ISSN (Print) : 0974-6846
Indian Journal of Science and Technology, *Vol 8(32), DOI: 10.17485/ijst/2015/v8i32/92131, November 2015* ISSN (Online) : 0974-5645

Real Time Object Detection & Tracking over a Mobile Platform

M. A. Mekhtiche, M. A. Bencherif*, M. Algabri, M. Alsulaiman, R. Hedjar, M. Faisal and K. AlMutib

Center for Smart Robotics Research, CCIS, King Saud University, Riyadh, Saudi Arabia; mbencherif1@yahoo.com, mmekhtiche@ksu.edu.sa, malgabri@ksu.edu.sa, msuliman@ksu.edu.sa, he-djar@ksu.edu.sa, mfaisal@ksu.edu.sa, muteb@ksu.edu.sa.

Abstract

Background/Objectives: Nowadays, object detection and tracking is an important issue in robotics and computer vision systems, especially in video surveillance, robot navigation and autonomous vehicle navigation. **Methods**: In this paper, we propose a fast method for object tracking and recognition within the context of a mobile robot acquiring real time images from a top mounted IP camera. The aim of the proposed method is to let the robot identify multi-targets within a scene, then move toward the desired object. The method is based on a new vectored contour to identify objects from HSV images. **Findings**: Experimental results have shown that our method best fitted the mobile platform and gave excellent competitive results in real time tracking. Our proposed method has shown better adaptation com-pared to SURF and other state of the art tracking methods, especially in terms of time and simplicity, specifically when the camera is not in front of the target object, i.e. at different inclination angles and distances. **Application**: The object detection and tracking proposed in this work can be implemented on many fields such as video surveillance robotic navigation or in industry in classification of objects depending on their forms or colors.

Keywords: Mobile Robot, Object Recognition and Tracking

1. Introduction

In autonomous mobile navigation, tracking objects is crucial, mainly in the framework of detecting and moving toward the desired objects. Different algorithms have been proposed in the literature. The most popular method is the Scale Invariant Feature Transform (SIFT)[1-3] or its improved version Speeded-Up Robust Features (SURF)[4]. Unfortunately, both methods suffer from the zoom issue, especially when the object is at a far distance and require enormous computations. Moreover, their performance degrades when tracked objects have very few details. The background subtraction as improved by[1-5] is mainly used for fixed cameras. The frame difference[6-7][6, 7] becomes useless, if the tracked object stops moving. The mean shift algorithm[8-10] is mainly effective for a single object tracking, while color tracking[11] is more related to the presence of that colored object within the image, the shape has to be identified by a complementary method.

In section 2, we describe the proposed method, in section 3, we present the results of the identification, in section 4, a time complexity study is developed, section 5 discusses the results, and we conclude at section 6.

2. Methodology

The goal of this work is to let the robot find different required objects within a real time video stream, then position itself autonomously and move toward the desired target. For a real time implementation, a fast tracking algorithm is required, we propose, in this context, the use of a color histogram and a normalized vectored contour. The histogram is used to identify the regions of the image having the same color of the object.

**Author for correspondence*

The suggested normalized vectored contour is then fed as input to the multi-object identification stage, as illustrated in Figure 1.

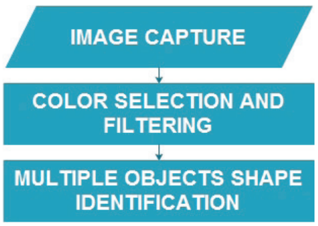


Figure 1. Flowchart of the tracking algorithm.

Our proposed method comprises two stages. The first stage deals with template generation of all the required objects to be identified. It consists of acquiring images of theobject templates, at different angles and distances, then converting each contour to a linear form, as it will be detailed in section 2.3. The second stage is the real time tracking, as it requires in addition to the color selection and contour generation, the dynamic template warping for multi-object identification.

2.1 Image Capture

Real time images are captured via a D-LINK 5222L IP camera with a resolution of 800 x 448 pixels, a frame rate of 25 fps, a focal length of 3.6 mm, within angles of view (H=70,V=53,D=92). The VLC media player ActiveX within Matlab generates the images from the acquired video at a down rate of 2 to 3 frames per second, experimental results have shown these frames are sufficient to track an object in real time. The camera is mounted on the DR-robot Scout-II[12], as shown in Figure 2.

Figure 2. DR-Robot platform

2.2 Color Selection and Filtering

The captured RGB images are transformed into HSV format. From the HSV images of the template of the desired target, we generate a histogram for the color of the desired object. For example, Figure 3 shows the histogram for the orange color, where the HSV parameters have a Hue (H=0.12), a Saturation (S=1), and a Value (V=1).

From the real time HSV images, as in Figures 4(b), 5(b), we select the pixels with the HSV parameters of the color of the object (orange in this case). To allow for a slight illumination, we also select pixels within a (±0.1) HSV parameters of the desired color. Figures 4(c), 5(c) show examples of the color selection.

After the color selection (CS) step, noise is removed by a median filter, some smoothed image examples are shown in Figure 4(d) and 5(d).

2.3 Multiple Object Recognition

The CS step finds, within the image, all the objects having the same color. The proposed solution is to recognize the shape of each object and identify it using a predefined

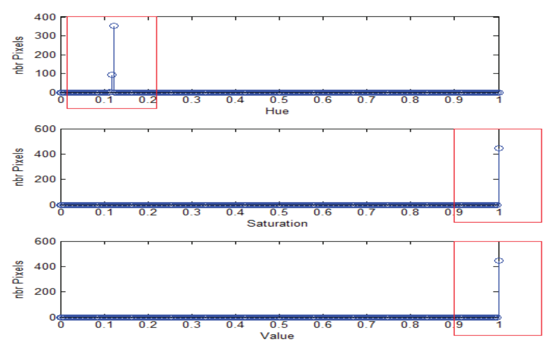


Figure 3. Histogram of the target objects.

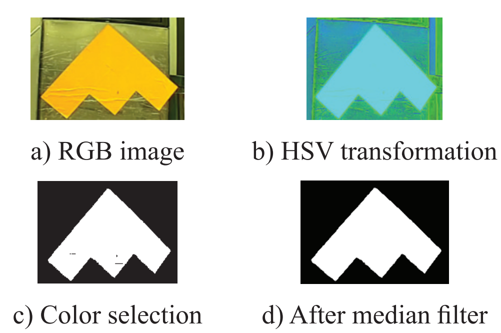
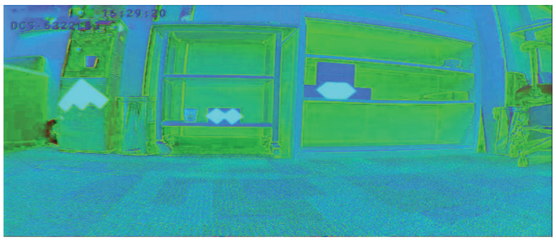


Figure 4. Template color selection and filtering.

a) Real time RGB frame image



b) HSV transformation

c) Color selection

d) Median filtered image

Figure 5. Scene color selection and filtering.

contour template. The use of the linear contour as a new feature is described below:

Given any found object with pixel coordinates (X_i, Y_j) where i,j belong to the contour of the object, as shown in Figure 6.

a. A new one dimensional vector is formed by concatenating all X then Y coordinates of each contour. A plot of the newly vector of the leftmost object of Figure 6 is presented in Figure 7.
b. Each normalized contour is then compared to all the contour templates through dynamic template warping,

Figure 6. Edge extraction.

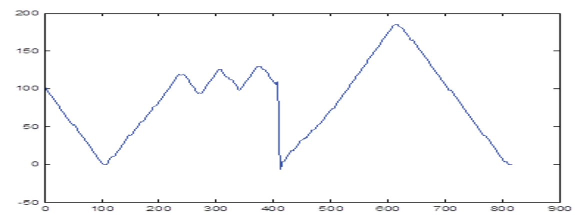


Figure 7 Normalized contour of the leftmost detected object of Figure 6.

known also in time dependent variables as dynamic time warping[13], as illustrated in Figure 8. The template comparison uses the sum of minimum distances to compare the two sequences (R and T, where R is the contour of the Object and T the contour template), as defined by equation (1).

$$\gamma(i,j) = d(R_i, T_j) + min \begin{cases} \gamma(i-1, j-1) \\ \gamma(i-1, j) \\ \gamma(i, j-1) \end{cases} \qquad (1)$$

Where $\gamma(i,j)$ is the cumulative distance at (i,j) and $d(R_i, T_j)$ the local distance, between their respective (i,j) samples.

c. The least distance to any template gives the identity of the objects within the scene.

The Mobile platform redirects itself toward the selected object and moves at a fixed speed, then repeats the procedure each half a meter approximately, until the object is at distance of 40 cm from the camera.

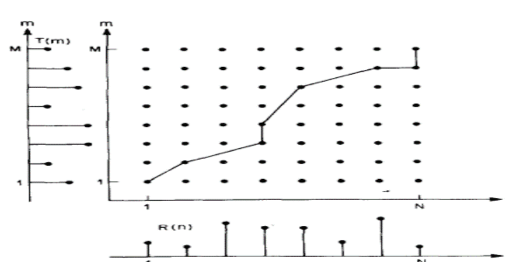


Figure 8. DTW scoring matrix(18).

3. Experimental Results

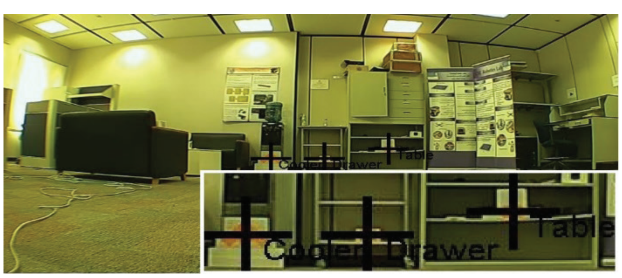
The mobile robot moves at anaverage speed of 1m/s, the top mounted D-Link IP camera acquires images each 40 ms, (25fps), the proposed algorithm requires approximately 540 ms to achieve the detection of the predefined target. Different experiments were conducted while the robot was moving, no movement compensation was included in our process, as the floor was approximately flat, and the video was transmitted fluently. Experiments were made with different angles of inclination, ranging from zero (robot perpendicular to the object) to an inclination of more than 65 degrees. The results are presented in the following sections.

3.1 Zero Degree Inclination Experiments

Experiments were performed, while the robot was between two to six meters far away from the central object. Figures 9(a) to 9(c) illustrate the success of the method in detecting the three target objects.

a) 2m distance@ 0degree

b) 4m distance

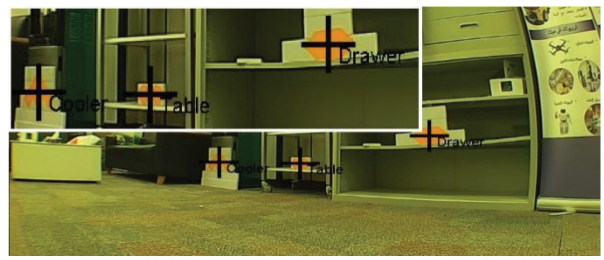


c) 6m distance

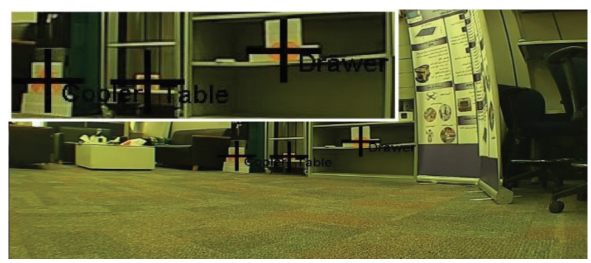
Figure 9. Zero degree inclination results. (The white square shows the zoomed region of the detected objects)

3.2 Varied Degrees Inclination Experiments

Similar distance experiments were conducted, but with different angles of inclination, (45, 70 and -70 degrees), as illustrated in Figure 10 (a-d).



a) 3m distance @ 45degrees



b) 5m distance @ 45degrees

c) 3m distance @ 70degrees

d) 2m distance @ -70degrees

Figure 10. Tracking with varied angles.

Table 1. Estimated time in (sec) and detection success

Algorithm \ @dist.&angle	@2m 0deg	@2m 45deg	@4m 0deg	@4m 45deg
Proposed Method	**0.28**	**0.36**	**0.41**	**0.31**
SURF	1.91	object not identified	object not identified	object not identified
Correlation[16]	4.621	object not identified	object not identified	object not identified
FastMatching[17]	0.192	object not identified	0.217	object not identified

At inclination angles greater than 70 degrees, our method is no more able to accurately recognize the detected objects, as show in Figure 10(d), where two objects are detected as coincident.

4. Comparison with Existing Methods

- In order to validate our tracking method, we made some comparisons with some state of the art methods, within the following framework:
- Detecting a single object over a mobile platform, from a video stream, where the Top mounted camera is positioned at different angles and distances from the target. The tracking results are illustrated in Table 1.

5. Complexity Analysis

Computations, of our proposed algorithm, require 540ms, to find 3 objects and track the predefined target, using an Intel 7-2670QM, 2.2GHz, 2GB Ram, 64Bits windows7 OS, with Matlab 2014a, 32bits. Detailed timings of the most relevant functions are listed in Table 2.

Table 2. Time complexity of the main functions

Functions	**Time (ms)**
Template generation (3 objects)	208
Getframes from VLC player ActiveX (Matlab)	45
RGB to HSV transformation (frame image)	1
Dynamic Template Warping	187

6. Discussion

Real time detection and tracking is a hard problem, as complexity arises from different aspects, we will describe the two most important points, as follows:

6.1 Distance to Tracked Objects

In our experiments, the height of the templates varied between 10 cm to 18 cm, and the width between 20 cm to 30 cm, false detections started from 6 meters and above.

In order to cope with the distance false detection, templates can be increased in size, but at the expense of large contours, leading to heavy computations.

We also tried to use the digital zoom of the camera, but this did not perform well, as it zooms digitally till 10x, pixels are size amplified and no information can be added to the algorithm. An additional strategy is to segment the frame images into 2 or 4 regions, and work by region, but the target objects may split between the image regions, and might not be detected.

6.2 Illumination

Light can be controlled in a closed environment. Regrettably, it is not the case for all the tracking environments. In our case, the HSV range can be increased, but at the expense of the time of computations.

At short distances, with zero angle, our proposed method had significant shorter time than SURF, but less time than fast matching. For larger distances, at different angles, SURF and other methods failed to identify the object.

We remark that the other algorithms did not identify the tracked object, as they suffer either from the far distance of the tracked object, or the angle of the object relative to the center of the camera.

Let us remark that fast matching[14] gave better fast results at zero inclination, but mislead and failed when the target is at an angle greater than zero. (Robot not directly facing the object).

6.3 Why Dynamic Template Warped Worked?

Objects seen at different far distances and angles tend to lose detail[15], and small deformations tend to be approximately linear, but when a template is dynamically wrapped, and compared to the detected contours, the deformation applies to all objects of the scene and tends to influence all the comparisons with predefined templates. As the DTW distance is cumulative, for all the objects, deformations will impact all the comparisons in a nearly similar manner.

Other methods such as SURF or SIFT, are computationally based on details; when a deformation occurs, either from distance or angle changes, both methods, lose many points of interest, and could no more find the original details, leading to a failure detection.

7. Conclusion

In this paper, we have proposed a simple and fast tracking method, usingcolor selection and shape recognition, via a new approach of contour linearization; the object matching is done via a dynamic warping comparison. Our method required shorter time (crucial for real time), compared to different other methods, through the whole process of capturing and detecting the specific target.

The proposed method is more efficient and stable, as the detection and tracking computations required approximately 540 ms, over a platform moving at an average speed of 1m/s.

The mobile platform, with its top mounted IP camera, was able to detect small size specific object(s), at distances less than 6 meters, and within a maximal inclination angle of 65 degrees.

8. Acknowledgement

This work was supported by the Research Center, College of Computer and Information Sciences, King Saud University.

9. References

1. Rahman MS, Saha A, Khanum S, editors. Multi-object tracking in video sequences based on background subtraction and sift feature matching. IEEE, 2009 ICCIT'09 Fourth International Conference on Computer Sciences and Convergence Information Technology. 2009.
2. Chen W, Zhao Y, Xie W, Sang N, editors. An improved sift algorithm for image feature-matching. IEEE, 2011 International Conference on Multimedia Technology (ICMT). 2011.
3. Li K, Zhou S, editors. A fast sift feature matching algorithm for image registration. IEEE, 2011 International Conference on Multimedia and Signal Processing (CMSP). 2011.
4. Juan S, Qingsong X, Jinghua Z, editors. A scene matching algorithm based on SURF feature. IEEE, 2010 International Conference on Image Analysis and Signal Processing (IASP). 2010.
5. Lee Y-S, Lee H, editors. Multiple object tracking for fall detection in real-time surveillance system. IEEE, 2009 ICACT 2009 11th International Conference on Advanced Communication Technology. 2009.
6. Zeng L, Xu L, editors. Moving multi-object tracking algorithm based on wavelet clustering and frame difference. IEEE, 2009 SMC 2009 IEEE International Conference on Systems, Man and Cybernetics. 2009.
7. Chu H, Ye S, Guo Q, Liu X, editors. Object tracking algorithm based on camshift algorithm combinating with difference in frame. IEEE, 2007 IEEE International Conference on Automation and Logistics. 2007.
8. Jackin IM, Manigandan M, editors. Wireless Vision Based Object Tracking Using Continuously Adaptive Mean Shift Tracking Algorithm. IEEE, 2009 ARTCom'09 International Conference on Advances in Recent Technologies in Communication and Computing. 2009.
9. Tao L, Xiao-Ping C, editors. Improved mean shift algorithm for moving object tracking. 2010 2nd International Conference on Computer Engineering and Technology (ICCET). 2010 April 16-18.
10. Guo-Liang W, De-Qun L, Yan-Chun W, Zhao-Hua H, editors. Algorithm for tracking of fast motion objects with adaptive mean shift. IEEE, 2007 SNPD 2007 Eighth ACIS International Conference on Software Engineering, Artificial Intelligence, Networking, and Parallel/Distributed Computing. 2007.
11. Chung J, Shim K, editors. Color object tracking system for interactive entertainment applications. IEEE, 2006 ICASSP 2006 Proceedings 2006 IEEE International Conference on Acoustics, Speech and Signal Processing. 2006.
12. Dr Robot Inc: Scout 2. 2015.

13. Myers C, Rabiner LR, Rosenberg AE. Performance tradeoffs in dynamic time warping algorithms for isolated word recognition. IEEE Transactions on Acoustics, Speech and Signal Processing. 1980; 28(6):623-35.
14. Briechle K, Hanebeck UD, editors. International Society for Optics and Photonics: Template matching using fast normalized cross correlation. Aerospace/Defense Sensing, Simulation, and Controls; 2001.
15. O'Shea RP, Blackburn SG, Ono H. Contrast as a depth cue. Vision Research. 1994; 34(12):1595-604.
16. Bay H, Tuytelaars T, Van Gool L. Surf: Speeded up robust features. Springer: Computer vision-ECCV 2006. 2006; p. 404-17.
17. Korman S, Reichman D, Tsur G, Avidan S, editors. FAsT-Match: Fast Affine Template Matching. IEEE, 2013 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). 2013.