

**Military Technical College
Kobry El-Kobbah,
Cairo, Egypt**



**7th International Conference
on Electrical Engineering
ICEENG 2010**

Monocular Object Tracking based on Corner detection

By

Ahmed Ali *

Gouda I. salama *

M. S. Hamed *

Abstract:

This paper addresses corner based method for allocating the position of an object in video frames. Corners distributed inside a predefined search area used to represent the object shape. It is used to estimate the most probable target position in the next frame. Tracking is done through matching process between corner's information in current frame (candidate corner) and that in the previous frame (reference corner). Finally, an updating step for the reference corners distribution model and the predefined search area position is carried out for the next frame. The proposed technique has shown a very good performance when applied to real captured videos frames. The calculated errors in the locations of the moving object were never more than 8 pixels.

Keywords:

Object detection, Corner-based tracking

* Egyptian Armed Forces

1. Introduction:

Numerous approaches for object tracking have been proposed. These approaches primarily differ from each other based on the way they answer the following questions: First, which object representation is suitable for tracking. Second, which image features should be used. Finally, how the motion, appearance, and shape of the object should be modeled [1]. In literature, Objects can be represented by points, primitive geometric, object contour, articulated shape, and skeletal models [2]. Corners in images represent critical information in describing an object, while corner detection has various application areas including motion tracking. It depends on determination of object contour. The computational time for matching process in corner tracking is proportional to the square of corner number, while for contour tracking is proportional to the square of the object perimeter. So, for applications that require less computational time, corner tracking is preferable [3, 4]. Corner tracking has been used for two motivations: First, detecting object position; Second, correcting the prediction of object position [5,9]. M. Asadi et al [6] used a model similar to the proposed technique to detect an object inside a predefined area. They used SUSAN corner extractor for corner extraction, but SUSAN corner extractor is time consuming and a single scale detector [7].

The proposed tracking technique uses an enhanced multiscale algorithm based on curvature scale space (CSS), which was proposed by Xiao Chen He [7]. Again, this model has been used by L.Marcenaro et al as a part of tracking algorithm to recognize objects in the scene when an overlap exists between their bounding boxes [8]. Gabriel et al. [9] exploit corners information to correct the prediction of the object position done by Kalman filtering.

The following section of this paper demonstrates the functional block diagram of the proposed tracking technique in details. Section 3 discusses performance evaluation of CCS and SUSAN corner extractor algorithms. Section 4 including experimental results and error analysis. Concluding remarks are presented in the last section.

2. Proposed Tracking technique:

Figure (1) depicts the functional block diagram of the proposed tracking technique. It consists of four steps: reference corners step, candidate corners step, matching corners step and updating step. The technique starts with reference corner step, in which the user creates a bounding box around the desired object in the first frame, then the CCS corner extractor is applied to the bounding box to extract a reference corner. The candidate corner step which is responsible for extracting candidate corners inside the same position of the bounding box in the next frame begins. The matching corners step finds the most probable target position inside the bounding box by correlating the two sets of corners, Finally, an updating step updates the reference corners distribution

model and the search area positioning in the next frame.

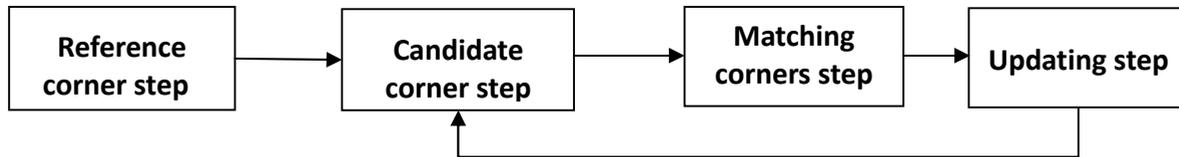


Figure (1): Functional block diagram of the proposed tracking technique

The complete steps of the above technique are described by the following procedure assuming that the first frame contains a target:

- S1: Read the first frame F.
- S2: Select a bounding box around target.
- S3: Reference corners model → apply CSS corner detector to bounding box.
- S4: Read next frame F2.
- S5: Candidate corners → apply CSS corner detector to F2 inside Dimension of bounding box.
- S6: Matching step between Model corners and Candidate corners
- S7: Determine the new Object location
- S8: The target model and bounding box position will be updated
- S9: Repeat step S3 until end of frames.

2.1 Proposed Tracking technique:

The aim of reference corner step is to form the Corners model that describes an object. Starting from first frame, the user creates abounding box around the object to be tracked in the sequence, then CCS corners extraction algorithm is applied inside the bounding box to extract reference corners set $\{RC_i\} \ 1 \leq i \leq n$ that represents object. Each corner has a set of information described by the following definition [6]:

$$RC_i = [x_i, y_i, I_i, dx_i, dy_i] \quad (1)$$

where x_i and y_i are the coordinates of the corner i . I_i is its intensity value. dx_i and dy_i is the relative coordinate of i^{th} corner with respect to the center of bounding box .

2.2 Candidate corner step:

This step is mainly dedicated to form the Candidate corner in the next frame, assuming that the object position in two successive frames are overlapped, such that the new object position is still inside the previous bounding box. The Candidate corner step is similar to the Reference corner step. The Candidate set Corners is formed by $\{CC_j\}$

1 j m to describe the object in the next frame according to the following definition[6]:

$$CC_j = [x_j, y_j, I_j, dx_j, dy_j] \quad (2)$$

where x_j and y_j are coordinates of corner j . I_j is its intensity value. dx_j and dy_j is relative coordinate of j^{th} corner with respect to the center of bounding box .

2.3 Matching corners step:

This step is to estimate the most probable target position inside the bounding box by correlating the two sets of corners. Each corner in the reference model corner is compared with candidate corners to find the best match .Firstly; two corners having intensity value within the same range are checked. Secondly, if this condition is satisfied, the acting algorithm then finds the relative distance between the two corners. Thirdly, a bounding box position is shifted by this distance. Finally, a block matching is applied to the new position of bounding box in the current frame and the old bounding box position in the previous frame, these four steps are applied to all corners to obtaining the shift distance that gives best match.

2.4 Updating step:

After matching step, the reference corners and the bounding box position will be updated for the next frame as follows: First, the reference corners distribution for the next frame $\{RC_i\}$ are updated with the candidate corners distribution of current frame $\{CC_j\}$, Second, after finding the shift distance that gives the best correlation between the two sets of corners(reference and candidate corners) in the matching step, the bounding box position in the next frame is shifted by the same distance .Finally, Candidate corner step will be repeated for the next frame inside the new position of the bounding box.

3. Performance evaluation of CCS and SUSAN corner extractor:

This section demonstrates the effect of corners extractor type that is used in object tracking technique. M. Asadi, C.S. Regazzoni [6] used SUSAN corner extractor in their model, but in the proposed technique CCS corners extractor is used. To evaluate the performance of the two models: First two corners extractor algorithms to some published test shape are applied [7].Finally, the tracking technique is applied on a real video sequence using CCS and SUSAN corners extractor for performance evaluation.

3.1 Test Results of CCS and SUSAN:

This subsection demonstrates the results obtained by applying the CCS and SUSAN corners extractor to some published test shape [7]. The corners can be classified into true corners C_t , false corners C_f , round corners C_r and missed corners C_m . Figure (2) depicts the Extracted corners in testing shapes using CCS and SUSAN. In the proposed technique, the time for the matching step would be effected by number of extracted corner. Table.1 shows that CCS is less affected by false corner and round corner than SUSAN. This result leads to decreasing the overall execution time in case of using CCS instead of SUSAN as a corner extractor.

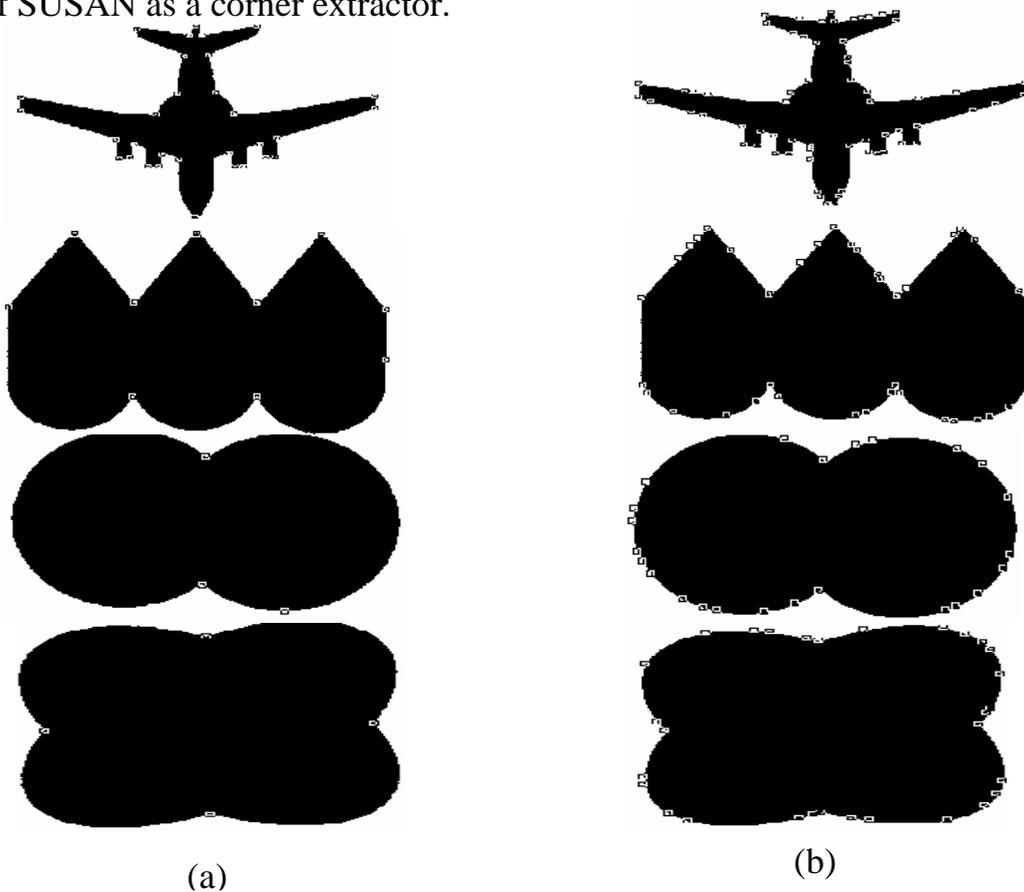


Figure (2): Selected image shapes after applying (a) CCS and (b) SUSAN

Table (1): True C_t , False C_f , Round C_r and Missed C_m corners detected by CCS and SUSAN for Selected image shapes Figure (2).

| Detector | Shape (1) | | | | Shape (2) | | | | Shape (3) | | | | Shape (4) | | | |
|----------|-----------|-------|-------|-------|-----------|-------|-------|-------|-----------|-------|-------|-------|-----------|-------|-------|-------|
| | C_t | C_f | C_r | C_m |
| CCS | 31 | 0 | 0 | 6 | 9 | 1 | 0 | 2 | 2 | 0 | 1 | 0 | 4 | 0 | 0 | 0 |
| SUSAN | 36 | 26 | 2 | 2 | 9 | 13 | 12 | 2 | 2 | 0 | 22 | 0 | 4 | 0 | 26 | 0 |

4. Discussion and Results:

The results obtained by applying the proposed tracking technique using CCS extractor to real video sequences are compared with results obtained by applying the technique using SUSAN corner extractor. The sequences include video frames of a flight plane, a moving car and a toy [wireless car].

In the first experiment, as illustrated in Figure (3), the system tracked a flight plane over a monocular sequence of 112 frames. Figure (4.a) shows the trajectory of the plane in the images, together with the detected location generated using CCS corners extractor. Figure (4.b) shows the plane trajectory again, this time using SUSAN corners extractor. Figure (5) contains plots of the errors exhibited by the system for the sequences. Figure (5.a) represents the image distance (measured in pixels) from each calculated location using CCS corners extractor to the true target location. Figure (5.b) represents the image distance separating each calculated location using SUSAN corners extractor to the true target location. It can be seen that the error in the calculated location was never more than approximately 7 pixels distant from the actual target location in case of CCS corners extractor, and approximately 10 pixels using SUSAN corners extractor. The tracking system takes long time for execution with SUSAN than CCS corner extractor.

In the second experiment, as illustrated in Figure (6), the system tracked a real moving car over a monocular sequence of 20 frames. Figure (7.a) shows the trajectory of the car in the images, together with the detected location generated using CCS corners extractor. Figure (7.b) shows the car trajectory again, this time using SUSAN corners extractor.

Figure (8) contains plots of the errors exhibited by the system for the sequences. Figure (8.a) represents the image distance (measured in pixels) from each calculated location using CCS corners extractor to the true target location. Figure (8.b) represents the image distance separating each calculated location using SUSAN corners extractor to the true target location. It can be seen that the error in the calculated location was never more than approximately 4 pixels distant from the actual target location in case of CCS corners extractor, and approximately 4 pixels using SUSAN corners extractor. The tracking system takes long time for execution with SUSAN than CCS corner extractor.

In another experiment, as illustrated in Figure (9), the system tracked a wireless car over a monocular sequence of 371 frames. Figure (10.a) shows the trajectory of the wireless car in the images, together with the detected location generated using CCS corners extractor. Figure (10.b) shows the car trajectory again, this time using SUSAN corners extractor. Figure (11) contains plots of the errors exhibited by the system for this image sequences. Figure (11.a) represents the image distance (measured in pixels) from each calculated location using CCS corners extractor to the true target location, Figure (11.b) represents the image distance separating each calculated location using SUSAN corners

extractor to the true target location. It can be seen that the error in the calculated location was never more than approximately 7 pixels distant from the actual target location in case of CCS corners extractor, and approximately 8 pixels using SUSAN corners extractor, the tracking system take long time for execution with SUSAN than CCS corner extractor. The system was successfully tested using several other image sequences. Due to space limitations, only two examples are shown here.

The proposed technique was implemented in matlab and it was executed in 1.7GHZ, core 2 duo laptop, with 4Gbyte of memory, and the execution time of the proposed technique was evaluated. as depicted in Table.2 .below illustrates the execution time for CCS and SUSAN corner extractors along with the calculated Max error it is clear that the execution time for CCS is less than SUSAN, and is due to the fact that number of false and round corners is greater in SUSAN than in CCS.

Table (2): shows that Execution time for tracking technique using CCS and SUSAN

| Detector | Experiment #1 | | | Experiment #2 | | | Experiment #3 | | |
|----------|---------------|----------------|-----------|---------------|----------------|-----------|---------------|----------------|-----------|
| | Frame NO. | Execution time | Max error | Frame NO. | Execution time | Max error | Frame NO. | Execution time | Max error |
| CCS | 112 | 234.205 | 6 | 20 | 8.358 | 5 | 371 | 80.0140 | 6 |
| SUSAN | 112 | 240.38 | 10 | 20 | 8.8352 | 5 | 371 | 106.390 | 7 |

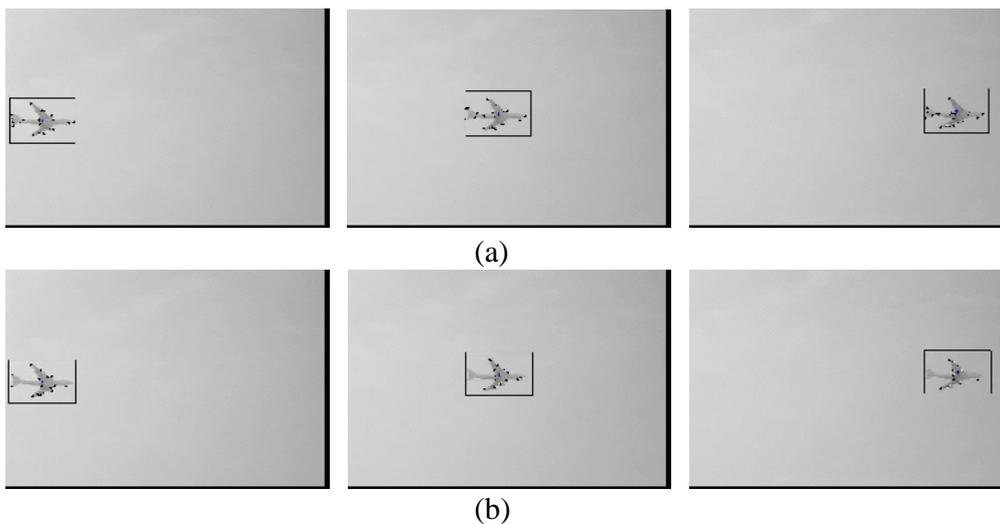


Figure (3): selected images from "real plan" image sequences after applying the tracking algorithm. (a) frame 2 ,frame 50 and frame 100 in sequence using CCS corners extractor.(b) frame 2 ,frame 50 and frame 100 in sequence using SUSAN corners extractor. Each "+" sign points to the tracked target detected by the corresponding algorithm.

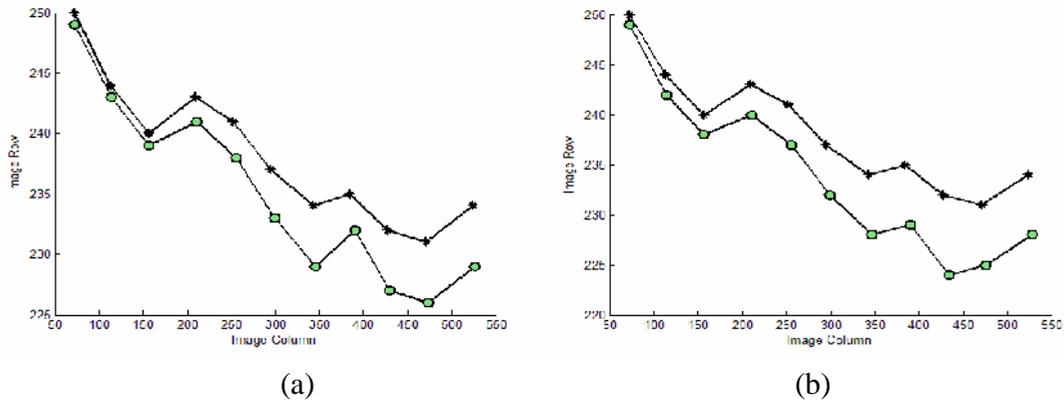


Figure (4): Actual trajectory (o) and points Detected (*). (a) using CCS corners extractor, (b) using SUSAN corners extractor.

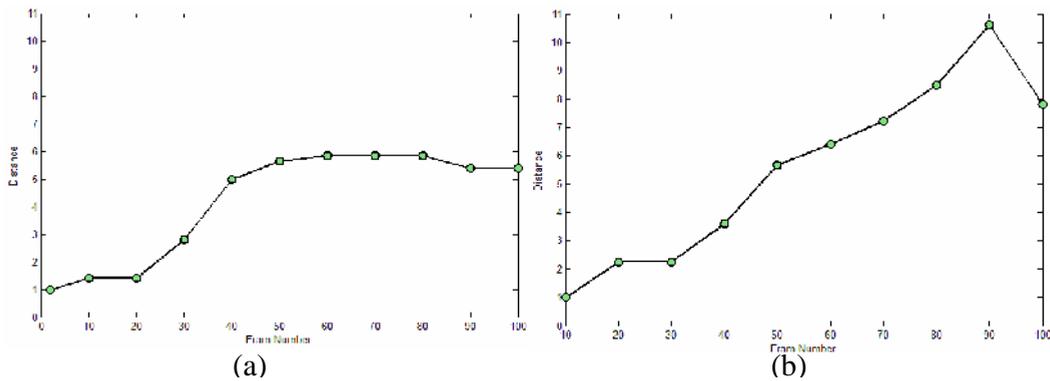


Figure (5): Euclidean distance from actual target location to calculated location (a) using SUSAN corners extractor, (b) using CCS corners extractor.

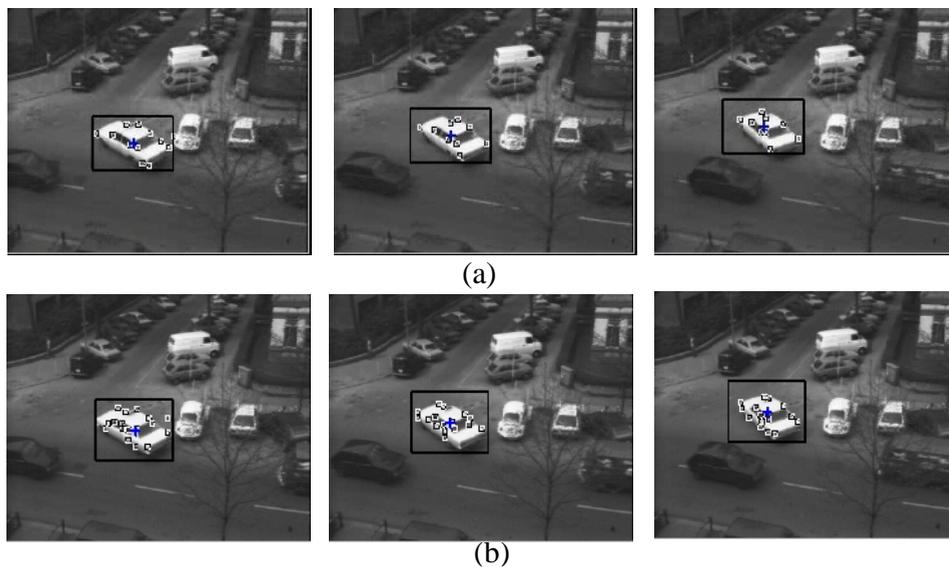


Figure (6): selected images from "real car" image sequences after applying the tracking algorithm. (a) frame 2 ,frame 8 and frame 18 in sequences using CCS corners extractor.(b) frame 2 ,frame 8 and frame 18 in sequences using SUSAN corners extractor. Each "+" sign points to the tracked target detected by the corresponding algorithm.

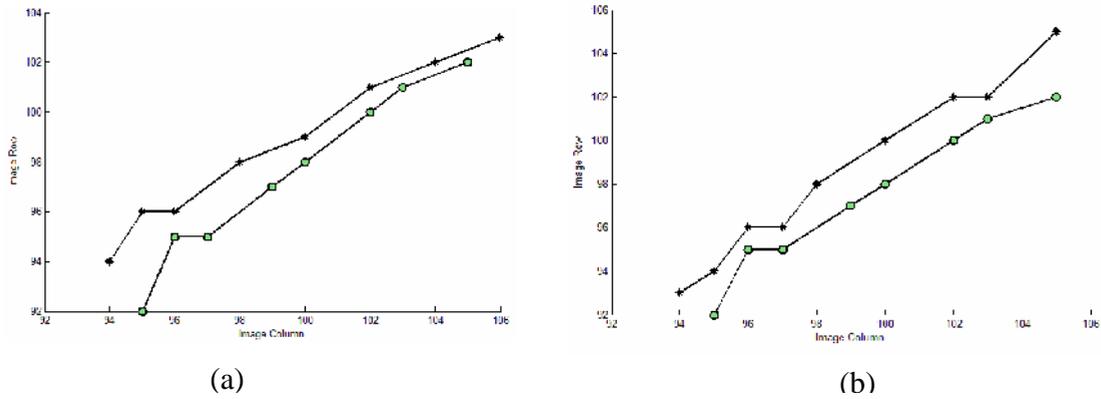


Figure (7): Actual trajectory (o) and points Detected (*). (a) using CCS corners extractor, (b) using SUSAN corners extractor.

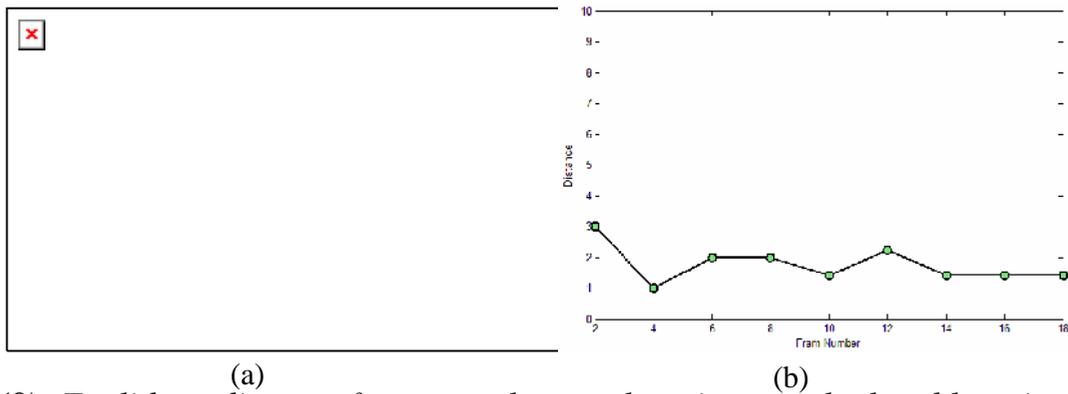


Figure (8): Euclidean distance from actual target location to calculated location (a) using SUSAN corners extractor, (b) using CCS corners extractor.

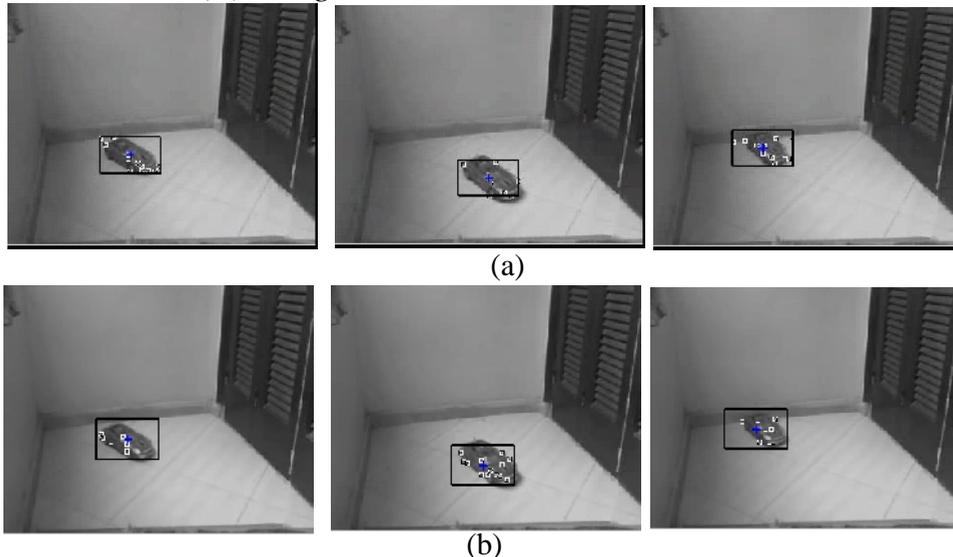


Figure (9): selected image from " wireless car "image sequences after applying the tracking algorithm.(a) frame 1 ,frame 60 and frame 100 in sequence using CCS corners extractor .(b) frame 1 ,frame 60 and frame 100 in sequence using SUSAN corners extractor. Each "+" sign points to the tracked target detected by the correspondening algorithm.

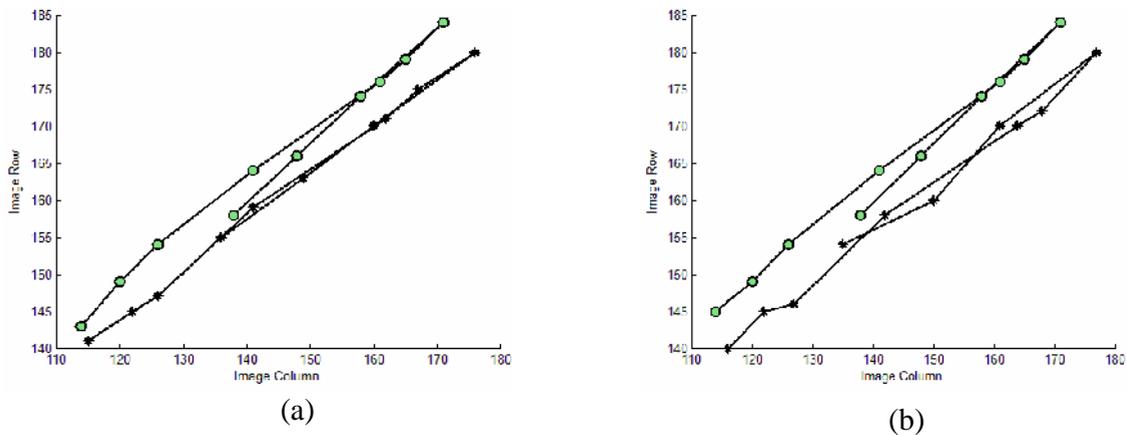


Figure 10): Actual trajectory (o) and points Detected (*). (a) using CCS corners extractor, (b) using SUSAN corners extractor.

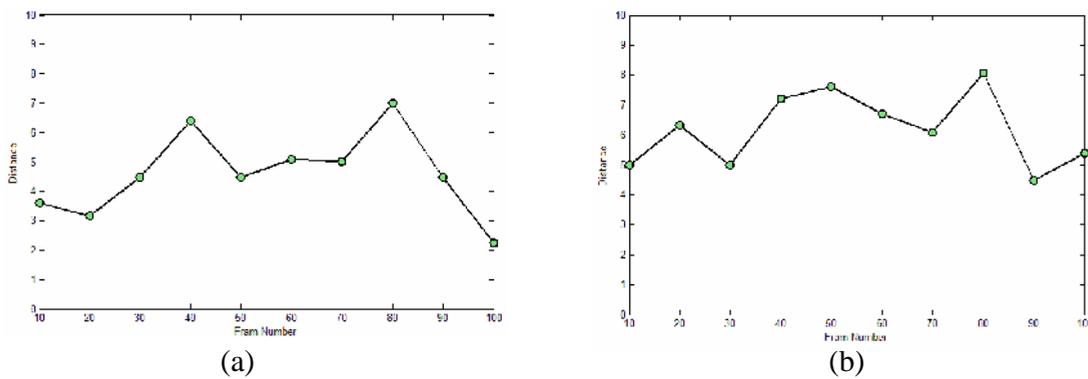


Figure 11): Euclidean distance from actual target location to calculated location (a) using SUSAN corners extractor, (b) using CCS corners extractor.

5. Conclusions:

The proposed tracking technique introduced in this paper can track a moving object in an image sequence where the camera is fixed. The execution time of this technique is very low compared with corner tracking using SUSAN corner extractor, so it would be suitable for real time applications. The error between the calculated location and actual target location was never more than 8 pixels using CSS corner extractor method and 10 pixels using SUSAN corner extractor method. The technique described in this paper can assist other computer vision, such as object recognition, by always keeping the object of interest in view for studying and recording the motion of the object being tracked, and This may allow the development of a path-based object recording systems.

References:

- [1] Zalili Musa and Junzo Watada, *Vidio Ttracking System A Survey*, ICIC International, Vol 2, P.65-72, March 2008.
- [2] Alper Yilmaz, Omar Javed, and Mubarak Shah , *Object tracking A survey*, ACM Computing Surveys, Vol. 38, No. 4, P.45, December 2006.
- [3] Chris Harris. and Mike Stephens, *A combined corner and edge detector*, In 4th Alvey Vision Conference. P 147-151, 1988.
- [4] Springer Berlin and Heidelberg, *Real-Time Optical Edge and Corner Tracking at Subpixel Accuracy*, *Computer Analysis of Images and Patterns*, P.836, June 1999.
- [5] Romer Rosales and Stan Sclaroff, *3D Trajectory Recovery for Tracking Multiple Objects and Trajectory Guided Recognition of Actions*, IEEE Conf. on Computer Vision and Pattern Recognition, June 1999.
- [6] M. Asadi and C.S. Regazzoni, *A Comparison Of Different Approaches to Nonlinear Shift Estimation For Object Tracking*, IEEE International Conference on Image Processing, P.16-19, September, 2007.
- [7] Xiao Chen He and Nelson H. C. Yung, *Corner detector based on global and local curvature properties*, *Optical Engineering*, Vol 47, P.057008, May 2008.
- [8] L. Marcenaro, L. Marchesotti, and C.S. Regazzoni, *Self organizing shape description for tracking and classifying multiple*, Elsevier Image and Vision Computing, P.1179-1191, 2006.
- [9] P.Gabriel, J.BHayet and J.Verly, *Object tracking using color interest points*, IEEE AVSS, P.159-164, Italy, September 2005.