Adaptive Variable-size Search Window based on SURF

Heba kandil, Eman Eldaydamony and Ahmed Atwan

Information Technology Department, Faculty of Computer and Information Sciences, Mansoura University,El-Gomhoria St.,35516, Egypt.
heba_kandil@mans.edu.eg , atwan@mans.edu.eg
Fax:+ 2 35516, Tel: +20 1003250373

Abstract

Video tracking is a rich research point nowadays due to its wide range of applications such as surveillance. One of the challenges in video tracking is to exactly determine the location of the tracked object within each frame. Most of tracking algorithms make use of a fixed size search window regardless of the tracked object scale change over time. The fact is that too small search window may lose details of the tracked object. Besides, undue increase of computational complexity is resulted of inaccurate large search window. Adaptive variable-size search window algorithm is proposed to overcome these problems. Even if the tracked object is partially or completely occluded the algorithm should locate the expected location of it in an efficient way. The proposed algorithm is based on speeded up robust features (SURF). SURF is one of the fastest descriptors which generate a set of interest points that are invariant to various image deformations and robust against occlusion conditions during tracking. SURF points of the tracked object are extracted from the initially determined search window. The proposed algorithm makes use of the positional information of the extracted SURF points to update the size and location of the search window in the following frames. The results achieved more accuracy of the tracking process. The proposed algorithm produces a search window that is more fitted to the tracked object than search windows produced by common tracking algorithms such as mean shift do. Any tracking algorithm can make use of the proposed algorithm as it works in parallel with it to update the search window location and size to precisely track the object. Less computational time in the search window is an added value. Prediction of the exact location of the tracked object under occlusion condition is more precise than existing algorithms.

Keywords: *SURF, Mean shift, Search window, visual tracking*

1. Introduction

Visual tracking is the job of locating, tracking and analyzing one or more objects in a video. Many important applications today are mainly based on visual tracking such as human-computer interaction, surveillance, video editing, vehicle navigation etc. In such applications, tracking algorithms are used to analyze video frames and extract the object(s) of interest throughout the continuous sequence of frames. Many challenges affect the efficiency of a tracking algorithm such as the presence of lightening change, occlusions, scaling change and determining the search window for the tracked object(s). Determining the search window aims to track the region of interest throughout the sequence of frames. Mean shift is used as a

tracking algorithm in [1] and uses some kind of kernel functions to determine the shape and size of search window. The limited nature of kernel bandwidth leads to inappropriate determination of the size of the search window in cases of scale change [2]. CAMSHIFT [3] introduced the modified mean shift version that adjusts the size of the search window by invariant moments. CAMSHIFT adjusts the size of the search window dynamically during its operation. The initial size of the search window could be set at any reasonable size, and then CAMSHIFT extracts a piece of information called the zeroth moment on which it relies to continuously adapt the search window size [3]. On the other hand, the computation of moments hinders the real time property [2]. Trackers introduced in [3, 4, 5] are based on searching for a fixed shape variable- size window that best matches a reference model based on color content. Problems exhibit when colors of background are similar to those of the tracked object or when parts of the tracked object are completely occluded over a sequence of frames. If the window size is too large, it may include objects other than the tracked one. On the other hand, if the window size is too small, it may miss important details of the tracked object. In [6] an adaptive search window technique is achieved by inter-frame texture analysis. The temporal texture analysis of frames is used to find the direction and speed of the tracked object. The search window location is updated based on the speed and direction of the object motion. [6] Achieved good performance of tracking under occlusion conditions particularly in crowds. The challenge of this technique is the period and procedure it uses in the updating process [6]. In addition, the tracked object falls outside the search window in presence of noise, occlusion, abrupt transformation and zooming.

This paper is organized as follows. Section 2 presents SURF descriptor. Mean shift algorithm is discussed in Section 3. The proposed adaptive variable-size search window algorithm is presented in Section 4. Experimental results are given in Section 4. Finally conclusion and future work are presented in Section 5.

2. Speeded Up Robust Features (SURF)

Local invariant features are known to perform well in pattern recognition problems due to their robustness, distinctiveness and repeatability characteristics. A comparison and evaluation of different descriptors is presented in details in [8].

The task of finding correspondence between images of the same scene or object is essential in many computer vision applications. This can be achieved using three steps namely, detection, description and matching [7]. In detection step, interest points are selected from distinctive locations in an image such as corners and blobs. These interest points should be distinctive and repeatable, that's, they could be detected under different and even sever viewing conditions. In description step, the neighborhood of each interest point is represented by a feature vector. This process should be robust to noise, detection errors and geometric and photometric deformations. Finally, in matching step, feature vectors of different images are matched. This is usually done based on the distance between features vectors, *e.g.*, Euclidean distance for example.

Herbert Bay, *et al.*, [7] introduced the local invariant interest points' detector-descriptor (SURF). SURF is invariant to common image transformations, rotation, scale change, illumination change and small change in viewpoint.

SURF uses integral images (summed area tables), which are intermediate representations for the image and contain the sum of gray scale pixel values of image, to reduce computation time. The detector is based on Hessian matrix to make use of its good performance in computation time and accuracy.

Given a point x= (x, y) in an image I, H(x, σ) is the Hessian matrix in x at scale σ defined as:

$H(x, \sigma) = \begin{bmatrix} l_{xx}(x,\sigma) & l_{xy}(x,\sigma) \\ l_{xy}(x,\sigma) & l_{yy}(x,\sigma) \end{bmatrix}$, Where $l_{xx}(x,\sigma)$, $l_{xy}(x,\sigma)$ and $l_{yy}(x,\sigma)$ represent the convolution of the Gaussian second order derivative $\frac{\partial^2}{\partial x^2} g(\sigma)$ with image I in point x.

The descriptor makes use of Haar-wavelet responses within the interest point neighborhood. SURF descriptor works as follows: firstly, identify a reproducible orientation based on information from a circular region around the point of interest. Then, it builds a square region aligned to the selected orientation and extracts its SURF descriptor.

A. Orientation assignment: Firstly, Haar-wavelet responses in x and y direction are calculated in a circular neighborhood of radius 6s around the interest point, s is the scale that the interest point was detected at. The Haar-wavelet responses are represented as vectors. Then, all responses within a sliding orientation window covering an angle of 60 degree are summed. Both horizontal and vertical responses in the window are summed yielding a new vector. The longest such vector is the dominant vector.

B. Description: This step includes constructing a square region which is centered around the interest point, and oriented along the selected orientation. Then, the interest region is split into 4x4 square sub-regions with 5 x 5 regularly spaced sample points inside. Haar wavelet responses d_x and d_y are calculated, where d_x, d_y are the Haar wavelet response in horizontal and vertical directions respectively. These responses are then weighted with a Gaussian kernel centered at the interest point to increase the robustness towards deformations and localization errors. The responses d_x, d_y over each sub-region are summed up separately forming a first set of entries to the feature vector. To get information about the polarity of intensity changes, sum of the absolute values of the responses $| d_x |$,

$| d_y |$ is extracted.

The intensity structure for each sub-region is described by
$V= (\sum d_x , \sum d_y, \sum |d_x|, \sum |d_y|)$.

Finally, the vector is normalized into a unit length to achieve invariance to contrast.

3. The Mean Shift Algorithm

Mean shift is a non-parametric feature space analysis technique that is used in many application domains including image processing and computer vision. The algorithm iteratively estimates a mean shift vector M from the current mean location $\bar{x}$, which predicts exactly the mean location $\bar{x'}$ of sample points in search window W with radius r [9]. Mean shift was used as a video tracking algorithm that climbs a probability density gradient to identify the highest probability location of the tracked object [9]. Although mean shift is simple and computationally effective, it is based on a single hypothesis and may converge to a local maximum rather than a global maximum [9]. Thus it is possible that mean shift fail to recover the lost tracks. Figure 1 explains the mentioned symbols $\bar{x}, \bar{x'}, W, r, M$.

Mean shift algorithm is implemented using the steps presented in [9] as follows:

1. Suppose we have n_t a finite number of data samples $s_t^{(i)}$ within a region W in the d-dimensional space R^d. Suppose each data samples is weighted according to a weighting function w(.) at time t (Figure 2 (a)).
2. Mean shift vector is calculated M($\bar{x}_{t-1}$) (fig.2 (b),(c)).

$$M(\bar{x}_{t-1}) = \frac{\sum_{i=1}^{n_t} K\left(s_t^{(i)} - \bar{x}_{t-1}\right) w(s_t^{(i)}) s_t^{(i)}}{\sum_{i=1}^{n_x} K\left(s_t^{(i)} - \bar{x}_{t-1}\right) w(s_t^{(i)})} \quad (6)$$

Where the kernel density estimate can be described as

$$\hat{f}(\bar{x}) = \frac{1}{n_x\, r^d} \sum_{i=1}^{n_x} K\left(\frac{s^{(i)} - \bar{x}}{r}\right) w_t^{(i)} \quad (7)$$

Such that $w_t^{(i)}$ can be computed as:

$$w_t^{(i)} \alpha\, w_{t-1}^{(i)} \frac{p(z_t \mid x_t^{(i)}) p(x_t^{(i)} \mid x_{t-1}^{(i)})}{q\left(x_t^{(i)} \middle| x_t^{(i)}, z_t^{(i)}\right)} \quad (8)$$

The general kernel function K(.), such as K_G (Gaussian Kernel), K_U (Uniform kernel) and K_E (Equanechnikov Kernel) can be expressed as following:

$$K_G = \frac{1}{(2\pi)} e^{(-\|x\|^2/2)} \quad (9)$$

$$K_U = \begin{cases} 1, & if\ \|x\| \le 1 \\ 0, & otherwise \end{cases} \quad (10)$$

$$K_E = \begin{cases} (1 - \|x\|^2), & if\ \|x\| \le 1 \\ 0, & otherwise \end{cases} \quad (11)$$

3. Using the mean shift vector $M_r(\bar{x}_{t-1})$, $\bar{x}_{t-1}$ is shifted to the new mean shift position $\bar{x}'_{t-1}$, and $\bar{x}_{t-1}$ is set to $\bar{x}_t$ (fig.2 (d)).
4. Steps 2 and 3 are repeated until $M_r(\bar{x}_{t-1}) < \varepsilon$, such that ε represents the threshold indicating the suitable moving range of $\bar{x}_t$.

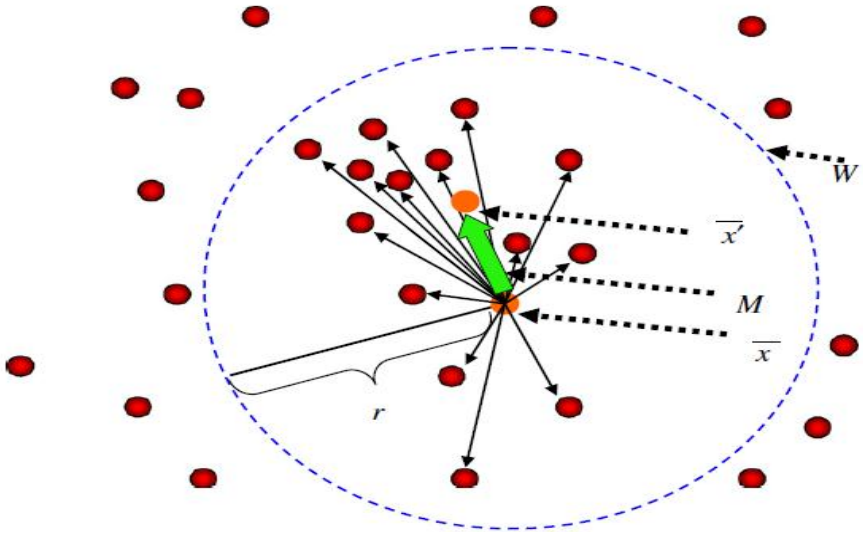

Figure 1. Symbolic Illustration of the Mean Shift Algorithm [9]

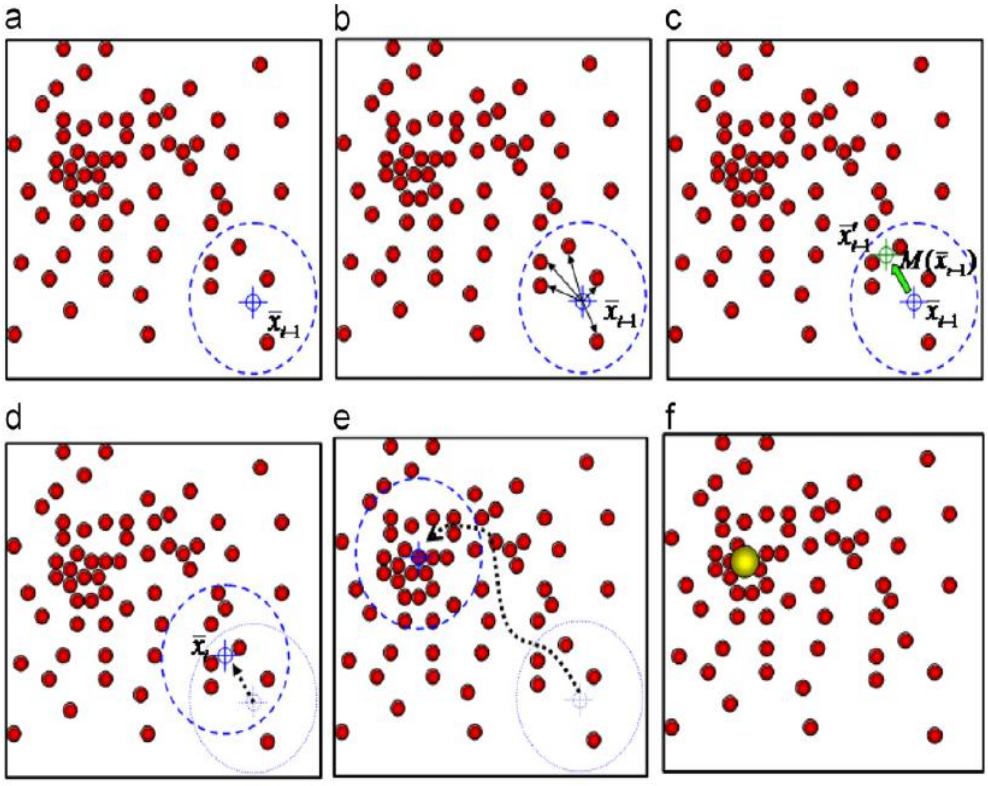

Figure 2. Mean Shift Procedure: (a) Initialization, (b), (c) mean shift vector estimation, (d) shift $\bar{x}_{t-1}$ to the mean position through the mean shift vector and (e),(f)move to the location of mode [9]

4. Proposed Adaptive Variable-Size Search Window Algorithm

To implement the proposed algorithm, a video tracking system was implemented using mean shift tracker as an example of visual tracking algorithm. The proposed search window updating algorithm can be embedded into any other tracking algorithm by just modifying the step in the algorithm that updates the search window and replacing it with the proposed algorithm. The proposed algorithm works in parallel with the main tracker (mean shift in our case) so that it can update the search window in synchronization with the work of mean shift. What is done exactly in the proposed algorithm is represented by the flow chart in Figure 3.

Initially the user should select the target object to be tracked by drawing a rectangle around it. That rectangle represents the initial search window. Then, the mean shift tracker starts its tracking operation while in parallel the SURF algorithm extracts the interest points of the tracked target inside the boundaries of the rectangle (search window). The proposed algorithm processes the resulting SURF interest points and makes use of the position information of each point to find the outer most points. The outer most points are then used to draw the new rectangle representing the new search window. Mean shift tracker should then continue its prediction of the location of the tracked object within the new search window. The process continues until reaching the end of the video or user abruption. Pseudo code of the proposed algorithm is described as follows:

Algorithm pseudo code:

1. Initialize the size of the search window by drawing a rectangle around the target.
2. Set the number of video frames F.
3. Start tracking using the specified tracker (*e.g.*, particle filter or mean shift tracker) and In parallel do the following:
4. Feature Extraction using SURF.
5. Updating the initial search window size according to the resulted SURF feature points
    a. Suppose N is the total number of SURF Points
        i. Suppose L is the list of SURF feature points extracted of size N.
        ii. Split L into 2 lists Lx, Ly each of size N such that Lx contains x-coordinate data and Ly contains y-coordinate data
        iii. Find min(Lx), min(Ly)
        iv. Find max(Lx), max(Ly)
        v. Draw a new rectangle (search window) using the two points ((min(Lx),min(Ly)), (max(Lx), max(Ly)))
    b. Replace the old search window (rectangle) with the new rectangle.
6. Continue until end of F or user interaction.

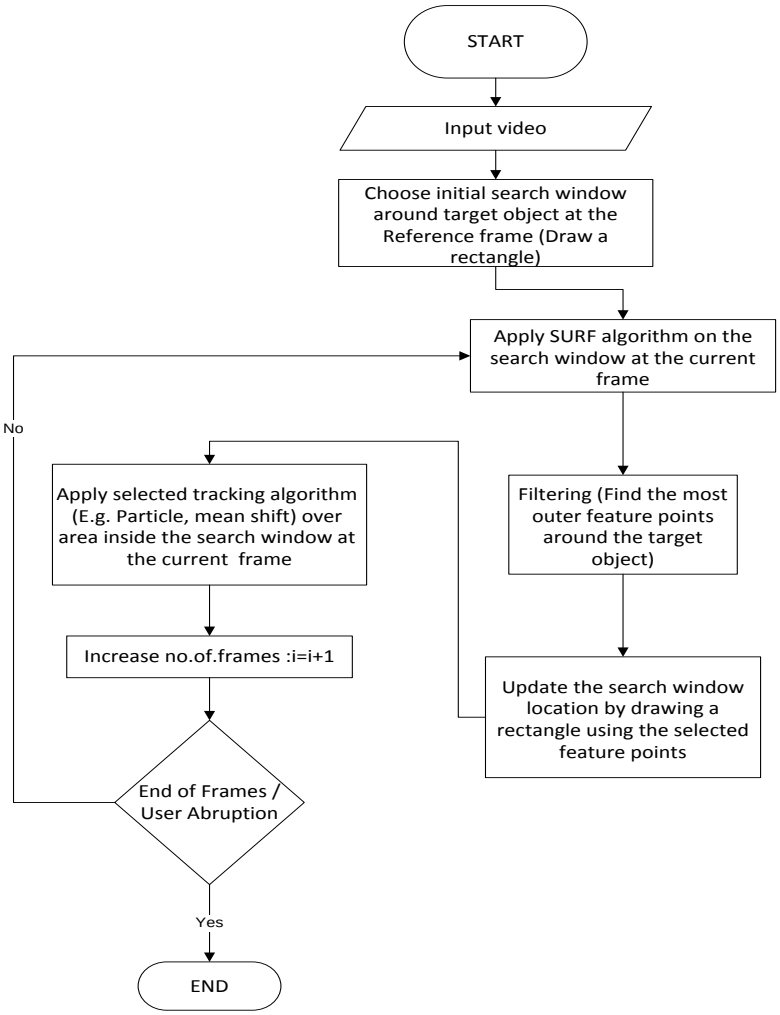

Figure 3. Adaptive Variable-size Search Window Flowchart

5. Experimental Results

The proposed algorithm is tested using dataset that presents various scenarios of people acting out various interactions [10]. Videos are captured at 25 frames per second rate. The resolution is 640 x 480. Videos are in Avi format. Results presented here are based on video named "1-11200.avi".

Figure 4 shows the results of applying the proposed adaptive variable-size search window algorithm on the mean shift tracking system. Rectangle shape represents the search window. Object within the rectangles is the target to be tracked. Rectangle drawn in red represents the search window predicted by the mean shift tracker. The small red circle represents the center of the red rectangle. On the other hand, blue rectangle represents the search window predicted by the proposed algorithm. Blue circle represents the center of the blue search window (rectangle).

As shown in the results, it is clear that the blue search window's size is variable in each frame as it depends on the SURF points extracted in each frame. The size of the blue rectangle is just large enough to include the tracked object. It is appropriately fitted around the object which means low computational cost. On the other hand, the red rectangle

representing the mean shift search window has a fixed size and can be adaptive only based on color feature. In contrast, the proposed algorithm is updated based on SURF.

At the bottom of fig.4 the occlusion case takes place where more than one person enters the search area. Because the mean shift search window is fixed, it included all the persons without any distinguishing. In contrast, it is obvious that the proposed algorithm focuses on the tracked object more accurately.

It is noticed that the blue search window doesn't include the whole body of the target although it includes a large part of it. That is due to the idea behind SURF. SURF was built based on scale invariant feature transform (SIFT) [11]. SURF was designed to be a version that is faster than SIFT. The increase of speed is a result of extracting only the most distinguishing features and ignoring any other less discriminative features. So, SURF may discard a set of interest points to gain more speed. On the other hand, the results obtained by SURF is not only similar to SIFT's, but also better than them in the matter of speed, computational time and performance.

Figure 5 is a data sheet example that demonstrates the idea, centers of the tracked object, mean shift's search window and the proposed algorithm's search window, represented by X and Y coordinates, are given. Again it is clear that the center of the search window produced by the proposed algorithm is closer to the exact center of the tracked object than the mean shift's search window is.

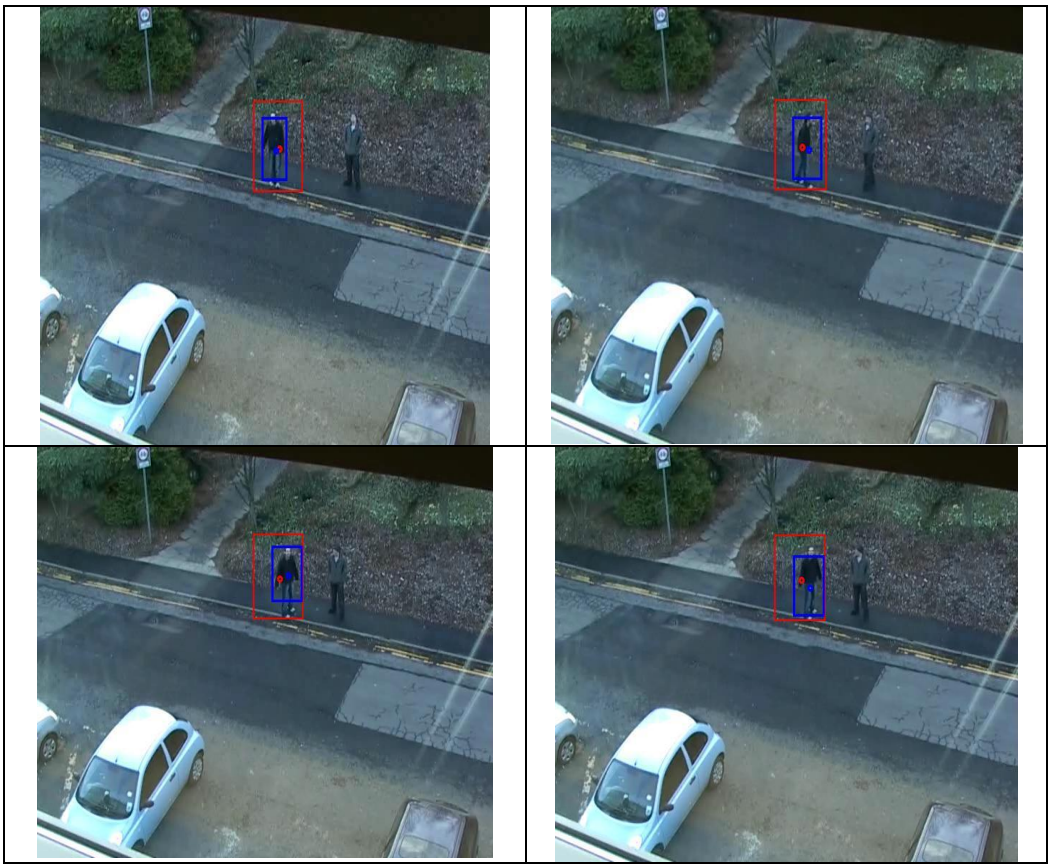

Figure 4. Results of Applying Adaptive Variable-size Search Window Algorithm

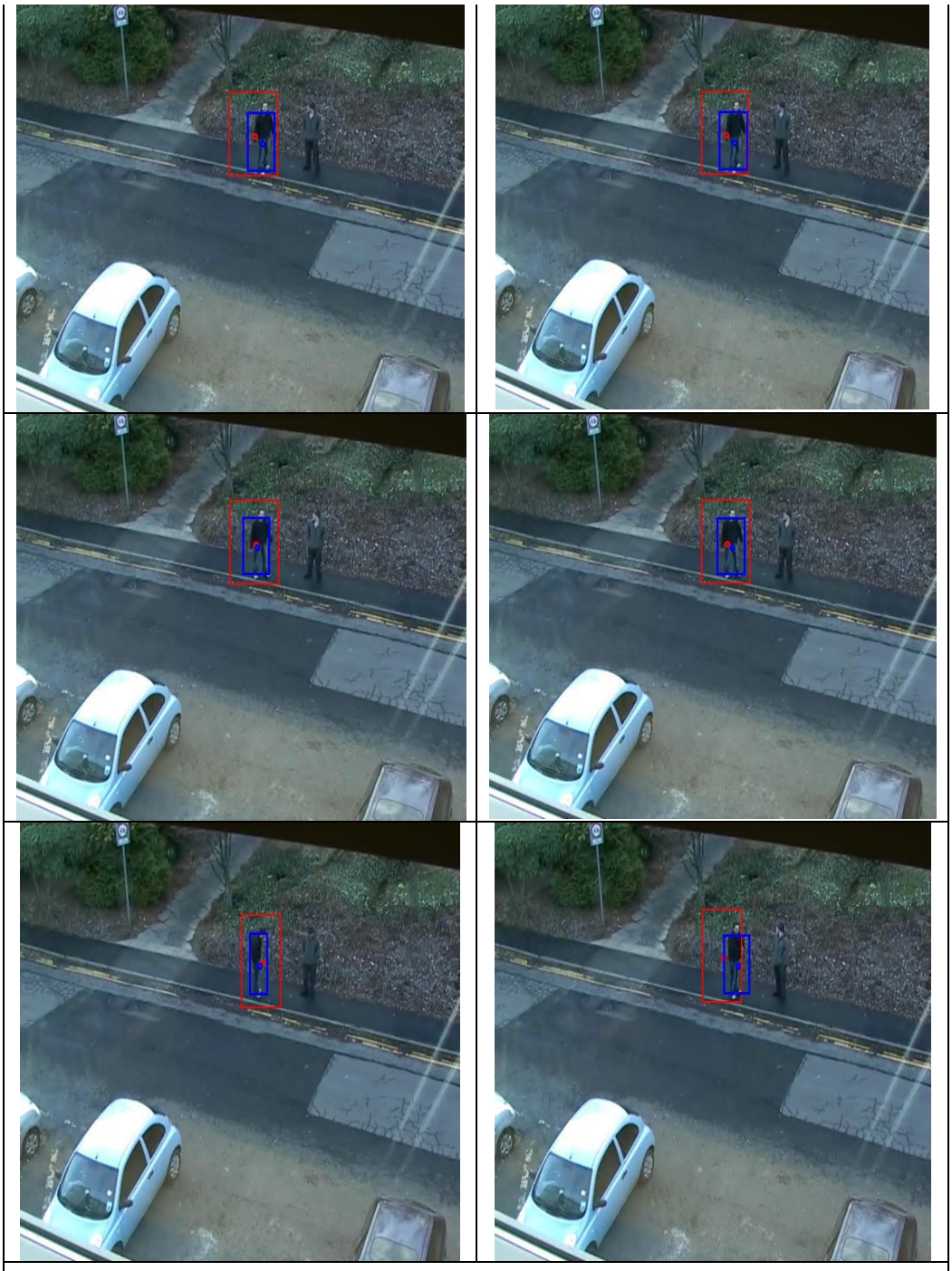

Figure 4. Results of Applying Adaptive Variable-size Search Window Algorithm (Continued)

Figure 4. Results of Applying Adaptive Variable-size Search Window Algorithm (Continued)

Figure 4. Results of Applying Adaptive Variable-size Search Window Algorithm (occlusion condition)

Center of object		Mean shift's search window center		Proposed search window center	
x	y	x	y	x	y
332	159	338	153	333	155
341	161	338	152	347	155
352	159	338	152	350	161
343	157	350	152	346	155
352	153	333	146	354	155
367	147	366	147	364	146
379	138	375	146	378	135
366	132	360	135	367	128

Figure 5. Data Sheet of the Center Point of the Tracked Object

6. Conclusions and Future Work

The proposed adaptive variable-size search window algorithm proved a high performance and efficiency than usual similar techniques. By having the search window compacted around the object, we guarantee less computational cost, better recovery during occlusion condition and good performance in lightening or scaling changes cases.

Future plans include implementing variations of SURF such as upright-SURF (U-SURF). U-SURF proved to be faster than SURF at the expense of reliability and performance. Enhancing the performance and reliability of the U-SURF is another goal in our future plans.

References

[1] D. Serby, S. Koller-Meier and L. V. Gool, "Probabilistic Object Tracking Using Multiple Features", Computer Vision Laboratory (BIWI), ETH Zürich, Switzerland, **(1998)**.
[2] Z. Chaoyang, "Video Object Tracking using SIFT and Mean Shift", Master Thesis in Communication Engineering, Department of Signal and System, Signal Processing Group, Chalmers University of Technology (CTH), Sweden (Report: Ex005/2011),Gothenburg, **(2011)**.
[3] G. R. Bradski, "Computer Vision Face Tracking as a Component of a Perceptual User Interface", IEEE Work, On Applic. Comp. Vis., Princeton, **(1998)**, pp. 214-219.
[4] H. T. Chen and T. L. Liu, "Trust-region methods for real-time tracking", In Proc. Int. Conf. Computer Vision, Vancouver, Canada, **(2001)** July, pp. 717–722.
[5] D. Comaniciu, V. Ramesh and P. Meer, "Real-time tracking of non-rigid objects using mean shift", In Proc. Conf. Comp. Vision Pattern Rec., Hilton Head, SC, **(2000)** June, pp. 142–149.
[6] M. Khansari, H. R. Rabiee, M. Asadi and M. Ghanbari, "Occlusion handling for object tracking in crowded video scenes based on the undecimated wavelet features", in: IEEE/ACS International Conference on Computer Systems and Applications, **(2007)**, pp. 692-699.
[7] H. Bay, T. Tuytelaars and L. Van Gool, "SURF: Speeded Up Robust Features", in: 9th European Conference on Computer Vision, **(2006)** May 7-13.
[8] K. Mikolajczyk and C. Schmid, "A performance evaluation of local descriptors", IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 27, no. 10, **(2005)** October, pp. 1615-1630.
[9] I. -C. Chang and S. -Y. Lin, "3D human motion tracking based on a progressive particle filter", Pattern Recognition, **(2010)** May 07.
[10] BEHAVE Interactions Test Case Scenarios, **(2012)** June 30, http://groups.inf.ed.ac.uk/vision/BEHAVEDATA/INTERACTIONS/.
[11] D. Lowe, "Distinctive image features from scale-invariant keypoints", in: International Journal of Computer Vision, vol. 60, **(2004)**, pp. 91-110.

Authors

Associate prof. Dr. Ahmed Atwan graduated with B.Sc. of electronics and communication from the faculty of engineering. He got M.Sc. in electrical communications 1998. He got Ph.D. in communications 2004. He works at the faculty of computer science and information systems, Mansoura university, Egypt. His current research interests are networks, computer vision, pattern recognition, image processing and others.

Heba Kandil graduated with B.Sc. of computer science from faculty of computer science and information systems, 2008. She got M.Sc. in computer vision 2012. She works as a teaching assistant at the faculty of computer science and information systems, Mansoura university, Egypt since 2009.

Dr. Eman Eldaydamony graduated with B.Sc. of electronics and communication from the faculty of engineering 1998. She got M.Sc. in electrical communications 2003. She got Ph.D. in communications 2008. She works as a lecturer at the faculty of computer science and information systems, Mansoura University, Egypt. Her current research interests are computer vision, biometrics, image processing and others.