities of the robot with necessary accuracy. This approach was computationally simple and takes less time in extraction of the feature (color).
- The validation was done using laser-coordinate measurement instrument, Faro. Thus, an application of this technical instrument was also demonstrated in this work.
- There were few challenges that came up such as clutter, illumination-effects and change in the appearance of the detected blob as the robot moved farther from the camera. Also, if the camera position changes, recalibration is necessary.
- Future version of this implementation includes trajectory and velocity tracking of multiple mobile robots.

ISSN (Online) 2456-1290
International Journal of Engineering Research in Mechanical and Civil Engineering (IJERMCE)
Vol 2, Issue 3, March 2017

REFERENCES

[1] A. Cavallaro and T. Ebrahimi, "Interaction between high-level and low-level image analysis for semantic video object extraction", EURASIP Journal on Applied Signal Processing, vol. 6, pp. 786-797, June 2004.
[2] C. Stauffer and W. E. L. Grimson, "Learning patterns of activity using real-time tracking, IEEE Trans. on Pattern Analysis and Machine Intelligence, vol. 22, pp. 747-757, 2000.
[3] D. Comaniciu, V. Ramesh and P. Meer, "Kernel-based object tracking", IEEE Trans. on Pattern Analysis and Machine Intelligence, vol. 25, pp. 564-577, May 2003.
[4] V. Parameswaran, V. Ramesh and I. Zoghlami, "Tunable kernels for tracking," IEEE Computer Society Conference on Computer Vision and Pattern Recognition, vol. 2, pp. 2179-2186, 2006.
[5] Y. Raja, S. J. Mckenna and S. Gong, "Segmentation and tracking using color mixture models", Proceedings of Third Asian Conference on Computer Vision, vol. 1, pp.607-614, Jan 1998.
[6] N. Paragios and R. Deriche, "Unifying boundary and region-based information for geodesic active tracking", IEEE Computer Vision and Pattern Recognition, 1999.
[7] Al. Bovik, The essential guide to video processing. AP, 2009.
[8] E. Maggio and A. Cavallaro, Video tracking-theory and practice, Wiley, 2001.
[9] T. B. Moeslund and E. Granum, "A survey of computer-vision based human motion capture", Computer Vision and Image Understanding, pp. 231-268, 2001
[10] F. Mondada, M. Bonani, S. Magnenat, X. Raemy et al., "The e-puck, a robot designed for education in engineering", Proceedings of the 9th Conference on Autonomous Robot Systems and Competitions, vol. 1, pp. 59-65, 2009.
[11] G. Kumar and P. K. Bhatia, "A detailed review of feature extraction in image processing QW systems", Proceedings of the fourth International Conference on Advanced Computing and Communication Technologies, pp. 5-12, 2014, .
[12] P. Fieguth and D. Terzopoulos, "Color based tracking of heads and other mobile objects at video frame rate", IEEE Conference on Computer Vision and Pattern Recognition, pp. 21-27, 1997.
[13] F. Liu, Q. Liu and H. Lu, "Robust color-based tracking", IEEE Proceedings of the Third International Conference on Image and Graphics, pp. 1-4, 2004.
[14] S. S. Rumma, "Study of name plate detection using blob analysis method", International Journal on Recent and Innovation Trends in Computing and Communication, vol. 4, pp. 320-322, Feb. 2016.
[15] B. Tharanidevi, R. Vadivu and K. B. Sethupathy, "Moving object tracking distance and velocity determination based on background subtraction algorithm", IOSR Journal of Electronics and Communication, vol. 8, pp. 61-66, Oct. 2013.
[16] S. Lu, G. Tsechpenakis, D. N. Metaxas, M. L. Jensen and J. Kruse, "Blob analysis of the head and hands: a method for deception detection", Hawaii International Conference on System Science, 2005.
[17] Patil, "Blob detection technique using image processing for identification of machine printed characters", International Journal of Innovations in Engineering Research and Technology, vol. 2, pp. 1-8, Oct. 2015

