

# We are IntechOpen, the world's leading publisher of Open Access books Built by scientists, for scientists

6,900

Open access books available

186,000

International authors and editors

200M

Downloads

Our authors are among the

154

Countries delivered to

TOP 1%

most cited scientists

12.2%

Contributors from top 500 universities



WEB OF SCIENCE™

Selection of our books indexed in the Book Citation Index  
in Web of Science™ Core Collection (BKCI)

Interested in publishing with us?  
Contact [book.department@intechopen.com](mailto:book.department@intechopen.com)

Numbers displayed above are based on latest data collected.  
For more information visit [www.intechopen.com](http://www.intechopen.com)



# Block Matching-Based Background Generation and Non-Rigid Shape Tracking for Video Surveillance

Taekyung Kim and Joonki Paik

*Graduate School of Advanced Imaging Science,  
Multimedia, and Film Chung-Ang University,  
Korea*

## 1. Introduction

Video analysis and tracking is a fundamental problem of dynamically extracting two-dimensional (2D) information in most visual applications including; image processing, computer vision, video surveillance, image synthesis for contents creation, human-computer-interaction (HCI), video compression, and etc. Most existing video surveillance systems simply record and transmit video for crime investigation and traffic flow monitoring. Recent advances in video technology, however, realize intelligent features such as motion detection, tracking, classification, recognition, synthesis, and behaviour analysis.

A video tracking system consists of a series of computational modules, each of which performs location, recognition, and trace of an object. Since the tracking system utilizes spatio-temporally dynamic information, a moving object should first be differentiated from stationary background. The difference-based tracking algorithm, however, cannot detect an object that moves very slowly. Furthermore two or more objects are considered as a single object when occlusion occurs. Occlusion is another challenge, and adaptive background generation is necessary for robust tracking under unstable environment including illumination change, dynamic shading, and camera jitter [Mckenna et al. 1999]-[Javed et al. 2004].

The active shape model (ASM)-based tracking algorithm localizes non-rigid objects with a priori trained shape information [Mckenna et al. 1999], [Cootes et al. 1992], [Nascimento et al. 2004], [Calvagno et al. 2004], and [Koschan et al. 2003]. After modeling an object's shape, it iteratively performs model fitting with possible combination of motion information. The hierarchical extension of the ASM has been proposed by Lee et al. to accelerate the iterative model fitting process with higher matching accuracy [Lee et al. 2007]. Model fitting in the low-resolution image significantly reduces the amount of computation, and coarse-to-fine estimate of the shape together with Kalman filter provides more robust tracking results.

Both the original and the extended versions of ASM-based tracking are, however, highly dependent on prior knowledge such as the number of landmark points and the shape of models in the training sets. In most cases the number of landmark points is manually determined by examining the training data.

To reduce the computational load a feature-based shape tracking method, called non-prior training active feature model (NPT-AFM), has been proposed in [Shin et al. 2005]. This

feature-based object modeling method can track objects by using the greatly reduced number of feature points rather than taking entire shapes in the existing shape-based methods. The on-the-fly update of the training set and the reduced number of feature points can make a real-time, robust tracking system possible. In spite of improved computational efficiency and occlusion handling performance, the NPT-AFM method still requires additional grouping and updating object's feature points. In addition it does not utilize background information, which contains the most part of information in the image.

Another successful tracking technique is block-matching algorithm to find a matching block from a frame to another one. A block-matching algorithm makes use of a matching criterion to determine whether a given block in a frame matches the search block in the reference frame [Zhang et al. 2004]. The major advantages of the block-matching algorithms are twofold (i) its direct matching nature simplifies motion estimation procedures, and (ii) the block preserves object's features that cannot be easily parameterized. There are many variations of the block-matching algorithm that improves the estimation accuracy and computational efficiency [Zhang et al. 2000], [Stefano et al. 1999], [Hariharakrishnan et al. 2005].

On the other hand the drawbacks of the block-matching algorithm include poor performance with non-rigid shapes and the existence of similar patterns in the background. For example, irregular deformation of non-rigid objects decreases the matching accuracy of block-matching algorithms.

In this paper we present a combined shape and feature-based video analysis for non-rigid object tracking that tightly coupled with an adaptive background generation method to compensate the weakness of block matching. The proposed algorithm includes motion detection using background information, feature extraction, and block matching. The background information is acquired by checking the correlation of block located at the same location in the consecutive frames. The proposed method generates a set of features called shape control points (SCPs) by detecting edges in the neighbouring four directions. SCPs are evenly distributed on the contour of the object, and the block-matching-based tracking algorithm is performed on the block containing the corresponding SCP.

In order to further improve the accuracy of block matching together with background generation, we additionally adopt periodic evaluation of the centroid of SCPs in every few frames to preserve the overall shape. We then compare and update each SCP with the centroid during the tracking process, where stray SCPs are removed, and the tracking continues with only qualified SCPs. As a result, the proposed method efficiently removes potential failing factors caused by spatio-temporal similarity between object and background, object deformation, and occlusion.

This chapter is organized as follows. Section 2 presents the fundamentals of ASM. We briefly introduce the NPT-AFM method in section 3, and the definition of SCP and the associated block matching-based non-rigid object tracking algorithm is explained in section 4. We present experimental results in section 5 and finally conclude the chapter in section 6.

## 2. Active shape model (ASM)

Finding the shape and location of an object is a fundamental task for tracking an object in sequential video images. Within the class of deformable models, the boundary information of the object is used to represent the object. For tracking a non-rigid object with prior knowledge ASM is one of the best approaches in the sense of both accuracy and efficiency.

In order to track an object, ASM uses a priori information about the target object such as the shape. Thus it can match and extract the outline of the object in the noisy, or occluded image. The prior knowledge about the object forms a training set that includes variously posed target objects. The training set can be built either automatically or manually by selecting landmark points on the boundary of the object in the sample images. The landmark points should have the distinguishable features of the object such as corner point. ASM-based object detection algorithm consists of four steps; (i) assignment of landmark points, (ii) principal component analysis (PCA), (iii) local structure modeling, and (iv) model fitting.

### 2.1 Obtaining landmark points

Given an input image, landmark points can be obtained by selecting proper feature points on the object's boundary. The feature points in one frame should be correlated to those in the next frame. We can represent  $n$  landmark points as a  $2n$  dimensional vector in a two-dimensional (2D) image as

$$X = [x_1, \dots, x_n, y_1, \dots, y_n]^T \quad (1)$$

We used 42 landmark points in the experiments. [Tanimoto et al. 1975] proposed an automatic landmark assignment method. The positions of landmark points are iteratively updated to minimize the differences with the real boundary of the target object.

### 2.2 Principal Component Analysis (PCA)

A set of  $n$  landmark points which is one member of the training set represents the shape of an object. Instead of using all landmark points in a member of the training set, PCA technique helps to model the shape of the object with less number of parameters. Let us assume that there are  $m$  members in the training set and  $x_i$  represents each member ( $i = 1, \dots, m$ ). The PCA algorithm is summarized in the following.

1. Find an average of  $m$  members.

$$\bar{x} = \frac{1}{m} \sum_{i=1}^m x_i \quad (2)$$

2. Find a covariance matrix from the training set.

$$S = \frac{1}{m} \sum_{i=1}^m (x_i - \bar{x})(x_i - \bar{x})^T \quad (3)$$

3. Obtain an eigenvector that has the  $q$  biggest eigenvalues from the covariance matrix  $S$ .  $q$  is defined to cover 98% of the variance of total data.

$$\Phi = [\varphi_1 | \varphi_2 | \varphi_3 | \dots | \varphi_q] \quad (4)$$

4. Approximate the shape of the object from the obtained  $\varphi$  and  $\bar{x}$  as

$$x_i \approx \bar{x} + \Phi b_i \quad (5)$$

and

$$b_i = \Phi^T(x_i - \bar{x}) \quad (6)$$

Vector  $b$  can be defined as a set of deformable model parameters and implies the shape of the object.

### 2.3 Local structure modeling

In order to analyze the shape of the target object we have to find the best set of landmark points that matches the object and the model. At each iteration the landmark points selected by PCA algorithm are relocated to the edge of the object along the line which is vertical to the boundary of the real object as shown in fig. 1.

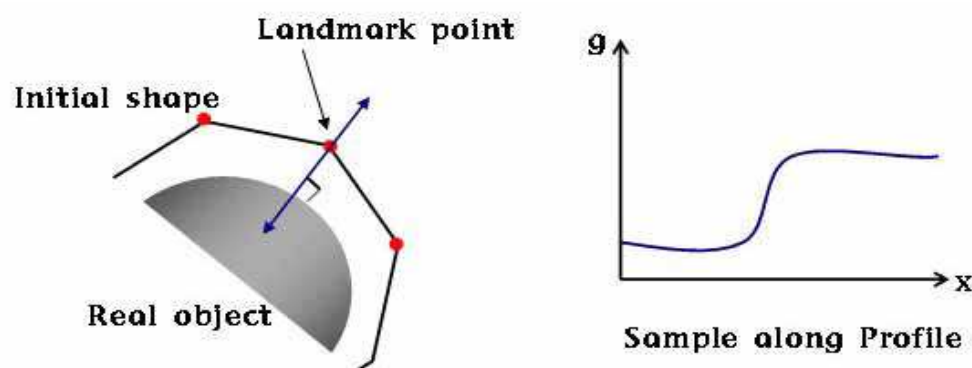


Fig. 1. Local structure modeling

### 2.4 Model fitting

The best parameters that represent the optimal location and shape of the object can be obtained by matching the shape of models in the training set to the real object in the image. The matching is performed by minimizing the error function as

$$E = (y - Mx)^T W (y - Mx) \quad (7)$$

where  $x$  represents the coordinates of the model,  $y$  the coordinate of the real object,  $W$  a diagonal matrix whose diagonal element is the weight to each landmark points,  $M$  a matrix for the geometrical transformation which consists of rotation  $\theta$ , transition  $t$ , and scaling factor  $s$ . The weight decides the distance between previous and new landmark points.

The geometrical transformation matrix for a single point  $(x_0, y_0)^T$  can be represented as

$$M \begin{bmatrix} x_0 \\ y_0 \end{bmatrix} = s \begin{bmatrix} \cos \theta & \sin \theta \\ -\sin \theta & \cos \theta \end{bmatrix} \begin{bmatrix} x_0 \\ y_0 \end{bmatrix} + \begin{bmatrix} t_x \\ t_y \end{bmatrix} \quad (8)$$

Once the set of geometrical parameters  $(\theta, t, s)$  is determined, the projection of  $y$  to the frame of model parameters is given as

$$x_p = M^{-1}y \quad (9)$$

The new model parameter  $b$  is updated as

$$b = \Phi^T(x_p - \bar{x}) \tag{10}$$

With the model parameters from equation (10) a new shape that consists of new landmark points is obtained by equation (5). The new shape can be used in equation (7) and the model fitting process repeats until optimal landmark points are achieved. After some iteration of model fitting, we can achieve the final shape  $x$ .

3. Non prior training active feature model (NPT-AFM)

In this section we present a feature-based object tracking algorithm using optical flow under the non-prior training active feature model (NPT-AFM) framework which generates training shapes in real-time without pre-processing. The NPT-AFM algorithm extracts moving objects by using motion between frames, and determines feature points inside the object. Selected feature points in the next frame are predicted by a spatio-temporally adaptive algorithm. If a feature point is missing or tracking fails, correction process restores feasible feature points. The NPT-AFM can track deformable, partially occluded objects by using the greatly reduced number of feature points rather than taking entire shapes in the existing shape-based methods. Therefore, objects can be tracked without a priori information or constraint with respect to the camera position or object motion.

The flowchart of NPT-AFM algorithm is shown in fig. 2.

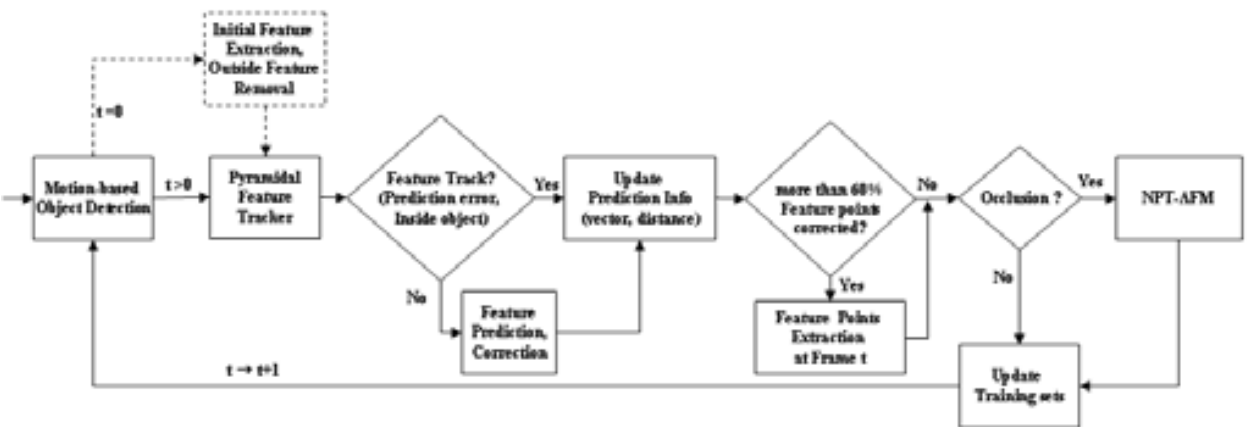


Fig. 2. Non-prior training active feature model (NPT-AFM) algorithm

In fig. 2, the upper, dotted box represents the algorithm of initial feature extraction. The remaining part of the flowchart relates two consecutive frames. In the motion-based object detection block, we extract motion by the simple Lucas-Kanade optical flow [Tekalp 1995] and classify object's movement into four directions such as left, right, up, and down. As a result, the region of moving objects can be extracted and the corresponding region is then labelled based on the direction of motion.

After detecting moving objects from background, we extract a set of feature points inside the object and predict the corresponding feature points in the next frame. We keep checking and restoring any missing feature points during the tracking process. If over 60% of feature points are restored, we decided the set of feature points are not proper for tracking, and



redefine new set of points. If occlusion occurs, the NPT-AFM process, which updates training sets at each frame to restore entire shapes from the occluded input, is performed to estimate the position of occluded feature points.

The advantages of the NPT-AFM algorithm can be summarized as: (i) it can track both rigid and deformable objects because a general feature-based tracking algorithm is applied, (ii) it is robust against object's sudden motion because motion direction and feature points are tracked at the same time, (iii) its tracking performance is not degraded even with complicated background because feature points are assigned inside an object rather than near boundary, and (iv) it contains the NPT-AFM procedure that can restore partial occlusion in real-time.

### 3.1 Feature point extraction

After detecting an object from background, we extract a set of feature points inside the object by using the Bouguet tracking algorithm [Isard et al. 1996][Bouguet 2000]. Due to the nature of motion estimation, motion-based object detection algorithms usually extract the object slightly larger than the real size of the object, which results in false extraction of feature points outside the object. Let the position of a feature point at frame  $t$  be  $v_i^t$ , where  $i$  represents the index of feature points and  $v_i^t = [x_i^t, y_i^t]^T$ . These outside feature points are removed by considering the distance between feature points given as

$$v_i^t = \begin{cases} v_i^t, & d_i \geq T \\ 0, & d_i < T \end{cases} \quad (11)$$

where  $d_i = \sum_{t=0}^{K-1} \sqrt{(x_i^{t+1} - x_i^t)^2 + (y_i^{t+1} - y_i^t)^2}$ ,  $t$  represents the index of frames, and  $i$  the index of feature points. Here  $d_i$  represents the sum of distance between  $t$ -th and  $t+1$ -st frames with respect to the  $i$ -th feature point. In general, the moving distance of a feature point in the background (outside object) is much less than that of a feature point in the tracked object. Although the value of  $T$  is equal to 3.5 or 7 was used for all test sequences, users can control the value depending on the environmental factors such as illumination, noise, and the complexity of background. The results of outside feature point removal are shown in fig. 3. Three feature points outside the 'Tang' indicated by circle in fig. 3(a) are efficiently removed in fig. 3(b).

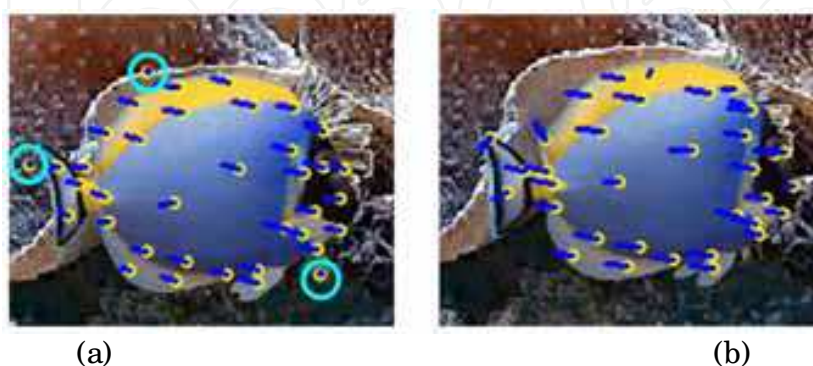


Fig. 3. Results of outside feature point removal: (a) the 2nd frame with three outside feature points highlighted by circles and (b) the 7th frame without outside feature points.

3.2 Feature point prediction and correction

Sometimes, a tracking algorithm may fail in tracking a proper feature point in the next frame. We stop tracking a feature point when an error value within small window is larger than a pre-specified threshold. More specifically, a threshold value is determined by the distance between spatio-temporally predicted vectors. After the spatio-temporal prediction, re-investigation is performed. Then, both tracked and untracked feature points are updated in a list.

In many real-time, continuous video tracking applications, a feature-based tracking algorithm fails due to the following reasons: (i) self or partial occlusions of an object and (ii) feature points on or outside the boundary of the object, which are affected by dynamic background.

In order to deal with the tracking failure, we should correct the erroneously predicted feature points by using the location of the previous feature points and inter-pixel relationship between the predicted points. This algorithm is summarized in table 1.

Step 1	<p><b>Temporal Prediction:</b> If the <math>i</math> th feature point at the <math>t</math> th frame is lost in tracking, it is re-predicted as</p> $\hat{v}_i^{t+1} = v_i^t + \frac{1}{K} \sum_{k=0}^{K-1} m_i^{t-k} \tag{12}$ <p>where <math>m_i^t = v_i^t - v_i^{t-1}</math>, and <math>K</math> represents the number of frames for computing the average motion vector.</p>
Step 2	<p><b>Spatial Prediction:</b> We can correct the erroneous prediction by replacing with the average motion vector of successfully predicted feature points. The temporal and spatial prediction results of Step 1 and Step 2 can be combined to estimate the position of feature points.</p>
Step 3	<p><b>Re-Investigation of the Predicted Feature Point:</b> Assign the region including the predicted and corrected feature points in the spatio-temporal prediction step. If a feature point is extracted within a certain extent in the following frame, it is updated as a new feature point. While the re-predicted feature points are more than 60% of the entire feature points, feature points keeps being estimated.</p>

Table 1. Spatio-temporal algorithm for correction of predicted feature points

A temporal prediction is suitable for deformable objects while a spatial prediction is good for non-deformable objects. Both temporal and spatial prediction results can also be combined with proper weights. Users can control the number of frames,  $K$ . The larger the value of  $K$  is, the better the performance of the algorithm is. Because of the trade-off between processing time and accuracy, the value around 7 was found to be reasonable for temporal prediction.

The existing ASM algorithm manually assigns landmark points on an object’s boundary to make a training set. A good landmark point has balanced distance between adjacent landmark points and resides on either high-curvature or “T” junction position. A good



feature point, however, has a different requirement from that of a good landmark point. In other words, a feature point is recommended to locate inside the object because a feature point on the boundary of the object easily fails in optical flow or block matching-based tracking due to the effect of dynamic, complicated background.

A set of feature points form an element shape in the training set. We update this training set at each frame of input video, and at the same time align the shape onto the image coordinate using Procrustes analysis [Goodall 1991]. In this work, the training set has 70 element shapes.

#### 4. Combined shape and feature-based analysis for non-rigid object tracking

After motion detection, surveillance systems generally track moving objects from one frame to another in the image sequence. Tracking over time typically involves matching an object-of-interest in temporally consecutive frames using features such as a point, a line, or a blob.

The major contribution of the proposed algorithm is the high reliability resulting from the use of background and shape boundary in detecting and tracking a moving object. In the block matching process to extract object's region, there are many empty blocks with only background information. On the other hand the use of the shape boundary significantly alleviates this problem by selecting blocks containing the object and placing feature points on the boundary of the shapes. Also, the proposed method can efficiently remove the misplaced feature points within the given block area using measures explained in the following sections. From the selected feature points it is also possible to select the feature points called as shape control points (SCPs) which play a significant role in the tracking the object.

The proposed combined shape and feature-based object tracking system is depicted in fig. 4. Three modules of the proposed system respectively represent (i) background generation, (ii) motion detection and shape control point (SCP) extraction, and (iii) object shape tracking modules.

As mentioned in the previous section, simple difference-based methods without appropriately generated background often fail to track an object when adjacent frames have little difference. In order to overcome this problem we evaluate difference between the generated background and the input image to detect object region. The detected region is considered as a rough estimate the object's shape, and is tracked by the system using the block matching algorithm (BMA).

In the background generating process, we exclude blocks with moving objects that cause large matching errors. Additional median filtering is necessary to compensate blocking artifacts due to discarded blocks. Since the effect of moving object is blocked by the median filter, the proposed background generation block can detect moving regions and shape variation with higher accuracy.

After detecting the region of moving objects, we compute the object's boundary using morphological edge operations. The boundary information differentiates the object from the background by classifying SCPs and the candidate of shape control points (CSCPs). SCPs are used for tracking object's shape, while CSCPs are used for updating SCPs when deformation or occlusion occurs. The block matching method is used for tracking deformable objects as well as background generation. If occlusion occurs, the current SCPs are replaced by suitable CSCPs depending on the size of moving region and the number of SCPs.

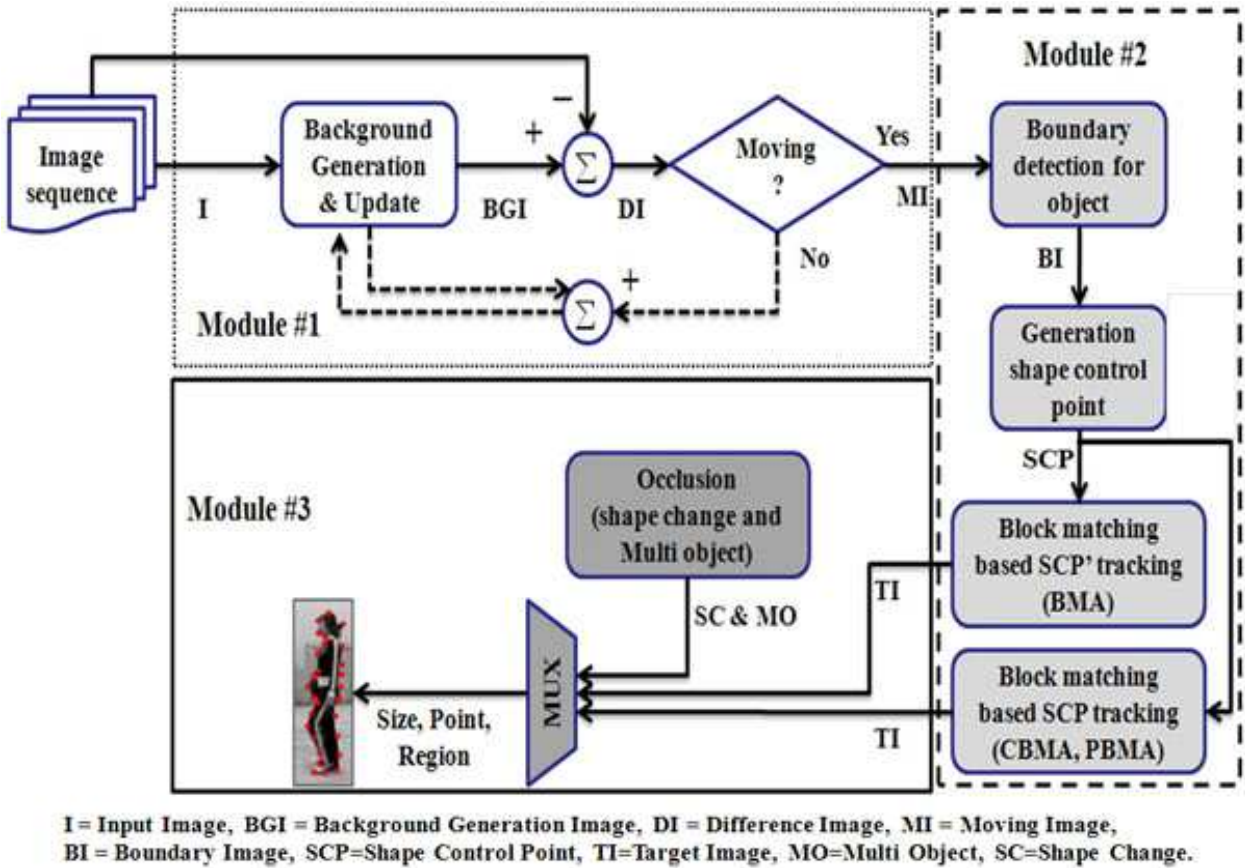


Fig. 4. The block diagram of proposed block matching-based object shape tracking

4.1 Background generation and updating

Once a stable background is generated, the object’s moving region can easily be detected by comparing the generated background and the current input frame. The background is generated using only blocks with the matching error lower than a pre-specified threshold. Thus we can avoid the undesired effects caused by the internal motion and illumination change. Fig. 5 shows a typical process of BMA.

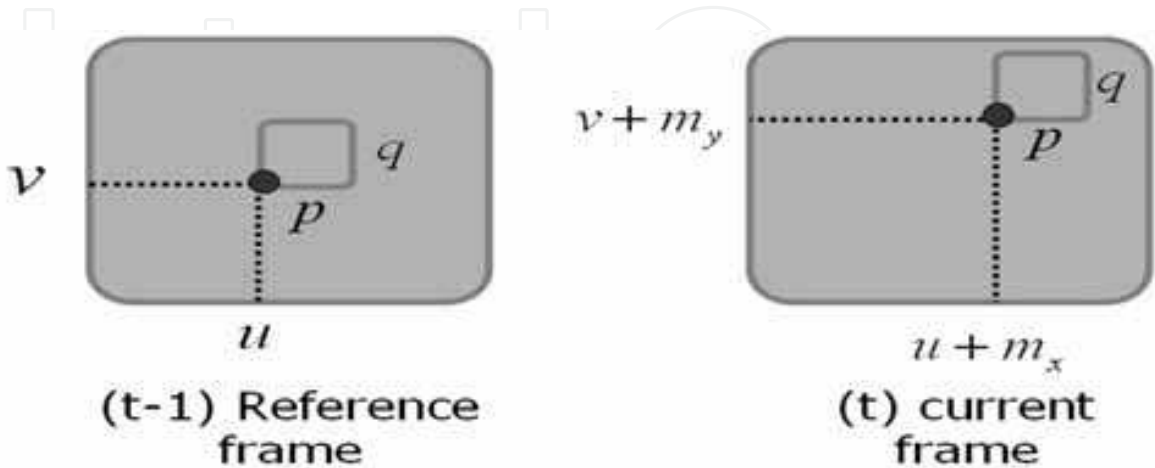


Fig. 5. Block matching algorithm between two temporally adjacent frames

The sum of absolute difference (*SAD*) is adopted as a measure of matching error as

$$SAD(u, v, t; m_x, m_y) = \sum_{i=0}^{p-1} \sum_{j=0}^{q-1} \left| I(i+u, j+v, t-1) - I(i+u+m_x, j+v+m_y, t) \right|, \quad (13)$$

where  $(u, v)$  represents the horizontal and vertical coordinates,  $(m_x, m_y)$  the motion vector, and  $(p, q)$  the horizontal and vertical sizes of the image block. If *SAD* is smaller than an experimentally chosen threshold, background is updated at the corresponding block region. In the experiment we have used 0.05 for the threshold value. For a block with high *SAD* value the background is generated by minimizing the *SAD* value, while the median filter is used for the rest blocks. Although the median filter is robust against noise and illumination change, it is still possible to lose the object due to dynamic environmental factors such as background change, internal reflection, and motion change, to name a few. For removing such dynamic factors, it is necessary to keep updating the background.

The desirable property of background is that it has a constant distribution. Based on this property, only change in video should not affect the constant distribution. For this reason W4 algorithm, for example, separates objects and background using temporal median filter, and as a result it can provide the constant distribution against illumination change [Haritaoglu et al. 2000]. W4 algorithm can also handle fast motion or abrupt change in the image because of the use of median filter.

Object motion can be detected if its amount is greater than the error of a block motion. So the background is generated such that the error is minimized. The background updating process can be expressed as

$$B(t) = (1 - \sigma)I(t) + \sigma B(t-1), \quad (14)$$

where  $B(t)$  represents the background at time  $t$ ,  $I(t)$  the input image at time  $t$ , and  $\sigma$  the mixing ratio in the range  $[0, 1]$ . To differentiate object's moving region from the background, we use the following equation. The initial background  $B(0)$  takes the first frame of the sequence.

$$D(t) = \begin{cases} 1, & \text{if } \|B(t-1) - I(t)\| \geq T \\ 0, & \text{otherwise} \end{cases}, \quad (15)$$

where  $D(t)$  represents the existence of difference between background and the input image frame. As given in (3), if  $D(t)$  is equal to 1, the corresponding region is defined the moving region. Otherwise, the region is considered as background.

#### 4.2 Detection of moving object

Incorporation of shape, feature, motion, and other possible information significantly increases the amount of computation in the video tracking system, which makes real-time tracking difficult. The proposed algorithm uses only SCPs from boundary information for detecting object's shape. The proposed method expresses object's moving region using binary data that is refined by morphological operations and edge detection. It is possible to estimate the object's boundary using the 2<sup>nd</sup> order derivative method.

The 2<sup>nd</sup> order derivative can be applied in both horizontal and vertical directions and object’s boundary information can be calculated based on the results. Derivative information together with morphological operations enables simple, efficient edge extraction. The proposed method is similar to colour composition between background and objects to reduce errors generated by the background difference image. In the proposed algorithm we apply morphological operations on the binary image that is made from the difference image between the previous and the current images. We then finalize the boundary of the object by merging the result of morphological operations with the result of Laplacian. Fig. 6 shows the procedures of the proposed background generation and motion detection algorithm, and fig. 7 show the corresponding experimental results of each step.

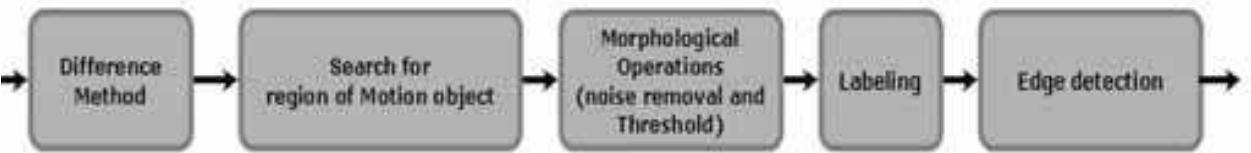


Fig. 6. Moving region detection algorithm based on background generation

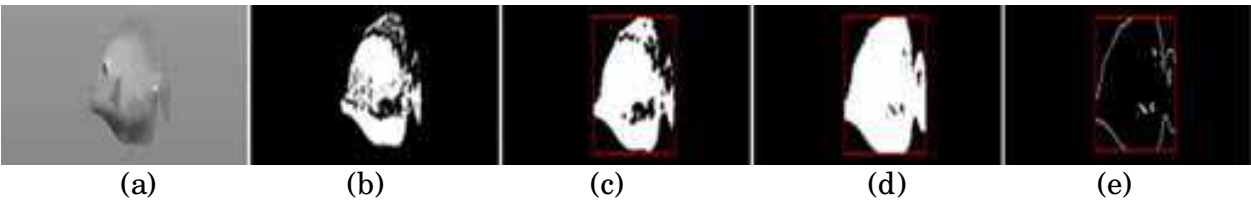


Fig. 7. Experimental results of the proposed moving region detection algorithm: (a) input image, (b) difference image, (c) object’s moving region, (d) result of morphological operation, and (e) result of edge detection

4.3 Shape control points (SCPs)

The object’s boundary information, as shown in fig. 6(e), is used to define SCPs. Since the feature-based tracking methods often fail due to the misidentification of an object, it is necessary to group object’s features by storing the boundary information. Fig. 8 shows the classification result of an image region into one of the background, the object, and the boundary.

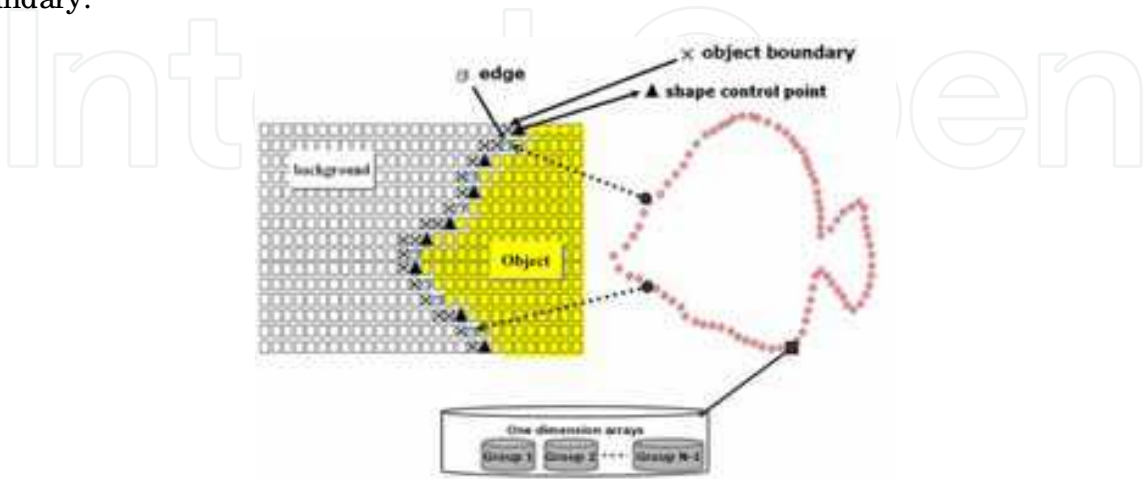


Fig. 8. Classification of meaningful regions for object detection



SCPs are obtained from the object's boundary that is produced by the moving region. Spurious edges are removed by using an empirically chosen threshold. Because matching the entire image is unrealistic in the sense of computational efficiency, the object containing region  $R$  is defined as

$$R(x,y) = \{(r,c) | x_1 \leq r \leq x_2 \text{ and } y_1 \leq c \leq y_2\}, \quad (16)$$

where  $(x_1, y_1)$  and  $(x_2, y_2)$  respectively represent the minimum and maximum coordinates of horizontal and vertical projections. Based on (16)  $R$  represents the minimum rectangular box enclosing the object, whose boundary is detected by Laplacian operator as

$$\nabla^2 R = \frac{\partial^2 R}{\partial x^2} + \frac{\partial^2 R}{\partial y^2}. \quad (17)$$

Only feasible boundary edge points are defined as SCPs, More specifically the  $i$ -th SCP  $A_i$  represents the corresponding coordinate in the minimum rectangle  $R(x,y)$  as

$$A_i = \{(x_i, y_i) | i = 1, 2, \dots, z\}, \quad (18)$$

where  $Z$  represents the total number of edge coordinates.

$$SCP_j = A_{jk} \text{ for } j = 1, 2, \dots, J, J = \lfloor I / k \rfloor. \quad (19)$$

where  $k$  represents the interval of skipping redundant SCPs. The selected set of SCPs is finally stored in a one-dimension (1D) array.

#### 4.4 Combined shape and feature-based object tracking

Although the primary assumption on BMA is that the original and the compared blocks have significantly high correlation, it is not always true in real applications. In this paper we present a modified BMA-based tracking approach, which is robust to occlusion and illumination change, as shown in fig. 9.

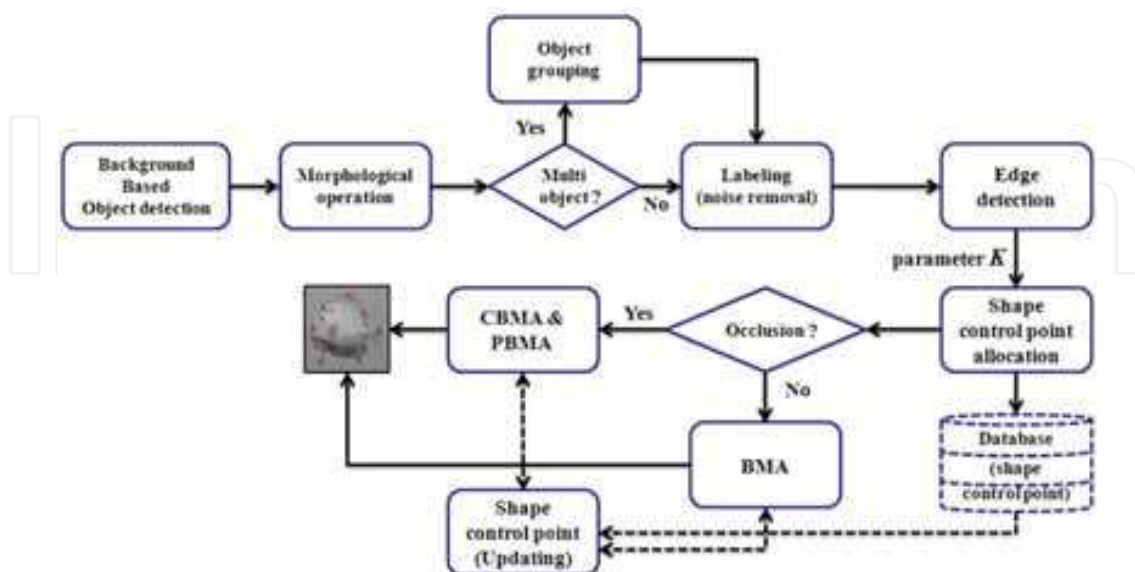


Fig. 9. Block diagram of combined shape and feature-based object tracking

Fig. 9 shows the tracking procedure given a detected object by using the generated background. The detected object is refined by the morphological operation. After labelling the object pixels, edges are detected for extracting SCPs. Extracted SCPs are saved in the database one hand and go through either BMA or CBMA/ PBMA depending on whether occlusion occurs or not. By matching and updating the SCPs, a deformable, moving object can robustly be detected and tracked.

Fig. 10 shows an elaborated illustration for the SCP generation process, where an SCP is allocated at the center of the block. Each block consists of background, an object, CSCPs and SCPs at the center of the block. Location of a deformable object can be detected by using only the SCP.

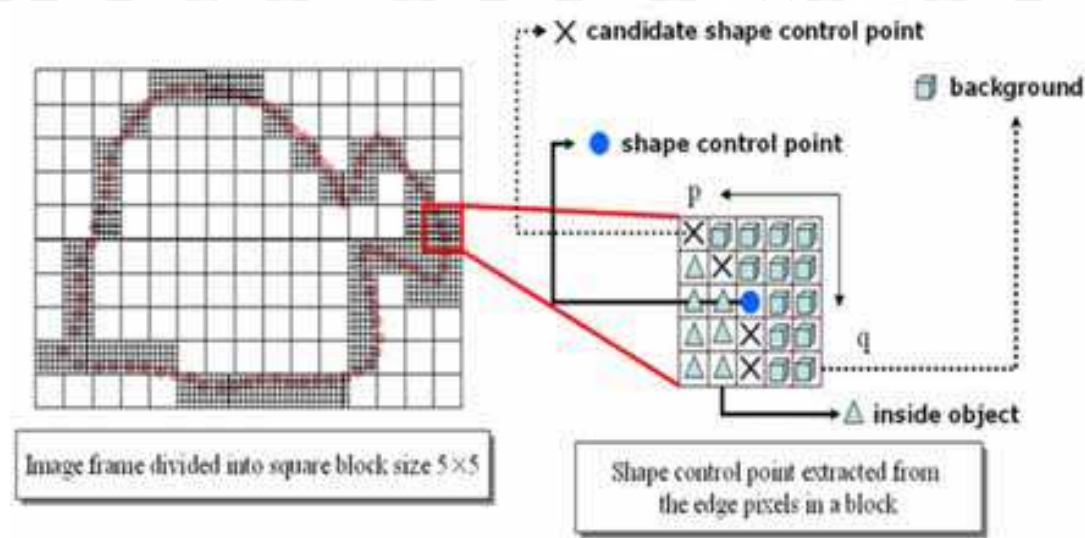


Fig. 10. Generation of an SCP

If an object deforms or occlusion occurs, SCPs cannot provide the correct location of the object. Thus CSCPs are used in this case.

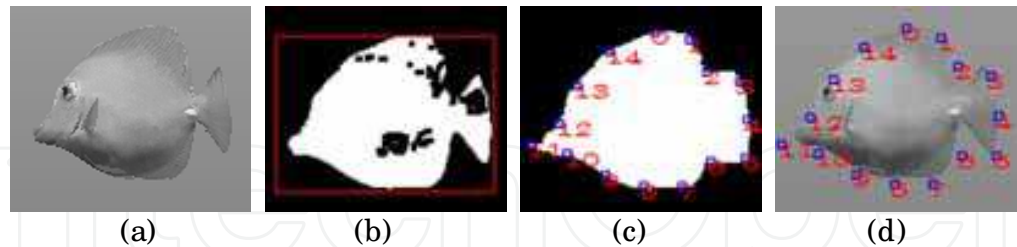


Fig. 11. Model matching: (a) input image, (b) moving region of the object, (c) SCPs on the object's boundary, and (d) assigned SCPs on the input image.

Fig. 11 shows the step-by-step results of the SCP generation process. Fig. 11(b) shows the initially detected object's area, fig. 11(c) shows the SCP's detected on the object boundary, and fig. 11(d) shows the detected SCP's superimposed on the input image.

An object can be tracked by using BMA without background information. However, the result using BMA can easily be affected by similarity between object and background, object deformation, occlusion, illumination change, and etc. In order to minimize the tracking errors we propose two additional methods that combine background generation and block matching.



The first method, which is called the center-of-gravity-based block matching (CBMA) algorithm, preserves the relative location of each SCP by computing distances among SCPs. More specifically, SCPs are maintained inside the possible range of errors, while others are replaced by new SCPs.

The second approach extracts SCPs from the region without motion, which is called the periodic-update-based block matching (PBMA) algorithm. This method can update missing or oscillating SCPs by comparing the corresponding pair of SCPs between two consecutive input frames. SCPs outside the motion region are removed, and tracking is performed using only remaining SCPs. Fig. 12 compares results of the above mentioned three tracking methods, such as BMA, CBMA, and PBMA.

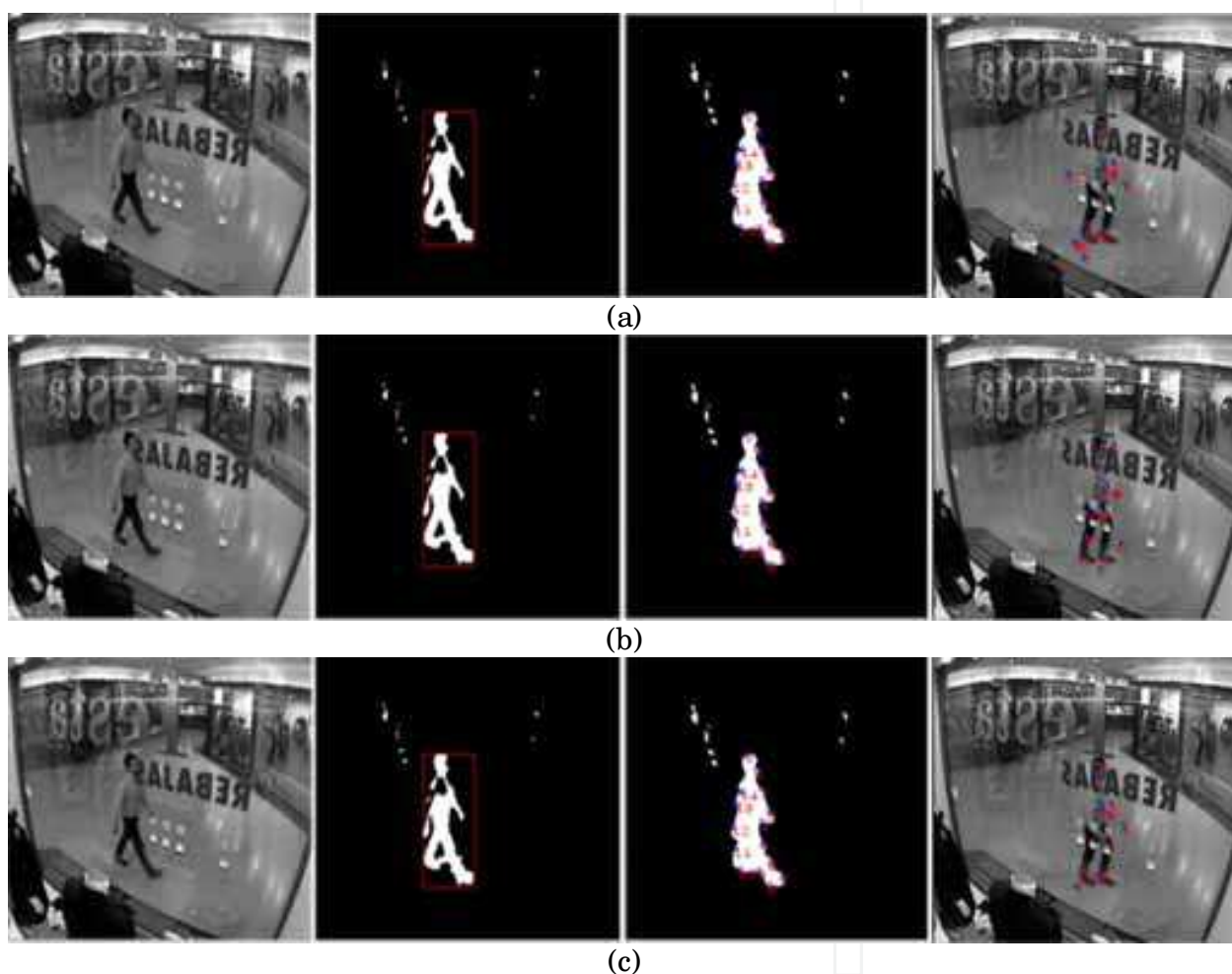


Fig. 12. Experimental results of three different block matching methods for object tracking: (a) BMA, (b) CBMA, and (c) PBMA.

## 5. Experimental results

The proposed object tracking algorithm was tested against shape deformation, occlusion, illumination change, and size change. The proposed algorithm is also quantitatively compared with the existing difference-based and active shape model-based algorithms.

Test sequences include (i) a computer-generated fish sequence, (ii) a PETS 2002 test sequence, (iii) in-house indoor and (iv) outdoor sequences. All test sequences have the same resolution of  $320 \times 240$ . To speed up the simulation we use only gray-scale images. For obtaining in-house test sequences, we used a three-CCD colour video camera. The fish sequence has similar intensity distribution in both the object and background, while the shape of the fish keeps deforming. The PETS sequence has internal reflection on the glass window and scaled objects. The indoor sequence has external illumination and considerable amount of noise. The outdoor sequence has multiple objects with occlusion. Fig. 13 respectively summarize the experimental environment and test images.

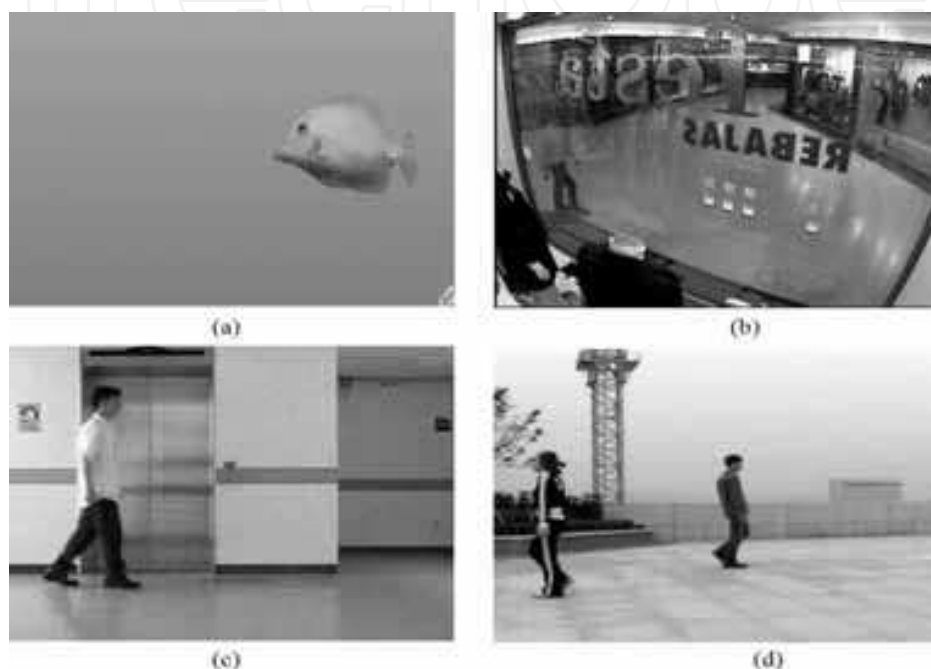


Fig. 13. The first image frames of four test sequences: (a) the fish, (b) the PETS 2002, (c) the indoor, and (d) the outdoor sequences

### 5.1 Combined shape and feature-based non-rigid object tracking

Fig. 12(a) has the worst case conditions for block matching, because texture of the object and background is similar and the object deforms. While existing block matching-based methods definitely fail in tracking the deformable fish in this sequence, the proposed algorithm provides reliable tracking results. Fig. 14 shows detection of object's moving region using background generation and the corresponding SCP extraction.

As shown in fig. 14(b) and (c), although there is no significant amount of noise in the input image, the initially detected object contains amplified noise during the subtraction process between the generated background and the input image. Such noise effect is removed by the proposed algorithm using a series of the morphological process, labelling, and SCP matching and updating.

Fig. 15 shows tracking results of the fish image using CBMA with 14 initial SCPs,  $k = 30$ ,  $5 \times 5$  blocks, and 65 frames. Most changes in the shape of fish occur especially at the tail part. In the dynamic tail part, we increase the number of SCPs, while keeping the same SCPs in the static region.

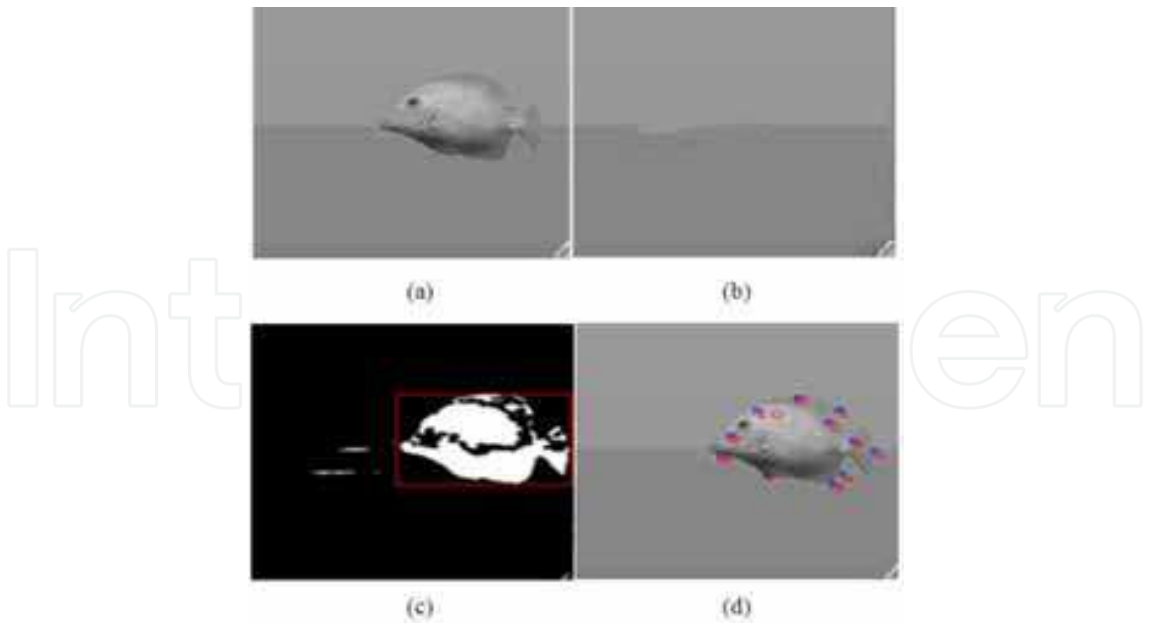


Fig. 14. Moving region detection and SCP extraction: (a) input image, (b) background generation, (c) region of object detection, and (d) SCPs superimposed on the input image

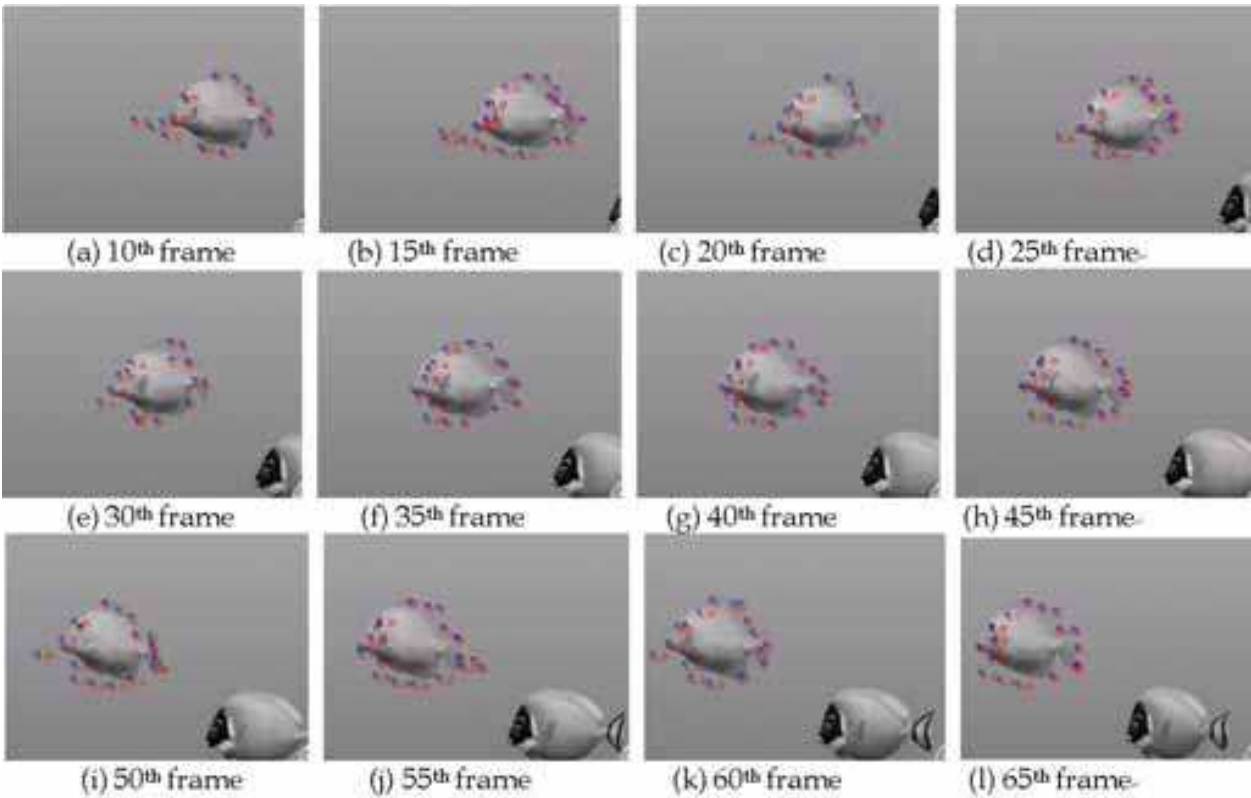


Fig. 15. Tracking results of the proposed shape and feature-based algorithm using CBMA

The block-matching method has weak from change of shape. But, because the proposed use to SCPs with some solved. Also Intensity trace in most of the block matching does not consider. Fig. 15 the result of the SCP in the background was of the location, object tracking,

so that the gradient search is available in strengths. Since the block matching has its weakness we propose the use of SCP. In addition, to improve overall use of SCP we integrated additional information in the form of tracking and gradient search method which is illustrated in fig. 15.

Fig. 16 shows tracking results of the fish image using PBMA with 11 initial SCPs,  $k = 30$ ,  $5 \times 5$  blocks, and 65 frames. We used 11 initial SCPs and  $k = 30$  for the BMA. PBMA updates SCPs at every 5 frames. CBMA shows recalculated SCPs, which represents replacement of SCPs.

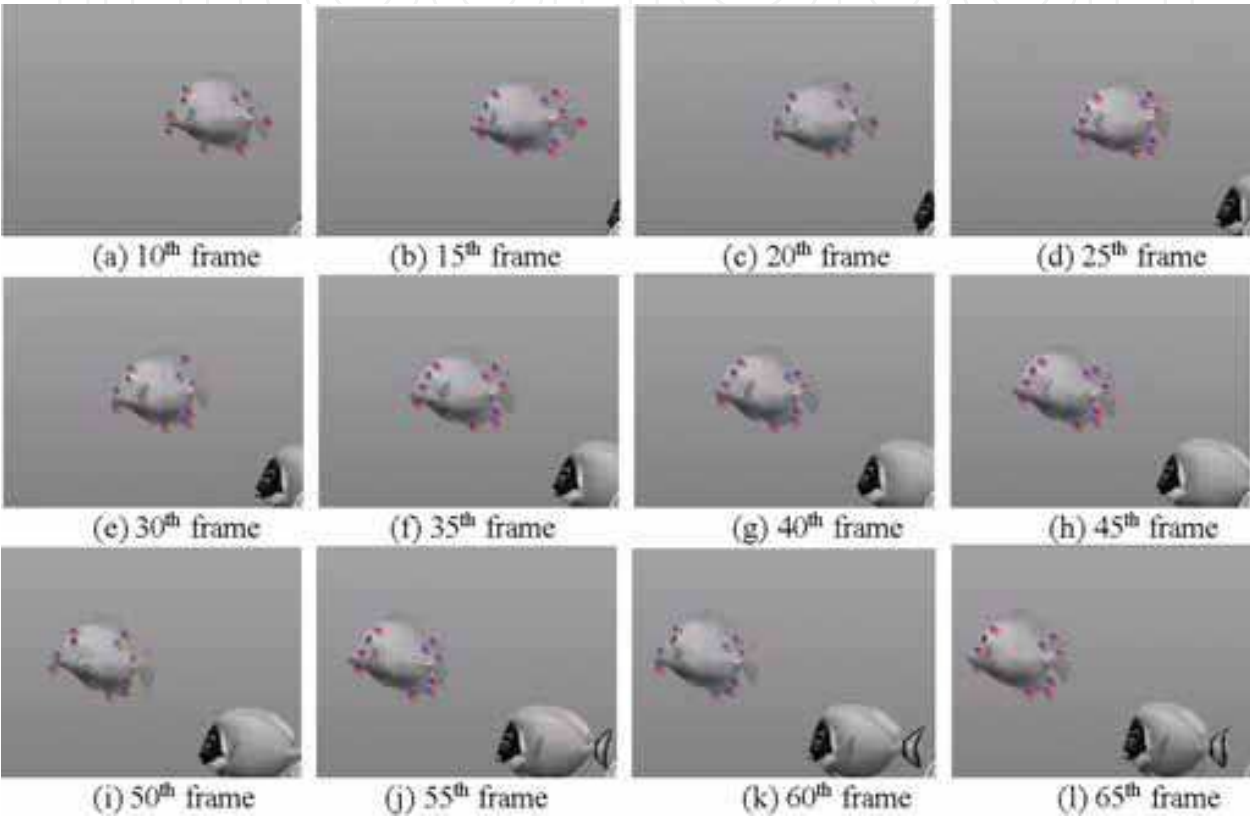


Fig. 16. Tracking results of the proposed shape and feature-based algorithm using PBMA

### 5.2 Performance analysis

We compare the tracking performance of the proposed method with the frame difference-based method and the ASM-based method. The location of the reference object is represented by the centroid of all pixels inside the manually specified object boundary. The accuracy of tracking is measured by the Euclidean distance between two points  $(x,y)$  and  $(\hat{x},\hat{y})$ , called the similarity measure defined as

$$1 / \rho = 1 / \sqrt{(\hat{x} - x)^2 + (\hat{y} - y)^2}, \tag{20}$$

where

$$x = \frac{1}{N} \sum_{i=0}^{n-1} x_i, y = \frac{1}{N} \sum_{i=0}^{n-1} y_i, \hat{x} = \frac{1}{S} \sum_{j=0}^{s-1} \hat{x}_j, \hat{y} = \frac{1}{S} \sum_{j=0}^{s-1} \hat{y}_j. \quad (21)$$

where  $(x_i, y_i)$ ,  $i = 0, 1, \dots, N-1$ , represent pixels inside the manually specified object boundary, and  $(\hat{x}_j, \hat{y}_j)$ ,  $j = 0, 1, \dots, S-1$ , the set of SCP's obtained by a tracking method to be compared. We can decide that a tracking algorithm is accurate if  $\rho$  is sufficiently small. Fig. 17 shows comparative results of the existing and the proposed tracking methods in the sense of the similarity measure.

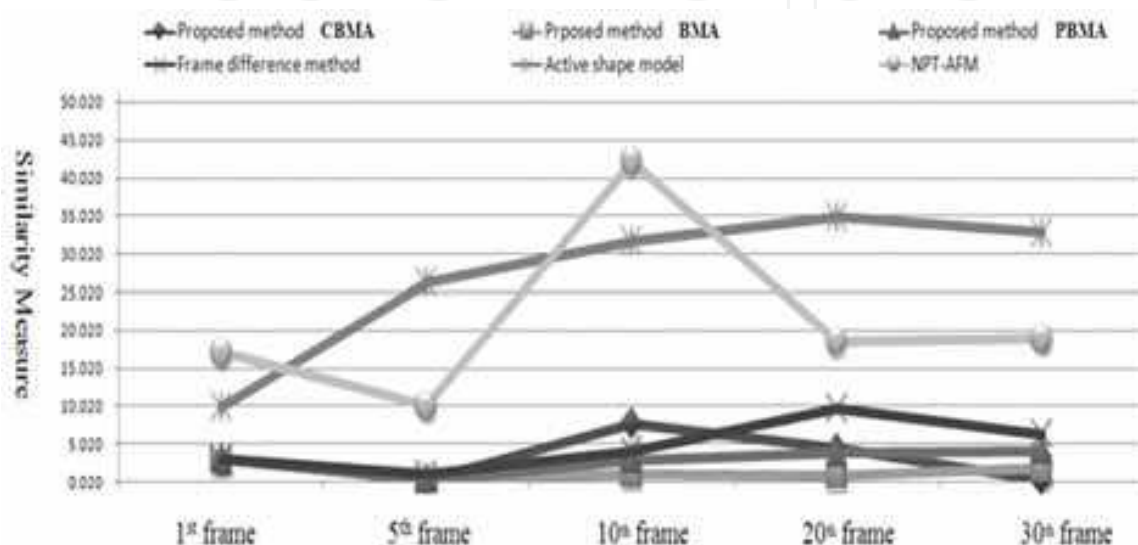


Fig. 17. Similarity curves: Comparison of various tracking methods using the fish sequence.

As shown in Fig. 17, all methods except the ASM-based method provide almost identical center points that coincides the reference center. As the number of frames increases, CBMA keeps increasing the similarity measure, while ASM and NPT-AFM methods become worse. It is to be noted that the simple difference-based method performs well in the starting frames, but the proposed method keeps increasing the similarity measure. We also note that ASM and NPT-AFM are both very sensitive to the initial training shape and boundary noise. Fig. 18 illustrates the comparison of tracking results between the proposed (CBMA and PBMA) with those of ASM and NPT-AFM. In case of ASM-based tracking lack of initialization in the starting frames led to poor convergence and increased error in the model fitting stages. This problem can be overcome using NPT-AFM-based method which yields higher feature detection and fitting accuracy. However the efficiency of NPT-AFM depends solely on the segmentation procedure which might lead to feature point's deviation from the object shape near the boundary. Both of the above mentioned drawbacks were overcome by using the proposed method because of the inclusion of SCP-based block matching and detection processes.

First column shows tracking results using ASM with 40 detected points in gray scale, Second column shows results of NPT-AFM algorithm in Y channel of YCrCb image format. Third column represents proposed CBMA and PBMA approach in gray scale.








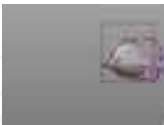
















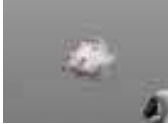
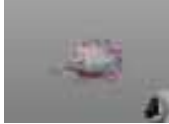








Fish sequence	ASM	NPT-AFM	Proposed method
	 Frame #1, Detected Point: 40	 Frame #1, Detected Point: 19	 Frame #1, Detected Point: 24
	 Frame #5, Detected Point: 40	 Frame #5, Detected Point: 19	 Frame #5, Detected Point: 25
	 Frame #10, Detected Point: 40	 Frame #10, Detected Point: 18	 Frame #10, Detected Point: 14
	 Frame #15, Detected Point: 40	 Frame #15, Detected Point: 19	 Frame #15, Detected Point: 15
	 Frame #25, Detected Point: 40	 Frame #25, Detected Point: 17	 Frame #25, Detected Point: 16
	 Frame #30, Detected Point: 40	 Frame #30, Detected Point: 18	 Frame #30, Detected Point: 18
	 Frame #35, Detected Point: 40	 Frame #35, Detected Point: 20	 Frame #35, Detected Point: 16
	 Frame #40, Detected Point: 40	 Frame #40, Detected Point: 19	 Frame #40, Detected Point: 15
(a)	(b)	(c)	(d)

Fig. 18. Illustration of tracking results using (a) input sequence, (b) ASM, (c) NPT-AFM, and (d) the proposed method



As shown in fig. 18, although NPT-AFM accurately tracks control points in the head region, it fails to track in the tail region because of concave shape in the tail region does not satisfy NPT-AFM’s assumption. On the other hand the proposed method keeps tracking all control points evenly. In conclusion, we can say that the proposed tracking methods outperform others in the sense of stability, robustness, and the least number of control points used.

Fig.19-21 shows the results of the proposed methods with different conditions of input images. Fig. 19 shows the result without reflection of illumination on change. On the other hand, fig. 20 and fig. 21 respectively show the results with reflection and illumination change. Based on the results shown in fig. 19-21, the proposed method is robust to various conditions such as reflection and illumination change.

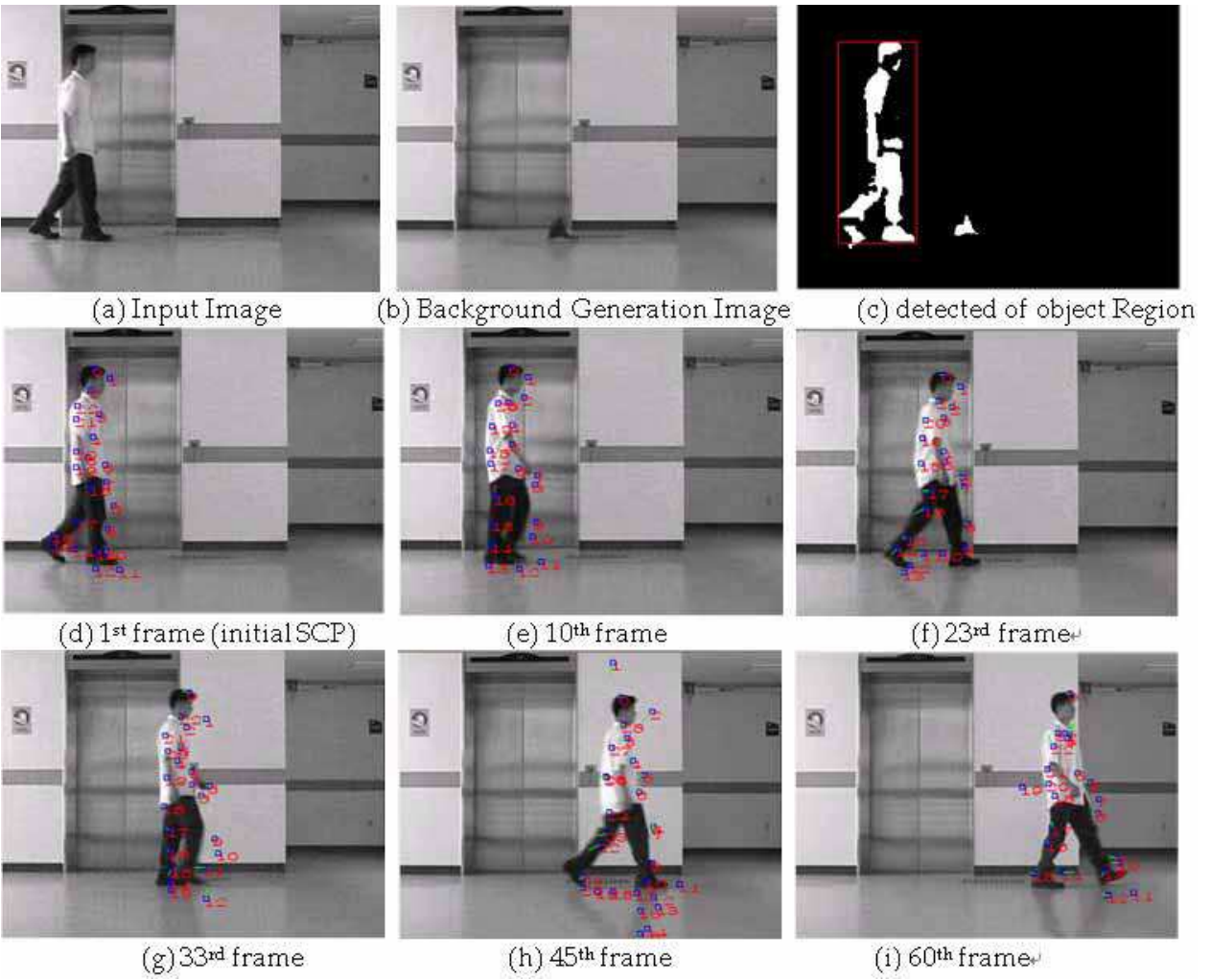


Fig. 19. Tracking results of the proposed method without any critical conditions. The initial background is generated using 50 frames. 5×5 blocks, 23 SCPs, and threshold of 25 are used.

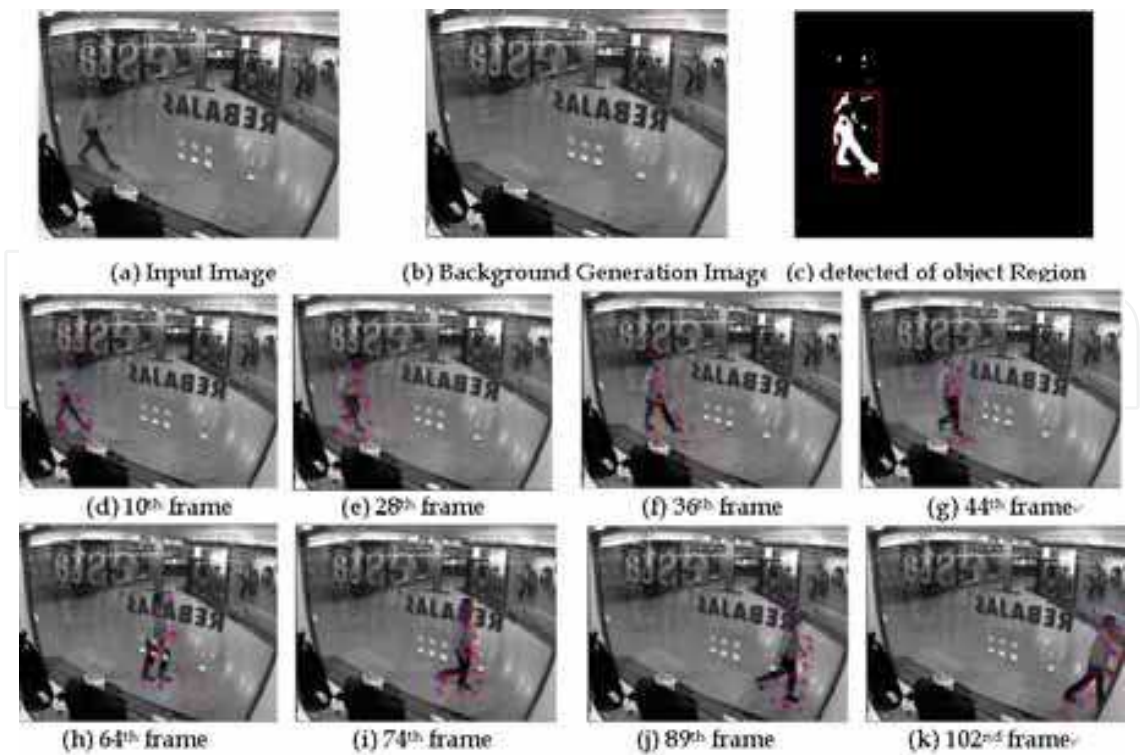


Fig. 20. Tracking results of the proposed method with strong reflections in the input image. The initial background is generated using 50 frames. 5×5 blocks, 12 SCPs, and threshold of 20 are used.

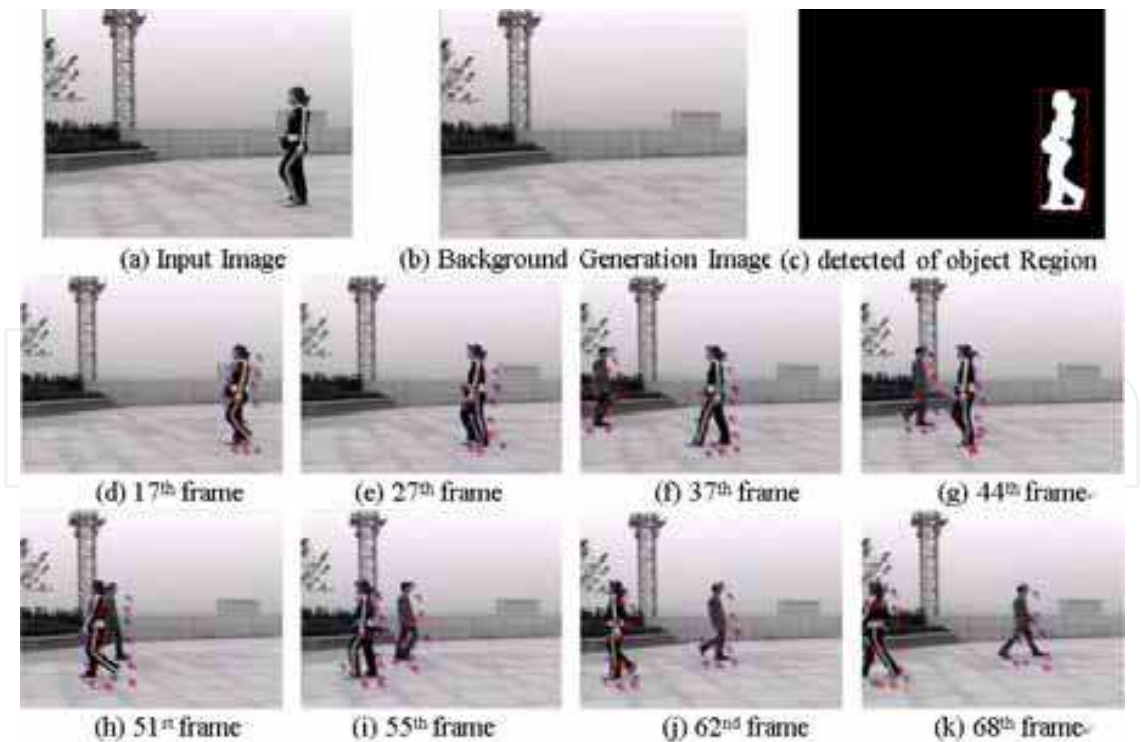


Fig. 21. Tracking results of the proposed method with illumination change in the input image. The initial background is generated using 50 frames. 5×5 blocks, 26 SCPs for group A, 12 SCPs for group B, and threshold of 25 are used.

The proposed method performs better than existing methods while maintaining the shape of an object is tracked. Therefore, the initial ASM generated shows good performance feature points to renew the NPT-AFM method does not use ASM initial feature points unstructured object to trace the missing feature points reconfiguration in test image(a) results were not good because it does not.

## 6. Conclusion

In this chapter we presented ASM, NPT-AFM, and combined shape and feature-based object tracking methods. The combined shape and feature-based method adaptively generates background, which serves as a fundamental building block for robust tracking by resolving inherent problems of existing BMA. After generating background the shape tracking module in the proposed algorithm determines object's moving region based on SCPs. Another contribution of this chapter is the CBMA method, which enables robust tracking with occlusion.

Extensive experiments have been performed using (i) computer generated fish sequence, (ii) PETS 2002 test sequence, (iii) in-house indoor sequence, and (iv) in-house outdoor sequence. Experimental results prove that the proposed method can provide robust tracking with complicated environment such as multiple objects, occlusion, and complicated background.

## 7. Acknowledgments

This research was supported by the Ministry of Knowledge Economy MKE of Korea, under the HNRC-ITRC support program supervised by the National IT Industry Promotion Agency (NIPA-2009-C1090-0902-0035) and by Basic Science Research Programs through National Research Foundation (NRF) of Korea funded by the Ministry of Education, Science and Technology (2009-0081059, 2009-0069382)

## 8. Reference

- Mckenna, S, J; Raja, Y. And Gong, S. (1999) Tracking colour objects using adaptive mixture models. *Image, vision Computing*, Vol. 17, pp. 225-231, March 1999.
- Plankers, R. and Fuz, P. (2001) Tracking and modeling people in video sequences. *Computer Vision, Image Understanding. Computer Vision, Image Understanding*, Vol. 81, pp. 285-302, March 2001.
- Cootes, T, F.; Taylor, C, T., and Graham, J (1992) Training models of shape from sets of examples. *Proc. Int. Conf. Machine Vision*, pp. 9-18, 1992.
- Lee, S, W., Kang, J, Shin, J, and Paik, J (2007) Hierarchical active shape model with motion prediction for real-time tracking of non-rigid objects. *IET Computer Vision*, Vol. 1, pp. 17-24, March 2007.
- Shin, J, Kim, S., Kang, S., Lee, S., and Paik, J (2005) Optical flow-based real-time object tracking using non-prior training active feature model. *Real-Time Imaging*, Vol. 11, pp. 204-218, June 2005.

- Zhang, X., and Minai, A. (2004) Temporally sequenced intelligent block-matching and motion-segmentation using locally coupled networks. *IEEE Trans. Neural Networks*, Vol. 15, pp. 1202-1214, September 2004.
- Li, L., Huang, W., Gu, I., and Qi, T. (2002) Foreground object detection in changing background based on color co-occurrence statistics. *Proc. Int. Conf. Applications, Computer Vision*, pp. 269-274, 2002.
- Chien, S., Ma, S., and Chen, L. (2002) Efficient moving object segmentation algorithm using background registration technique. *IEEE Trans. Circuits, Systems, Video Technology*, Vol. 12, pp. 577-586, July 2002.
- Nascimento, J., and Marques, J (2004) Robust shape tracking in the presence of cluttered background. *IEEE Trans. Multimedia*, Vol. 6, pp. 852-861, December 2004.
- Javed, O., Shafique, K., and Shah, M. (2004) A hierarchical approach to robust background subtraction using color and gradient information. *Proc. Int. Conf. Motion, Video Computing*, pp. 22-27, December 2002.
- Calvagno, G., Fantozzi, F., Rinaldo, R., and Viareggio, A. (2004) Model-based global and local motion estimation for videoconference sequences. *IEEE Trans. Circuits, Systems, Video Technology*, Vol. 14, pp. 1156-1161, September 2004.
- Koschan, A., Kang, S., Paik, J., Abidi, B., and Abidi, M. (2003) Color active shape models for tracking non-rigid objects. *Pattern Recognition Letters*, Vol. 24, pp. 1751-1765, July 2003.
- Zhang, D., and Lu, G. (2000) An edge and color oriented optical flow estimation using block matching. *Proc. Int. Conf. Signal Processing*, Vol. 2, pp. 1026-1032, August 2000.
- Stefano, L., and Viarani, E. (1999) Vehicle detection and tracking using the block matching algorithm. *Proc. Int. Conf. Circuits, Systems, Communications, Compute*, Vol. 1, pp. 4491-4496, 1999.
- Hariharakrishnan, K., and Schonfeld, D. (2005) Fast object tracking using adaptive block matching. *IEEE Trans. Multimedia*, Vol. 7, pp. 853-859, October 2005.
- Tanimoto, S., and Pavlidis, T. (1975) A hierarchical data structure for picture processing. *Computer Graphics, Image Processing*, Vol. 4, pp. 104-119, June 1975.
- Tekalp, A, M. (1995) Digital Video Processing. Prentice-Hall, 1995.
- Isard, M., and Blake, A. (1996) Condensation-conditional density propagation for visual tracking. *Int. Jour. Computer Vision*, pp. 1-30, 1996.
- Bouguet, J (2000) Pyramidal implementation of the Lucas Kanade feature tracker description of the algorithms. *OpenCV Documentation, MicroProcessor Research Labs, Intel Corporation*, 2000.
- Goodall, C. (1991) Procrustes methods in the statistical analysis of shape. *Jur. The Royal Statistical Society, Part B*, vol. 53, pp. 285-339, 1991.
- Haritaoglu, I., Harwood, D., and Davis, L. (2000) W4: real-time surveillance of people and their activities. *IEEE Trans. Pattern Analysis, Machine Intelligence*, Vol. 22, pp. 809-830, August 2000.

Kim, T., Paik, D., and Paik, J (2007) Block Matching-Based Background Generation and Non-Rigid Shape Tracking for Video Surveillance. *Proc. Int. Conf. ICWAPR*, vol. 1, PP. 415-420, November 2007.

IntechOpen

IntechOpen





## **Video Surveillance**

Edited by Prof. Weiyao Lin

ISBN 978-953-307-436-8

Hard cover, 486 pages

**Publisher** InTech

**Published online** 03, February, 2011

**Published in print edition** February, 2011

This book presents the latest achievements and developments in the field of video surveillance. The chapters selected for this book comprise a cross-section of topics that reflect a variety of perspectives and disciplinary backgrounds. Besides the introduction of new achievements in video surveillance, this book also presents some good overviews of the state-of-the-art technologies as well as some interesting advanced topics related to video surveillance. Summing up the wide range of issues presented in the book, it can be addressed to a quite broad audience, including both academic researchers and practitioners in halls of industries interested in scheduling theory and its applications. I believe this book can provide a clear picture of the current research status in the area of video surveillance and can also encourage the development of new achievements in this field.

### **How to reference**

In order to correctly reference this scholarly work, feel free to copy and paste the following:

Taekyung Kim and Joonki Paik (2011). Block Matching-Based Background Generation and Non-Rigid Shape Tracking for Video Surveillance, Video Surveillance, Prof. Weiyao Lin (Ed.), ISBN: 978-953-307-436-8, InTech, Available from: <http://www.intechopen.com/books/video-surveillance/block-matching-based-background-generation-and-non-rigid-shape-tracking-for-video-surveillance>

**INTECH**  
open science | open minds

### **InTech Europe**

University Campus STeP Ri  
Slavka Krautzeka 83/A  
51000 Rijeka, Croatia  
Phone: +385 (51) 770 447  
Fax: +385 (51) 686 166  
[www.intechopen.com](http://www.intechopen.com)

### **InTech China**

Unit 405, Office Block, Hotel Equatorial Shanghai  
No.65, Yan An Road (West), Shanghai, 200040, China  
中国上海市延安西路65号上海国际贵都大饭店办公楼405单元  
Phone: +86-21-62489820  
Fax: +86-21-62489821



© 2011 The Author(s). Licensee IntechOpen. This chapter is distributed under the terms of the [Creative Commons Attribution-NonCommercial-ShareAlike-3.0 License](https://creativecommons.org/licenses/by-nc-sa/3.0/), which permits use, distribution and reproduction for non-commercial purposes, provided the original is properly cited and derivative works building on this content are distributed under the same license.

IntechOpen

IntechOpen