Available online at www.sciencedirect.com

ScienceDirect

J. Vis. Commun. Image R. 17 (2006) 1190–1208

JOURNAL OF VISUAL Communication & IMAGE Representation

www.elsevier.com/locate/jvci

Video object tracking using adaptive Kalman filter

Shiuh-Ku Weng [a,*], Chung-Ming Kuo [b], Shu-Kang Tu [b]

[a] *Department of Information Management, Chinese Naval Academy, Zuoying District, Kaohsuing City 813, Taiwan, ROC*
[b] *Department of Information Engineering, I-Shou University, Kaohsiung County 840, Taiwan, ROC*

Received 3 September 2005; accepted 23 March 2006
Available online 6 June 2006

Abstract

In this paper, a new video moving object tracking method is proposed. In initialization, a moving object selected by the user is segmented and the dominant color is extracted from the segmented target. In tracking step, a motion model is constructed to set the system model of adaptive Kalman filter firstly. Then, the dominant color of the moving object in HSI color space will be used as feature to detect the moving object in the consecutive video frames. The detected result is fed back as the measurement of adaptive Kalman filter and the estimate parameters of adaptive Kalman filter are adjusted by occlusion ratio adaptively. The proposed method has the robust ability to track the moving object in the consecutive frames under some kinds of real-world complex situations such as the moving object disappearing totally or partially due to occlusion by other ones, fast moving object, changing lighting, changing the direction and orientation of the moving object, and changing the velocity of moving object suddenly. The proposed method is an efficient video object tracking algorithm.

Keywords: HSI color space; Adaptive Kalman filter; Occlusion ratio

1. Introduction

The researches for segmenting, estimating, and tracking semantic objects in video have received great attention for the last few years [1–26]. The moving object tracking is an important issue in video system, such as surveillance, sports reporting, video annotation, and traffic management system. However, if the background and moving objects vary dynamically, such situations will be complicated. In video analysis, we have to know the features of moving objects, such as color, texture, and shape, etc., so that the moving object can be detected and be tracked. There are some situations for video in a real world environment, including camera lens is fixed or not, multiple moving objects, rigid object, or non-rigid object, occlusion by the other objects, one or many cameras, full automatic or semi-automatic semantic object tracking, etc. According to the above discussion, to conclude, we will encounter the following three problems for tracking moving object.

* Corresponding author. Fax: +886 7 5829682.
E-mail address: skweng@mail.cna.edu.tw (S.-K. Weng).

1047-3203/$ - see front matter
doi:10.1016/j.jvcir.2006.03.004

- Initial moving object segmenting problem.
- Detection of moving object.
- Tracking the moving target in occlusion.

In the initial moving object segmenting problem, the methods of [5,9,14] use the user's assistance to appoint a moving object as target. This is called semi-automatic system. Or, in the environment of still background, [3,8] apply the difference information of two consecutive frames to extract the moving objects. For the sake of simplicity, to solve this problem, most of methods assume that they are in the environment of still background in the initial state. Although the semi-automatic system simplifies the initial moving object problem, its applications are still popular in real systems.

About the detection of moving object, most of techniques base on the features of color, texture, shape, edge, motion, and shape. Generally, the color feature is frequently used [9,14,17,20], since the human perception is sensitive to the color. However, the disadvantage of the single color feature for moving object detection is that it can only be used in the target object with uniformly distributed color. In addition, Jang and Choi [2] propose an active model to detect and track the moving object. However, the method [2] with the Kalman filter technique which predicts the detecting range to reduce the computational complexity can not be applied to solve the target in occlusion. Therefore, for tracking the moving target in occlusion, Jang and Choi in paper [1] propose the structural Kalman filter to estimate the motion information under a deteriorating condition as occlusion. The structural Kalman filter is a composite of two types of the Kalman filters: cell Kalman filters and relation Kalman filters. Their method partitions an object into several sub-regions and utilizes the relational information among sub-regions of a moving object. The cell Kalman filter estimates motion information of each sub-region of a target, and the relation Kalman filter is to estimate the relative relationship between two adjacent sub-regions. When a sub-region is judged not to be occluded, the cell Kalman filter of the sub-region is enough to estimate its motion information. If a sub-region is judged to be occluded, the relation Kalman filters of the adjacent sub-region are used to compensate the corrupted estimate of the cell Kalman filter. The compensation weighting is set by the degree of occlusion. The idea of the structural Kalman filter method is good, however, its applications are limited since the structural Kalman filter is very complicated and it is not easy to select the criteria or features to partition an object into sub-regions in the different real-world tracking application. In addition, the degree of occlusion needs the other model to judge. More importantly, a complex system is difficult to be expanded to keep tracking multiple objects.

Additionally, many works have been done by motion model based human recognition and estimating human body postures [21–25]. In general, the above methods offer resistance for the tracker to cope with partial occlusions, changing light condition, and object pose. However, these methods only adapt to human tracking, and more importantly, their computation for model estimation is expensive and the number of model parameters is usually large. And, these methods handle partial occlusions only, failing for severe or complete occlusions. Furthermore, according to the suggestions of the paper [17], for a large range of tracking applications where the object motion is rigid or the object is sufficiently distant from the camera, the tracking system will not need to separate model for the object shape since it can be determined from the object motion. And, tracking on object appearance rather than geometry is easier due to better identification power of appearance features.

In the paper [17], H.T. Nguyen and A.W.M. Smeulders think that the tracking algorithm needs to satisfy the two qualities: simplicity and robustness. Simplicity implies that the algorithm is easy to implement and has the minimum number parameters. Robustness implies the ability of the algorithm to track objects under different conditions which include: severe occlusions, lighting changes, object orientation changing, background clutter, a moving camera, zooms, and rotations. Therefore, in this paper, a simple and efficient method for tracking object using adaptive Kalman filter model is developed and its the robustness quality can be applied to a real-world environment. In the following section, the proposed method is shown. The experimental results are demonstrated in Section 3. Finally, the Section 4 brings to the conclusion and discussion.

2. The proposed method

In this paper, the initial moving object is selected by user. Because there may exist more than one moving objects in video, therefore the tracking target should be determined according to the user interested. After the selection, we initially segment the object in frame t by the difference of three consecutive frames, $t-1$, t, $t+1$, and then apply the region growing algorithm to segment the desired object. Afterwards, the dominant color feature of moving target is extracted from the RGB color space by K-means algorithm. After the initial moving object segmentation and feature extraction, to detect the moving object in consecutive frames, the HSI (hue, saturation, and intensity) color model is used to build up the resistance to lighting changes, since its three components are relatively independently.

In addition, we proposed an adaptive Kalman filtering algorithm to effectively track the moving object. The ratio of the moving object area in frame t to that in frame $t-1$ is used to as the object occlusion ratio. The occlusion ratio will be used to adaptively adjust the estimate parameters of the Kalman filter. Fig. 1 demonstrates the system structure. The main steps of the proposed tracking algorithm are listed as follows:

■ Initialization
1. Moving object segmentation by frame difference and region growing.
2. Object feature extraction.

■ Tracking by adaptive Kalman filter
1. Motion model construction to build the system state model of adaptive Kalman filter.
2. Moving object detection in consecutive frames for the correction step of adaptive Kalman filter.

2.1. Initial moving object segmentation and feature extraction

The purpose of initial moving object segmentation is to segment the user's interested target and extract the feature of the selected target. The traditional frame difference and region growing methods are applied to segment the moving object.

2.1.1. Frame difference and region growing

Frame difference is a simple method to segment the moving object in image video. After the user appoints a moving object as target, the target is segmented by the differences of frames in $t-1$, t, and $t+1$. It uses the difference of consecutive frames to detect the change area of frames. Since the user has assigned a moving object as target, therefore, for reducing the complexity, the frame difference will be operated within the range with center at the user' s selector locating point and radius 50 points rather than within all frame. The differential frame can be defined as

$$\mathrm{FD}(x,y,t) = \begin{cases} 0 & \text{if } |f(x,y,t+1) - f(x,y,t)| \leqslant \text{ threshold}, \\ 1 & \text{otherwise}, \end{cases} \tag{1}$$

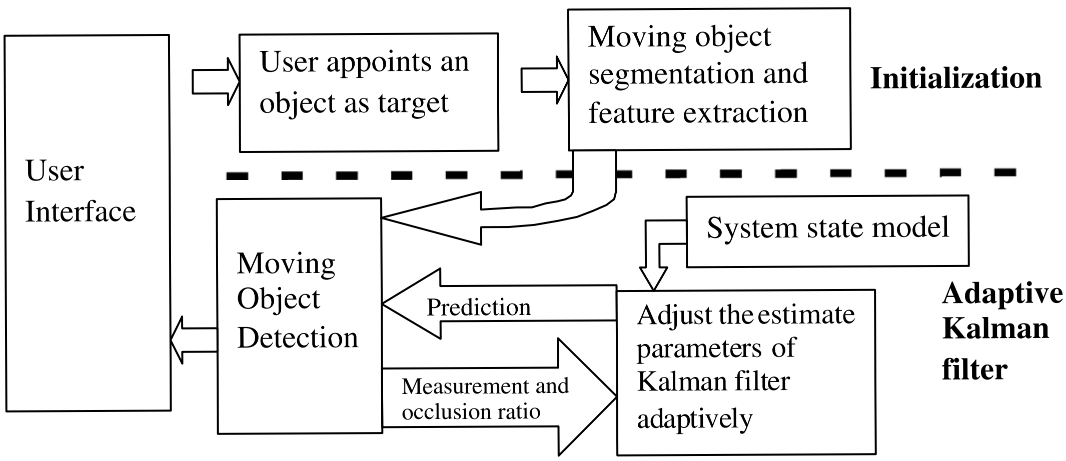

Fig. 1. The system structure of the proposed method.

where $f(x, y, t)$ and $f(x, y, t+1)$ are the pixels at position (x, y) in frame t and $t+1$. The frame is in RGB color space and the pixels in frame t are defined as $f(x, y, t) = \{R_{x,y,t}, G_{x,y,t}, B_{x,y,t}\}$. If the absolute of the difference is smaller than the threshold, it is the background and $\mathrm{FD}(x, y, t) = 0$. Otherwise, it is the target object and $\mathrm{FD}(x, y, t) = 1$.

To improve the segmentation accuracy, three consecutive frames are used to segment the moving object. The segmented moving object is obtained through the intersect of the two differences, $\mathrm{FD}(x, y, t-1)$ and $\mathrm{FD}(x, y, t)$, which are the differences between three consecutive frames. Therefore, the points of segmented moving object is defined as

$$\mathrm{MR}(x, y, t) = \mathrm{FD}(x, y, t-1) \cap \mathrm{FD}(x, y, t). \tag{2}$$

The set of segmented moving object area is described as

$$\mathrm{MRS} = \{(x, y) | \mathrm{MR}(x, y, t) = 1\}. \tag{3}$$

Fig. 2 demonstrates an example using the frame difference method to segment the moving object. The complete detected moving object is extracted from the original image and is shown in Fig. 2d.

Since, the frame difference method probably yields several segmented moving objects in the operated range, for constructing a connected object region, the region growing method is used at the segmented moving object which the user appoints. The region growing method consists of two basic morphological methods. In the first one, it is defined by the following iterative expression and it yields a connected object region [28].

$$\mathrm{MO}_k = (\mathrm{MO}_{k-1} \oplus \mathrm{SE}) \cap \mathrm{MRS}, \qquad k = 1, 2, 3, \ldots, \tag{4}$$

where MO_0 is the initial point which the user's selector locates, MO_k is the region set of the kth iteration, $\oplus$ is the dilation operator, SE is the structuring element, MRS is defined in Eq. (3). Fig. 3 shows an example of

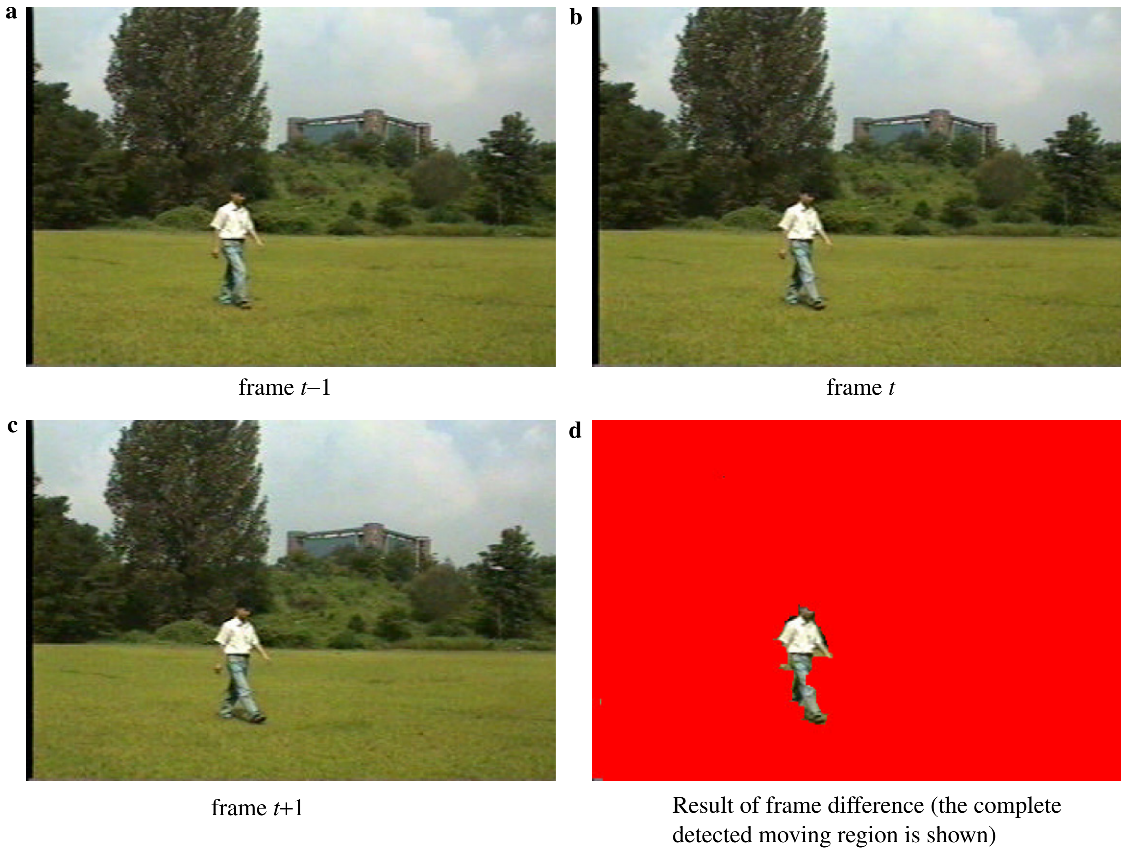

Fig. 2. An example of frame difference.

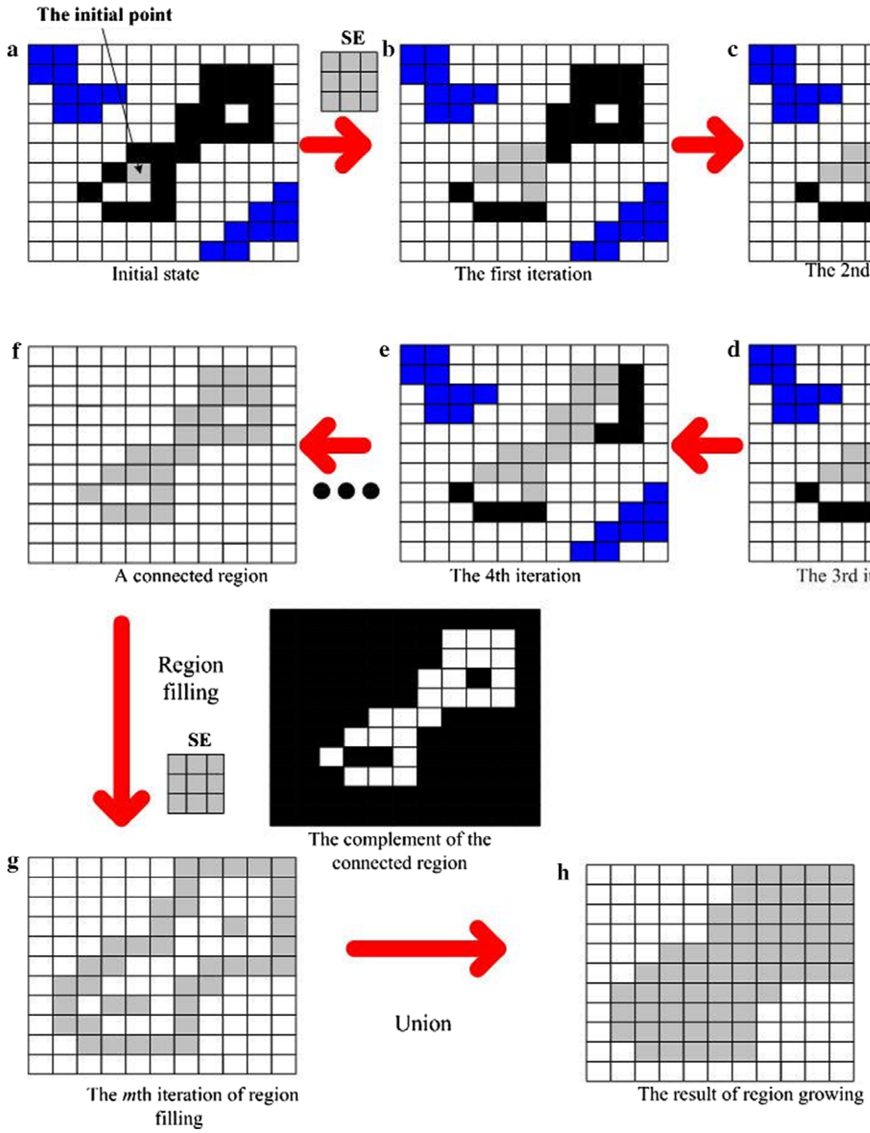

Fig. 3. An example of region growing method.

region growing method. In Fig. 3a, the black and blue points are the object and it is assumed that the region growing starts from the grey point which the user's selector locates. If the initial point is not located in one of the segmented regions, we select the segmented region that is closest to the user's selector locating point. The 3×3 block is used as the structuring element to group the object points. From Figs. 3b–e, they are the 1st, 2nd, 3rd, and 4th iteration, respectively. The iterations do not terminate until the connected region is extracted, i.e., $\mathrm{MO}_n = \mathrm{MO}_{n-1}$ at nth iteration, and then the other segmented objects will be given up. Once a connected segmented region is found, another morphological method called region filling [28] is applied to dilate the MO_n. The region filling is expressed as

$$\mathrm{OR}_k = (\mathrm{OR}_{k-1} \oplus \mathrm{SE}) \cap \mathrm{MO}_n^C, \qquad k = 1, 2, 3, \ldots, \tag{4-1}$$

where $\mathrm{OR}_0 = \mathrm{MO}_0$ is the initial point and OR_k is the result of the kth iteration. The process terminates at mth iteration if $\mathrm{OR}_m = \mathrm{OR}_{m-1}$. The union set OR, $\mathrm{OR} = \mathrm{OR}_m \cup \mathrm{MO}_n$, is a filled region. The final region growing result shows in Fig. 3h. When the object region is defined, the dominant color of the moving object can be extracted by the method mentioned in the following sub section.

2.1.2. Dominant color feature extraction

To find the dominant color, we process feature extraction in the RGB color space. The RGB color space is represented by a 3-dimensional cube with red, green, and blue additive primaries. When color clustering by K-means method operates in RGB space, it can directly use identical weighting in every dimension to calculate, since the RGB color space forms a cube and the clustering does not need to consider the special property of every dimension. For example, the cyclic property of the hue component in HSI color space needs to be considered. Therefore, the RGB color space is very suitable for dominant color extraction.

First, the moving object colors in RGB components are classified by K-means [27]. The K is experimentally set as 5. The color pixels of moving object are classified into different K groups by color and the average color of the every group in frame t is defined as $\mathrm{MC}_k(t)$. If the group d is the highest density (minimum distance summation) one, the $\mathrm{MC}_d(t)$ is set as the dominant color feature. The dominant color of the moving object in frame t is defined as

$$\mathrm{DomC}(t) = \left\{ \mathrm{MC}_d(t) \middle| \min\left(\frac{\sum_{(x,y)\in C_k} \sqrt{(x-\bar{x}_{C_k})^2 + (y-\bar{y}_{C_k})^2}}{N_{C_k}},\ k=1,\ldots,5 \right) \right\}, \tag{5}$$

where $C_k = \{(x_i, y_i, f(x_i, y_i, t)) | f(x_i, y_i, t) = \{R_{x_i,y_i,t}, G_{x_i,y_i,t}, B_{x_i,y_i,t}\}, i = 1, \ldots, N_{C_k}\}$ is the set of kth group, $\mathrm{MC}_d(t)$ is the average color of the highest density group, N_{C_k} is the number of group set C_k and $(\bar{x}_{C_k}, \bar{y}_{C_k})$ is the position of mass center in kth group set. The structure for initial segment of moving object and feature extraction is shown in Fig. 4. The dominant color will be used as the feature for detecting the moving object in the following section.

2.2. Tracking the moving object by the proposed adaptive Kalman filter

For the tracking problem, if we have to estimate the motion information of object under such a deteriorating condition as occlusion, a simple detecting method will not be able to measure and track the moving object

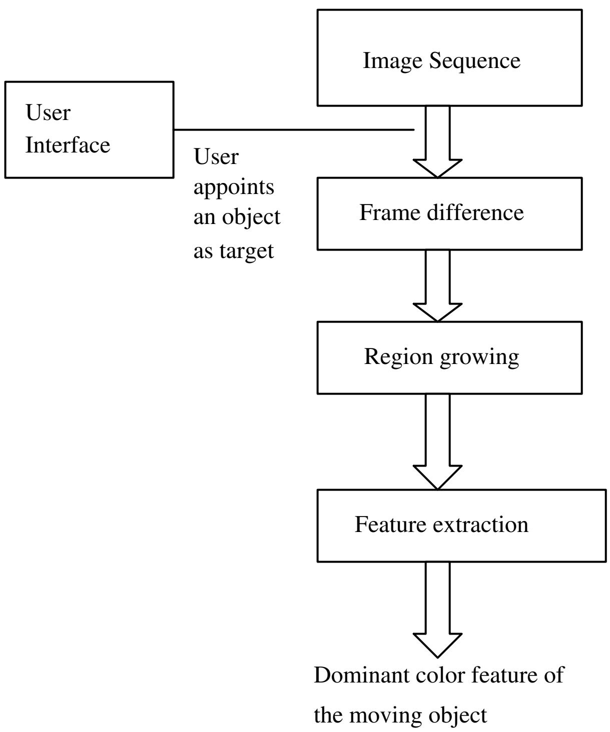

Fig. 4. The structure for initial moving object segmentation and feature extraction.

correctly. In the paper, we propose an adaptive Kalman filter to implement. In the proposed adaptive Kalman filter, a moving model is constructed to set the system state model and the result of the moving object detection in consecutive frames is fed back as the measurement for correction. In addition, the estimate parameters of Kalman filter are adjusted adaptively.

2.2.1. The typical Kalman filter

The Kalman filter has been used in engineering application successfully. The Kalman filter has two distinctive features. One is that its mathematical model is described in terms of *state-space* concepts. The other is that its solution is computed recursively. Usually, the Kalman filter is described by system state model and measurement model. The state-space model is described as

$$\text{System state model}: \quad \mathbf{s}(t) = \ddot{\mathbf{O}}(t-1)\mathbf{s}(t-1) + \mathbf{w}(t), \tag{6}$$

and

$$\text{Measurement model}: \quad \mathbf{z}(t) = \mathbf{H}(t)\mathbf{s}(t) + \mathbf{v}(t), \tag{7}$$

where $\ddot{\mathrm{O}}(t-1)$ and $\mathbf{H}(t)$ are the state transition matrix and measurement matrix, respectively. The $\mathbf{w}(t)$ and $\mathbf{v}(t)$ are white Gaussian noise with zero mean and

$$\begin{aligned} &E\{\mathbf{w}(k)\mathbf{w}^{\mathrm{T}}(l)\} = \mathbf{Q}\delta_{kl}, \\ &E\{\mathbf{v}(k)\mathbf{v}^{\mathrm{T}}(l)\} = \mathbf{R}\delta_{kl}, \end{aligned} \tag{8}$$

where δ_{kl} denotes the Kronecker delta function; $\mathbf{Q}$ and $\mathbf{R}$ are covariance matrices of $\mathbf{w}(t)$ and $\mathbf{v}(t)$, respectively. The state vector $\mathbf{s}(t)$ of the current time t is predicted from the previous estimate and the new measurement $\mathbf{z}(t)$.

The tasks of the Kalman filter have two phases: prediction step and correction step. The prediction step is responsible for projecting forward the current state, obtaining a priori estimate of the state $\mathbf{s}^{-}(t)$. The task of correction step is for the feedback. That is to say, it incorporates an actual measurement into the a priori estimate to obtain an improved a posteriori estimate $\mathbf{s}^{+}(t)$. The $\mathbf{s}^{+}(t)$ is written as

$$\mathbf{s}^{+}(t) = \mathbf{s}^{-}(t) + \mathbf{K}(t)(\mathbf{z}(t) - \mathbf{H}(t)\mathbf{s}^{-}(t)), \tag{9}$$

where $\mathbf{K}(t)$ is the weighting and is described as

$$\begin{aligned} \mathbf{K}(t) &= \mathbf{P}(t)^{-}\mathbf{H}(t)^{\mathrm{T}}(\mathbf{H}(t)\mathbf{P}^{-}(t)\mathbf{H}(t)^{\mathrm{T}} + \mathbf{R}(t))^{-1} \\ &= \frac{\mathbf{P}^{-}(t)\mathbf{H}(t)^{\mathrm{T}}}{\mathbf{H}(t)\mathbf{P}^{-}(t)\mathbf{H}(t)^{\mathrm{T}} + \mathbf{R}(t)}. \end{aligned} \tag{10}$$

In Eq. (10), the $\mathbf{P}^{-}(t)$ is the priori estimate error covariance. It is defined as

$$\mathbf{P}^{-}(t) = E\lfloor \mathbf{e}^{-}(t)\mathbf{e}^{-}(t)^{\mathrm{T}} \rfloor, \tag{11}$$

where $\mathbf{e}^{-}(t) = \mathbf{s}(t) - \mathbf{s}^{-}(t)$ is the priori estimate error. In addition, the posteriori estimate error covariance $\mathbf{P}^{+}(\mathrm{t})$ is defined as

$$\mathbf{P}^{+}(t) = E\lfloor \mathbf{e}^{+}(t)\mathbf{e}^{+}(t)^{\mathrm{T}} \rfloor, \tag{12}$$

where $\mathbf{e}^{+}(t) \equiv \mathbf{s}(t) - \mathbf{s}^{+}(t)$ is the posteriori estimate error. The prediction step and correction step are executed recursively in the following definition.

■ Prediction step

$$\mathbf{s}^{-}(t) = \ddot{\mathbf{O}}(t-1)\mathbf{s}^{+}(t-1). \tag{13}$$

$$\mathbf{P}^{-}(t) = \ddot{\mathbf{O}}(t-1)\mathbf{P}^{+}(t-1)\ddot{\mathbf{O}}(t-1)^{\mathrm{T}} + \mathbf{Q}(t-l). \tag{14}$$

■ Correction step

$$\mathbf{K}(t) = \mathbf{P}^{-}(t)\mathbf{H}(t)^{\mathrm{T}}\Big(\mathbf{H}(t)\mathbf{P}^{-}(t)\mathbf{H}(t)^{\mathrm{T}} + \mathbf{R}(t)\Big)^{-1}. \tag{15}$$

$$\mathbf{s}^{+}(t) = \mathbf{s}^{-}(t)\mathbf{K}(t)\Big(\mathbf{z}(t) - \mathbf{H}(t)\mathbf{s}^{-}(t)\Big). \tag{16}$$

$$\mathbf{P}^{+}(t) = (\mathbf{I} - \mathbf{K}(t)\mathbf{H}(t))\mathbf{P}^{-}(t). \tag{17}$$

The prediction-correction cycle is repeated. Looking at Eq. (15), the measurement error $\mathbf{R}(t)$ and Kalman gain $\mathbf{K}(t)$ are is in inverse ratio. The smaller $\mathbf{R}(t)$ is, the gain $\mathbf{K}(t)$ weights the residual more heavily. In this case, the measurement is trusted more and more, while the predicted result is trusted less and less. On the other hand, as the a priori estimate error $\mathbf{P}^{-}(t)$ approaches zero, the gain $\mathbf{K}(t)$ weights the residual less heavily. The actual measurement is trusted less and less, while the predicted result is trusted more and more.

Therefore, the system will get near optimal result, if we can decide which one we will trust. From Eqs. (13)–(17), the prediction-correction cycle of Kalman filter is repeated. For developing the proposed adaptive Kalman filter, firstly, the moving model of object is constructed and is used as the system state model of Kalman filter. The moving model will be applied in the prediction step of Kalman filter. Then, for the correction step of Kalman filter, a moving object detection method is proposed.

2.2.2. Motion model construction

In Kalman filter algorithm, the system state model is applied in the prediction step. Before using the Kalman filter, the system state model has to be determined. In our system, the system state model is the moving model. Since the time of the frame interval is very short, it is assumed that the moving object is in uniform velocity within a frame interval. Therefore, the velocity of moving object can be replaced as system parameter by position. In addition, the moving distance in every interval is equal and is set as

$$d(t) = d(t-1) + (d(t-1) - d(t-2)), \tag{18}$$

where $d(t-1)$ and $d(t-2)$ are the moving distances in frame $t-1$ and $t-2$, respectively. By Eqs. (6) and (18), the system state model of Kalman filter can be described as

$$\mathbf{s}(t) = \ddot{\mathbf{O}}(t-1)\mathbf{s}(t-1) + \mathbf{w}(t) = \begin{bmatrix} 2 & -1 \\ 1 & 0 \end{bmatrix}\begin{bmatrix} d(t-1) \\ d(t-2) \end{bmatrix} + \begin{bmatrix} w(t) \\ 0 \end{bmatrix}, \tag{19}$$

where $\mathbf{s}(t) = \begin{bmatrix} d(t) \\ d(t-1) \end{bmatrix}$, $\mathbf{s}(t-1) = \begin{bmatrix} d(t-1) \\ d(t-2) \end{bmatrix}$ and $\ddot{\mathbf{O}}(t) = \begin{bmatrix} 2 & -1 \\ 1 & 0 \end{bmatrix}$.

2.2.3. Moving object detection in consecutive frames

In this stage, the position of the moving object in consecutive frame will be detected. Once the dominant color feature in RGB color space is obtained, the color feature is converted from RGB color space to HSI color system, and then a color matching function is used to measure the area of the moving object. The main reason for using HSI color space is to increase tolerance toward light changes in video, since its three components are relative independent. In HSI color space, lighting changes mainly influence the intensity component.

The matching function for detection is written as

$$\mathrm{HSI_{M}atching}(ref, cur) = \begin{cases} 1 & \text{if} \quad \begin{array}{l} |ref\_H - cur\_H| < th\_H \ \& \\ |ref\_S - cur\_S| < th\_S \ \& \\ |ref\_I - cur\_I| < th\_I \end{array} \\ 0 & \text{otherwise} \end{cases} \tag{20}$$

where *ref* is the HSI dominant color in previous frame and *cur* is image pixels of search range in current frame. The search range depends on the matched object of previous frame. The *th_H*, *th_S*, and *th_I* are the thresholds for matching. In HSI color space, its three components (hue, saturation, and intensity) are relatively independent, therefore, in Eq. (20), the comparisons are made separately. If all of the three components are less than their thresholds, then the pixel is in the target object. Otherwise, it is the background. In Fig. 5, we use an image to evaluate the HSI matching function. The three components of dominant color in HSI color space are $H = 3.24$, $S = 0.05$, and $I = 0.89$. In addition, the thresholds are set as $th\_H = 1.05$, $th\_S = 0.5$, and

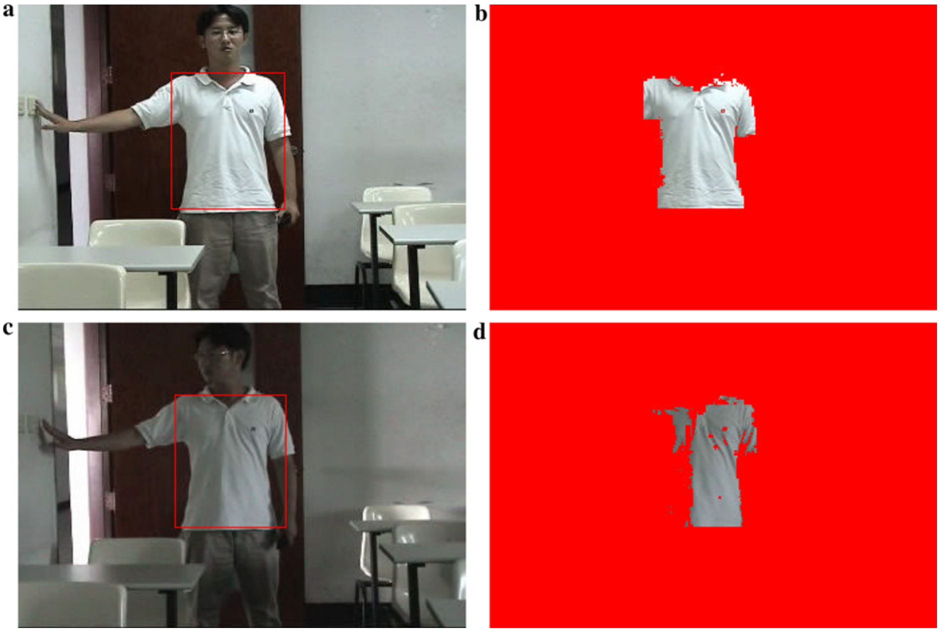

Fig. 5. The experimental results of HSI color matching method. The illumination in (a) and (c) are different.

$th\_I = 0.5$. Note that the illumination in Figs. 5a and c are different. The results are shown in Figs. 5b and d, respectively. Although the HSI matching function can not segment the moving object completely, the result is good enough to represent the object area.

In addition, to represent the position of the object, we have to find the center of the matched object area. We use a rectangle to represent the moving object's area and the rectangle is the minimum one to cover the moving object completely. Then, the center will be $(cx, cy) = (\frac{left+right}{2}, \frac{top+bottom}{2})$, where $(left, top)$ and $(right, bottom)$ are the rectangle's corner positions at left upper and right lower corners, respectively. The rectangle area will be extended some pixels as the search range of the next frame. The left upper and right lower corner positions of search range, $(search\_left, search\_top)$ and $(search\_right, search\_bottom)$, respectively. They are defined as

$$\begin{aligned} &search\_left = (pre\_left - (d(t-1) - d(t-2))),\\ &search\_top = (pre\_top - (d(t-1) - d(t-2))),\\ &search\_right = (pre\_right + (d(t-1) - d(t-2))),\\ &search\_bottom = (pre\_bottom + (d(t-1) - d(t-2))), \end{aligned} \tag{21}$$

where the prefix $pre\_$ represents the previous frame. The $d(t-1)$ and $d(t-2)$ are the object moving distances in frame $t-1$ and $t-2$, respectively. That is to say, the search range considers the object's moving speed. The center of search range is located at the predicted position of adaptive Kalman filter.

The Fig. 6 shows the structure of moving object detection using color matching method in HSI color space. In this structure, the HSI color matching method uses the dominant color DomC of the moving object to match the area of the moving object $\mathrm{MO}(t)$ in the image frame t. The input "predicted object position" in Fig. 6 is the prediction result of adaptive Kalman filter. The $\mathrm{OR}(t)$ is the occlusion rate and $\mathbf{z}(t)$ is the detected object position by HSI matching method. The occlusion rate is the ratio of the occlusion area (pixels number) in frame t to that in frame $t-1$. How to use the ratio to adjust the error parameters of the proposed adaptive Kalman filter will be mentioned in next subsection. The occlusion rate is defined as

$$\alpha(t) = \begin{cases} \left|\frac{PN(t)}{PN(t-1)} - 1\right| & \text{if} \left|\frac{PN(t)}{PN(t-1)} - 1\right| \leqslant 1 \\ 1 & \text{otherwise} \end{cases}, \tag{22}$$

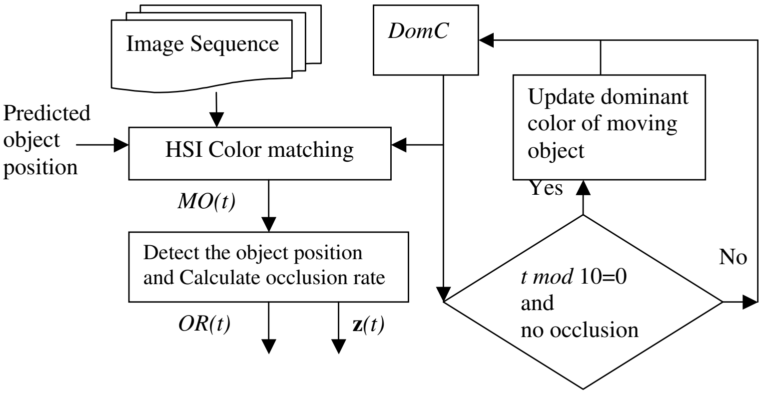

Fig. 6. The structure of moving object detection.

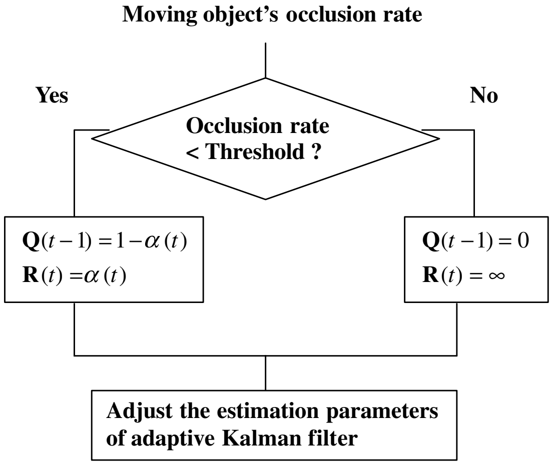

Fig. 7. The flowchart for adjusting the estimation parameters of Kalman filter.

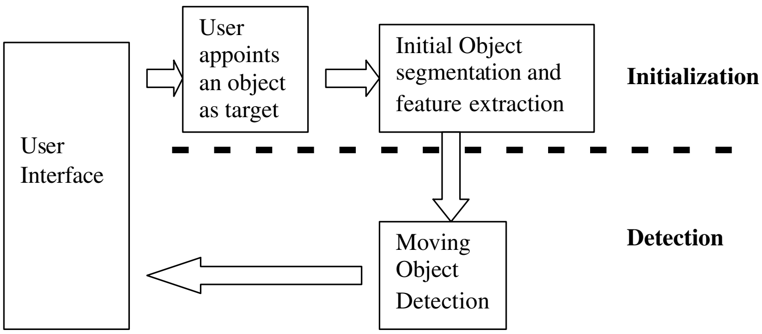

Fig. 8. The flowchart of moving object detection tracking method.

where $PN(t)$ and $PN(t-1)$ are the pixel number of the moving object in frame t and $t-1$, respectively. The value 1 represents the totally occluded. In addition, to resist the lighting probably changes in real world environment, the dominant color will be updated per 10 frames (t mod 10) except the occlusion happens.

2.2.4. Adaptive Kalman filtering

After constructing the motion model and achieving the measurement by moving object detection, we can apply the adaptive Kalman filtering to track the object in video sequences. The system state model in adaptive

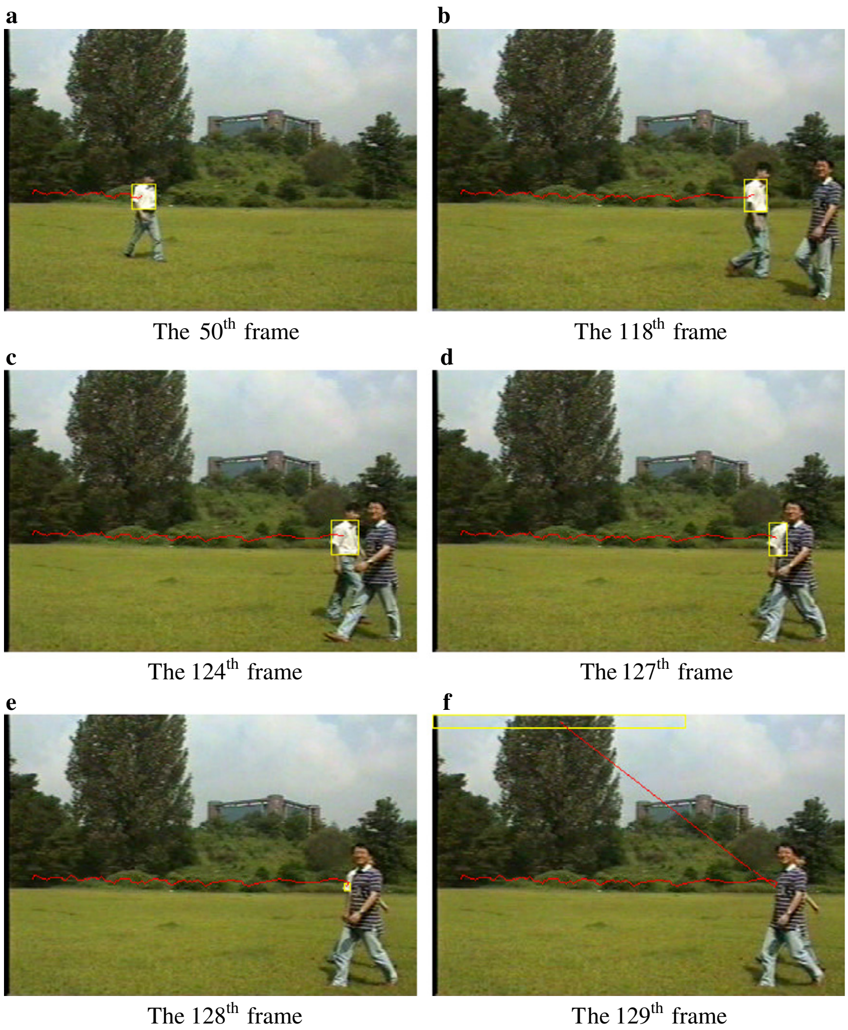

Fig. 9. The experimental results of moving object detection tracking method.

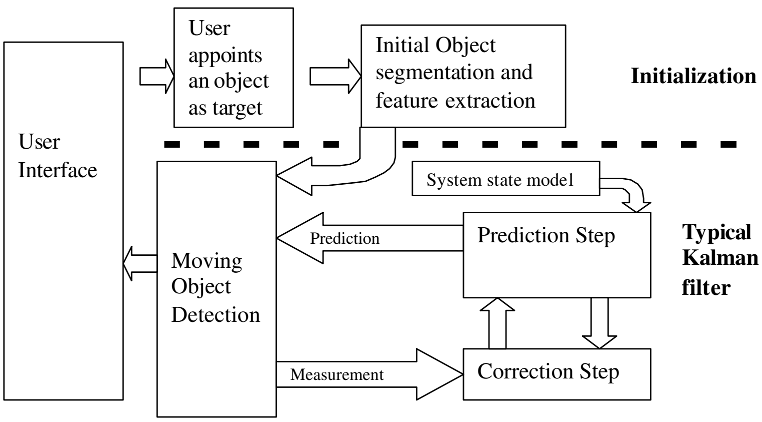

Fig. 10. The flowchart of typical Kalman filter tracking method.

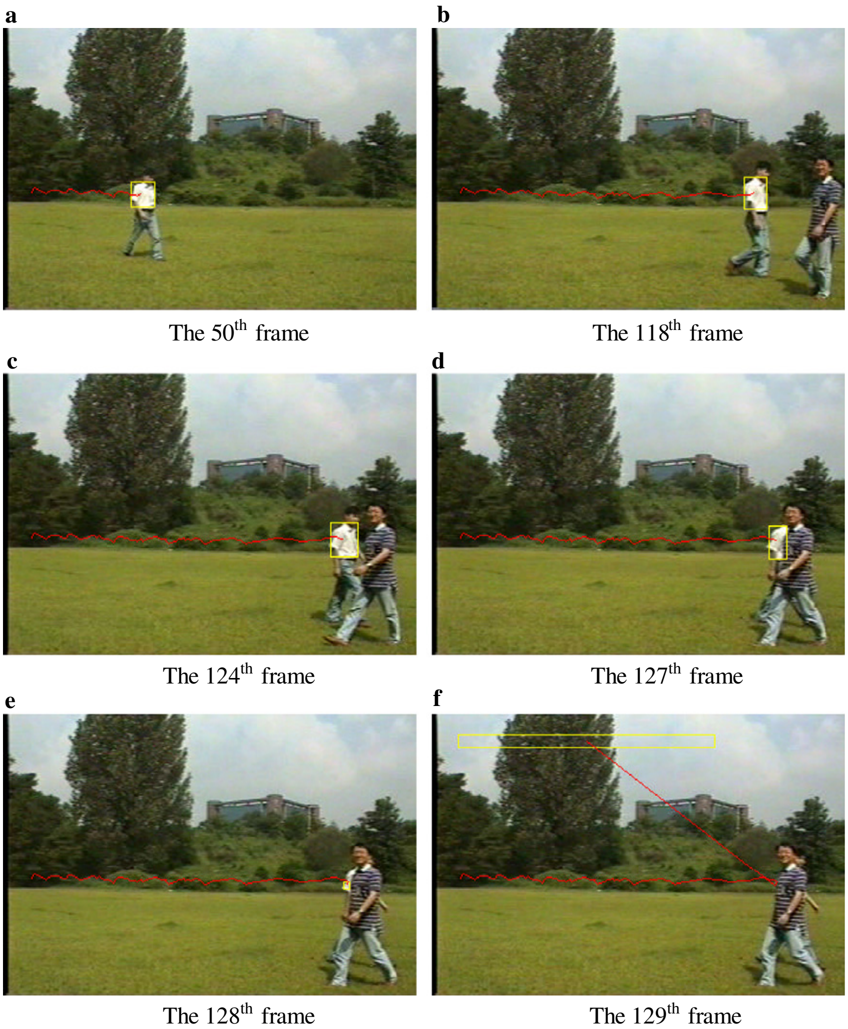

Fig. 11. The experimental results of typical Kalman filter tracking method.

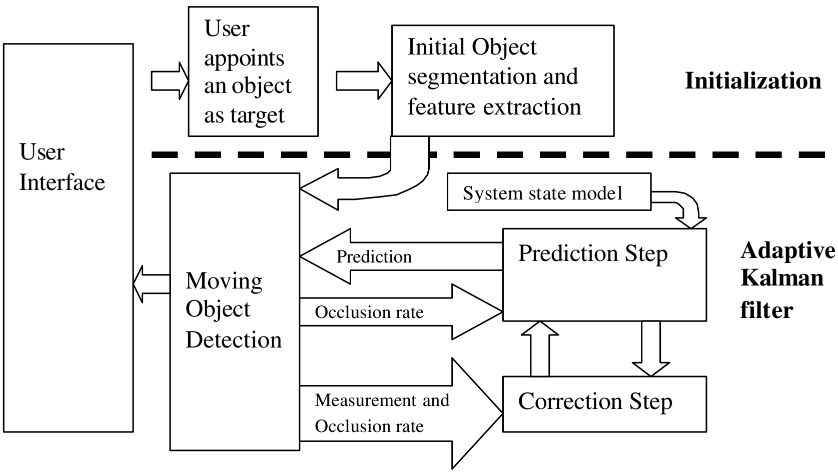

Fig. 12. The flowchart of adaptive Kalman filter tracking method.

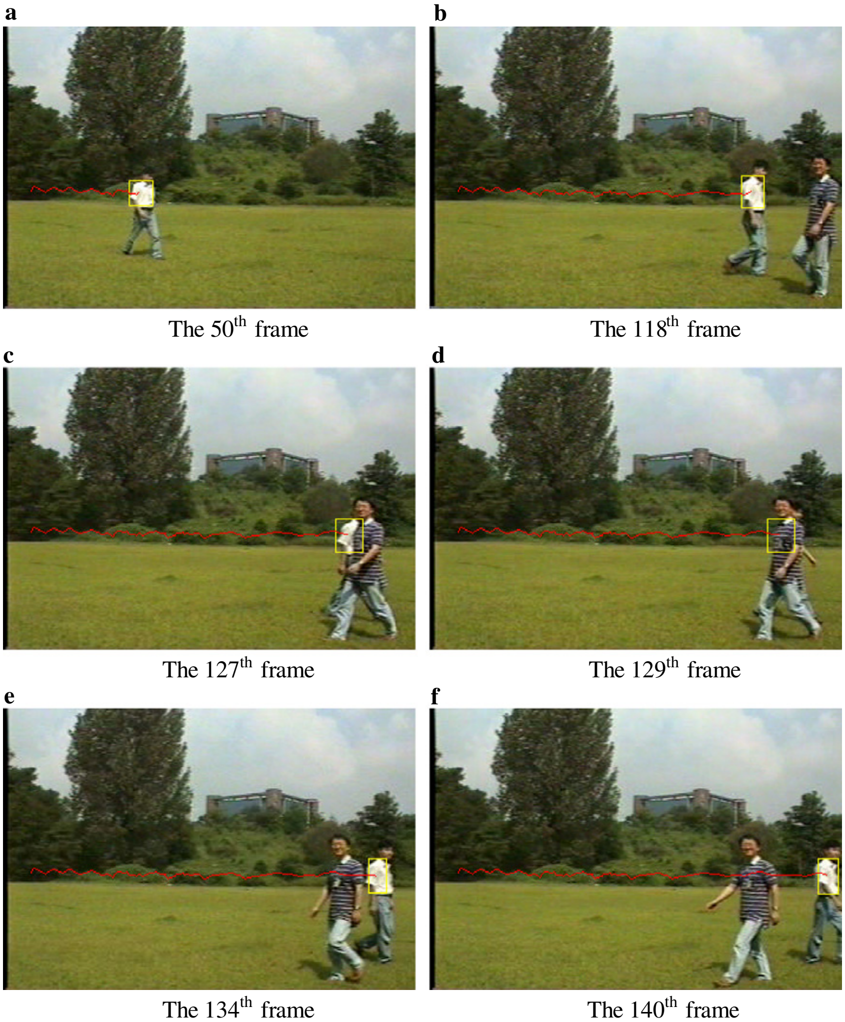

Fig. 13. Tracking results with short-time occlusion.

Kalman filtering is built by the motion model and it is used in prediction step. The so called adaptive Kalman filter is to let the estimate parameters of Kalman filter adjust automatically. Since the measurement error and occlusion ratio are in direct rate, therefore, in correction step, the occlusion rate is used to adjust the estimate parameters of adaptive Kalman filter. According to Eq. (15), the Kalman gain $\mathbf{K}(t)$ is in an inverse proportion to the measurement error $\mathbf{R}(t)$. If the occlusion rate is less than the threshold, then the value of $\mathbf{R}(t)$ is set as occlusion rate $\alpha(t)$ and the prediction error $\mathbf{Q}(t-1)$ of Eq. (14) is $1-\alpha(t)$. Otherwise, it is reasonable to let the measurement error $\mathbf{R}(t)$ and the prediction error $\mathbf{Q}(t-1)$ be infinity and zero, respectively, thus the Kalman gain $\mathbf{K}(t)$ is a zero value. According to the occlusion rate, the Kalman filter system is adjusted adaptively. That is to say, if the occlusion rate is less than the threshold, then the measurement result will be trusted more than predicted one. Otherwise, the system will trust the predicted result completely. Fig. 7 illustrates the flow chart for adjusting the estimate parameters of Kalman filter adaptively. In addition, using the moving model mentioned in previous subsection, the parameters of measurement model in Eq. (7) are defined as $\mathbf{s}(t)=\begin{bmatrix} d(t) \\ d(t-1) \end{bmatrix}$, $\mathbf{H}(t)=[1\ 0]$ and $\mathbf{z}(t)=[d(t)]$. By this way, the Kalman filter can be adjusted automatically to estimate the moving object.

Finally, we summarized the proposed tracking algorithm using adaptive Kalman filter again. In the initialization, moving object segmentation and feature extraction are included. First, the frame difference and region

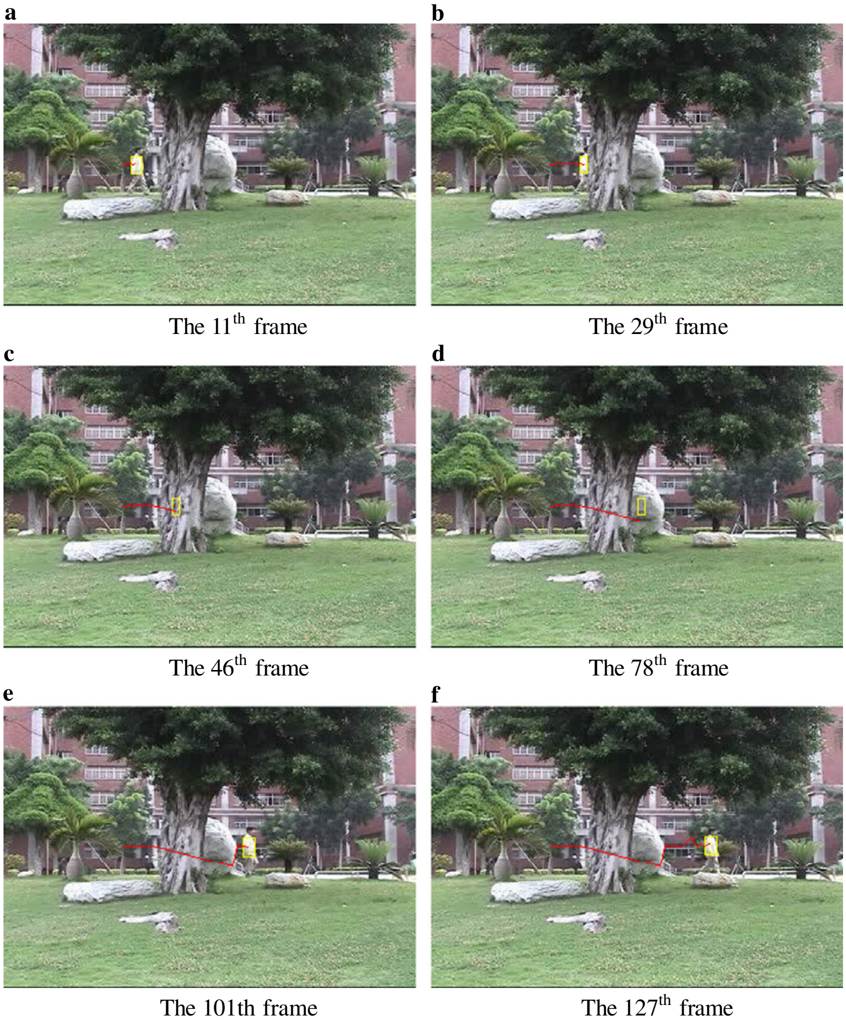

Fig. 14. Tracking results with long-lasting occlusion.

growing are used to segment moving object. Then, the dominant color is extracted from the segmented moving object. In the tracking procedure of adaptive Kalman filter, a motion model is constructed to build the system state and is applied to prediction step. In addition, the measurement of system is provided by moving object detection using HSI matching method. In addition, the occlusion ratio is applied to adjust the prediction and measurement errors adaptively. The two errors will make the adaptive Kalman filter system to trust prediction or measurement more and more.

3. Experimental results

In this section, we make comparisons between the proposed adaptive Kalman filter method and the others including moving object detection method without Kalman filter involved and typical Kalman filter. About experimental tools, the PC with AMD XP 2400 processor, Window XP operation system, and Borland C++ Builder 6.0 are used. In addition, the video image consists of bitmap image sequence with 352×240 pixels per frame. Some kinds of experimental videos are used to evaluate the robust ability of the proposed method, including the moving object disappearing totally or partially due to occlusion by other ones, the object which is fast moving, changing the orientation and direction of the moving object, and changing the velocity of moving object suddenly.

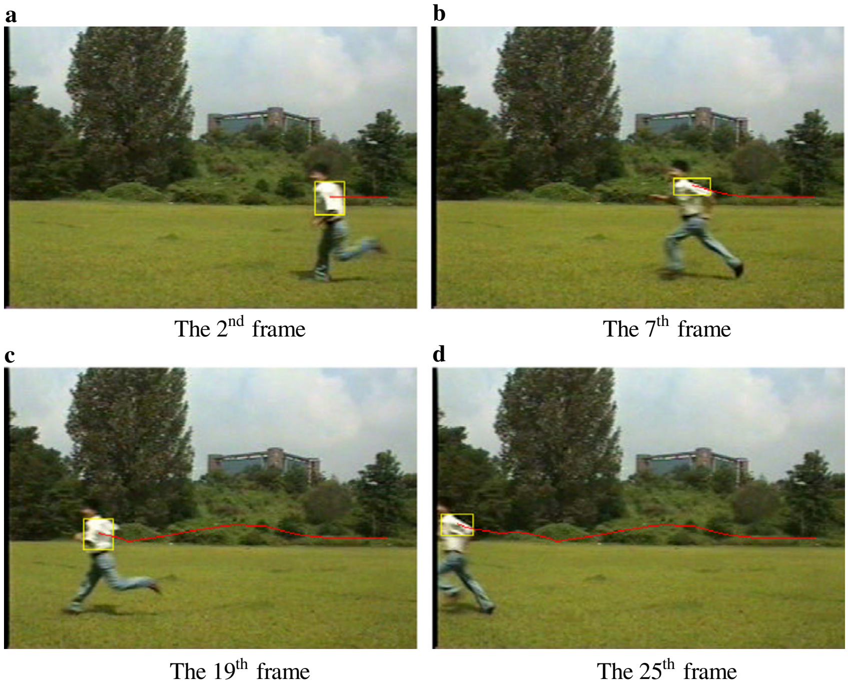

Fig. 15. Tracking results with the fast moving person.

It is noted that a rectangle window is shown to track the moving target in all of our experiments. In the first experiment, the moving object detection method mentioned in Section 2.2.3 is tested. The flowchart and the experimental results are shown in Fig. 8 and in Fig. 9, respectively. From the sequence of test images, Figs. 9a–d, a window tracks the moving person correctly. This represents that the moving object detection method can correctly track the moving person who is not occluded by the other one. However, when the moving person is occluded more and more by the other one, in Figs. 9e and f, the moving person can not be tracked anymore. The tracking window loses tracks of the moving target. Obviously, the moving object detection method can not be used in occluded situation.

In the second experiment, the typical Kalman filter is applied. In this method, the system is the same as Section 2.2.1 and the system state model is mentioned in Section 2.2.2. Both of the prediction error and measurement error are set as constant. They are experimentally given 0.8 and 0.2, respectively. The flowchart is shown in Fig. 10. The experimental results are demonstrated in Fig. 11. The results show that this method can not successfully track the moving person when the person is occluded by the other one. The results are the same with the first experiment.

In the following experiments, the proposed adaptive Kalman filter is evaluated. The flowchart of the proposed adaptive Kalman filter is shown in Fig. 12. The occlusion rate is used to adjust the prediction error and measurement error adaptively. If the occlusion rate is less than 0.3, then the measurement error is set as occlusion rate $\alpha(t)$ and the prediction error is $1 - \alpha(t)$. Otherwise, the measurement error will be set as infinity and prediction error is zero. Fig. 13 shows that the moving person can successfully keep tracking in occlusion and after being occluded by the other one. In Fig. 14, we test the case of long-lasting occlusion. The duration of the occlusion is from the 29th frame to near the 100th frame. To handle long-lasting occlusions are very difficult; however, if the moving object keeps moving in the uniform velocity and in the same direction, the occluding duration of the proposed method will depend on the occlusion time. The experimental results show that the proposed method can track the moving object successfully when it disappears completely or partially due to occlusion by other ones. In following several experiments, we test the cases for the fast moving object (Fig. 15), changing object' s direction and orientation (Fig. 16) and changing the velocity suddenly (Fig. 17), respectively. Particularly, in Fig. 17, the case of

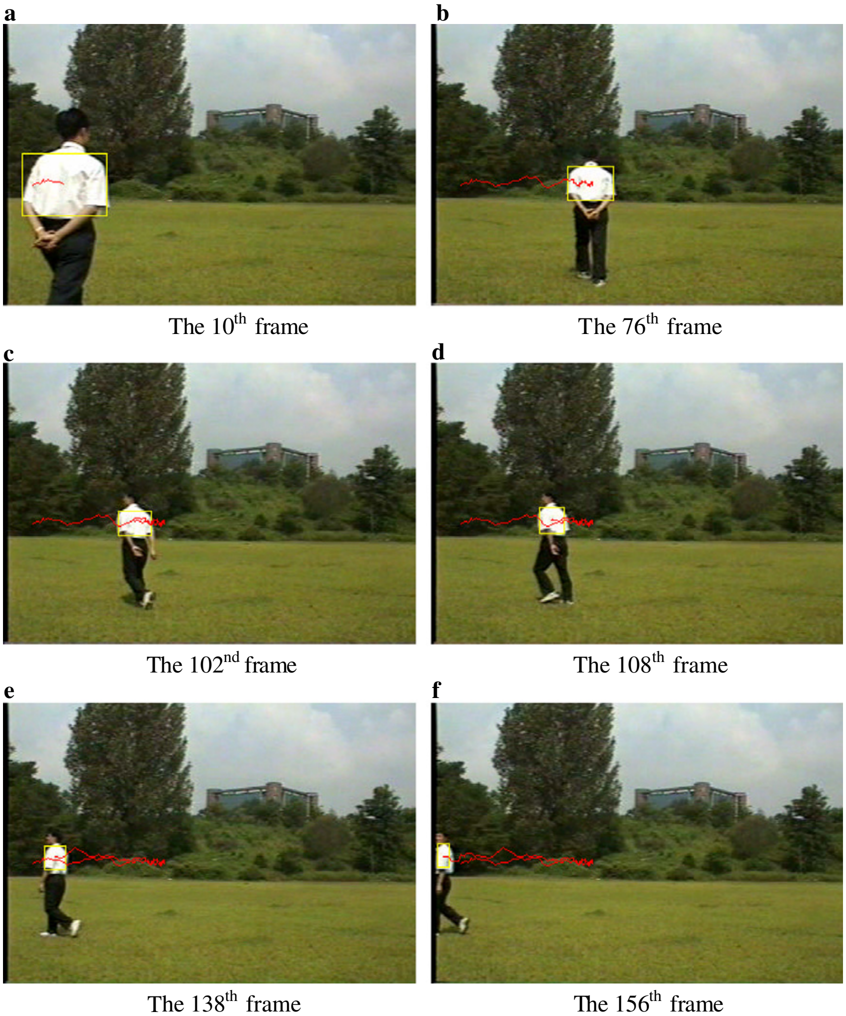

Fig. 16. Tracking results with changing direction and orientation.

changing velocity, the moving person in Fig. 17d stops and suddenly run fast in Fig. 17e and f. The results show that the proposed tracking method can track the moving person in these cases effectively in real time.

Finally, Table 1 lists the processing time of the test video sequences using the proposed tracking methods, respectively. The average processing time of every frame is between 0.01 and 0.02 second. Therefore, it can be applied to real-time application.

4. Conclusion and discussion

In this paper, an effective adaptive Kalman filter is proposed to track the moving object. In the proposed adaptive Kalman filter, the occlusion rate is used to adjust the error covariance of Kalman filter adaptively. The method can track the moving object in real-time. It successfully estimates the object's position in some kinds of real-world situations such as the fast moving object, partial occlusion, long-lasting occlusion, changing lighting, changing the direction and orientation of the moving object, and changing the velocity of moving object suddenly. Furthermore, to consider the situations of tracking multiple objects, every one of multiple objects can be set an adaptive Kalman filter to track it. Since the processing time using the proposed method to track the single object is short,

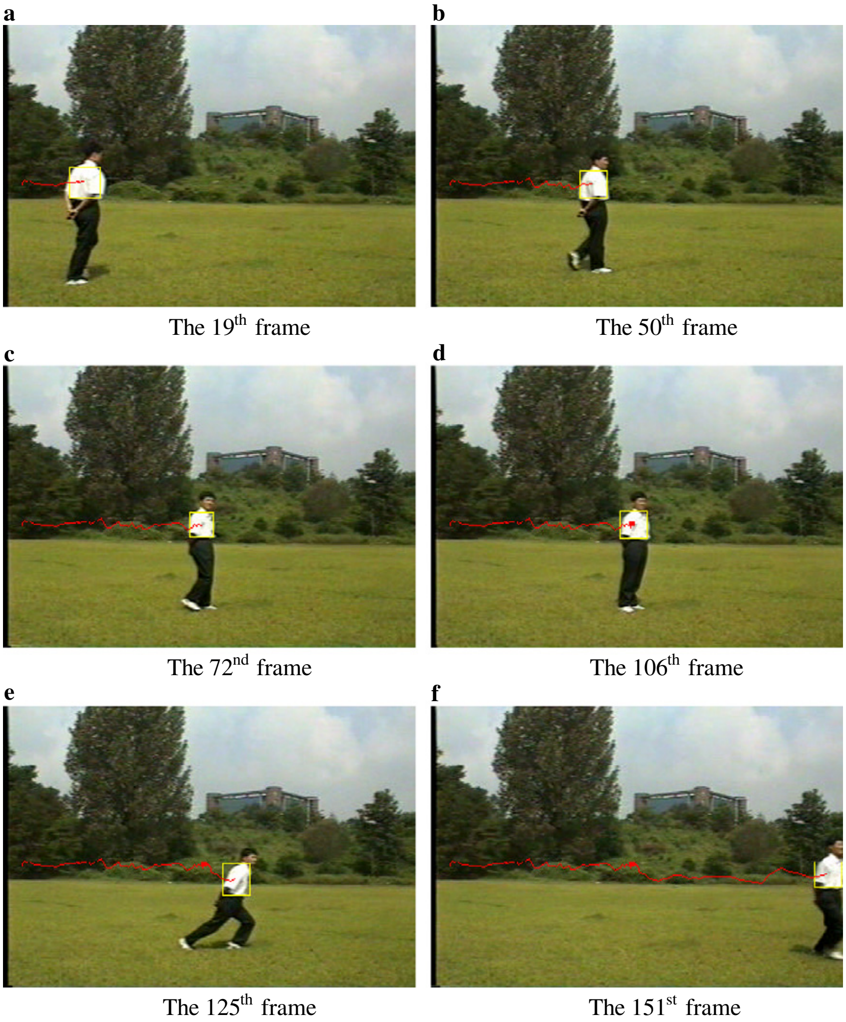

Fig. 17. Tracking results with the moving person in changing velocity suddenly.

Table 1
The processing time of test video sequences

Test video sequences	Frame number	Tracking time (seconds)
In Fig. 13	141	1.439
In Fig. 14	150	1.612
In Fig. 15	25	0.252
In Fig. 16	156	2.588
In Fig. 17	151	2.043

therefore, the systems implemented by the proposed method can afford to track multiple objects in real time.

Acknowledgments

The authors thank the anonymous reviewers for their valuable comments and suggestions. This work was supported by I-Shou University under Grant number ISU-94-01-16.

References

[1] Dae-Sik Jang, Seok-Woo Jang, Hyung-Il Choi, 2D human body tracking with structural Kalman filter, Pattern Recognition 35 (2002) 2041–2049.
[2] Dae-Sik Jang, Hyung-Il Choi, Active models for tracking moving objects, Pattern Recognition 33 (2000) 1135–1146.
[3] Shao-Yi Chien, Shyh-Yih Ma, Liang-Gee Chen, Efficient moving object segmentation algorithm using background registration technique, IEEE Transactions on Circuits and Systems for Video Technology 12 (7) (2002) 577–586.
[4] Changick Kim, Jenq-Neng Hwang, Fast and automatic video object segmentation and tracking for content-based applications, IEEE Transactions on Circuits and Systems for Video Technology 12 (2) (2002) 122–129.
[5] Dong Kwon Park, Ho Seok Yoon, Chee Sun Won, Fast object tracking in digital video, IEEE Transactions on Consumer Electronics 46 (3) (2000) 785–790.
[6] Greg Welch, Gary Bishop, An Introduction to the Kalman Filter, UNC-Chapel Hill, TR 95-041, March 11, (2002).
[7] Yiwei Wang, John F. Doherty, Robert E. Van Dyck, Moving object tracking in video, in: Proceedings of the 29th workshop of Applied Imagery Pattern Recognition, 2000, October 16–18, 2000, pp. 95–101.
[8] Alan J. Lipton, Hironobu Fujiyoshi, Raju S. Patil, Moving target classification and tracking from real-time video, Proceedings of the Applications of Computer Vision, WACV'98, October 19–21, 1998, pp. 8–14.
[9] Johnson I. Agbinya, David Rees, Multi-object tracking in video, Real-Time Imaging (1999) 295–304.
[10] Roberta Piroddi, Theodore Vlachos, Multiple-feature spatiotemporal segmentation of moving sequences using a rule-based approach, British Machine Vision Conference (2002) 353–362.
[11] Volker Rehrmann, Object oriented motion estimation in color image sequences, in: ECCV '98, vol. I, Lecture Notes of Computer Science 1406, 1998 704–719.
[12] Michael Mason, Zoran Duric, Using histograms to detect and track objects in color video, Applied Imagery Pattern Recognition Workshop, AIPR 2001 30th, October 10–12, 2001, pp. 154–159.
[13] Georgi D. Borshukov, Gozde Bozdagi, Yucel Altunbasak, A. Murat Tekalp, Motion segmentation by multistage affine classification, IEEE Transactions on Image Processing 6 (11) (1997) 1591–1594.
[14] George V. Paul, Glenn J. Beach, Charles J. Cohen, A Real-time object tracking system using a color camera, Applied Imagery Pattern Recognition Workshop, AIPR 2001 30th, October 10–12, 2001, pp. 137–142.
[15] Jong Bae Kim, Hye Sun Park, Min Ho Park, Hang Joon Kim, A Real-Time Region-Based Motion Segmentation Using Adaptive Thresholding and *K*-means Clustering, Kyungpook National University, Computer Engineering, South Korea.
[16] H. Tao, H.S. Sawhney, R. Kumar, Dynamic layer representation with applications to tracking, Proceedings of the Computer Vision and Pattern Recognition 2 (2000) 134–141.
[17] Y. Wu, T.S. Huang, A co-inference approach to robust visual tracking, Proceedings of the International Conference and Computer Vision 2 (2001) 26–33.
[18] A.D. Jepson, D.J. Fleet, T.F. El-Maraghi, Robust online appearance models for visual tracking, Proceedings of the Computer Vision and Pattern Recognition (2001).
[19] J. Vermaak, P. Perez, M. Gangnet, A. Blake, Towards improved observation models for visual tracking: selective adaptation, Proceedings of the European Conference on Computer Vision 1 (2002) 645–660.
[20] Hieu T. Nguyen, Arnold W.M. Smeulders, Fast occluded object tracking by a robust appearance filter, IEEE Transaction on Pattern Analysis and Machine Intelligence 26 (8) (2004) 1099–1104.
[21] T.B. Moeslund, E. Granum, A survey of computer vision-based human motion capture, Computer Vision and Image Understanding 81 (2001) 231–268.
[22] H. Sidenbladh, M.J. Black, D.J. Fleet, Stochastic Tracking of 3D Human Figures Using 2D Image Motion, Proceedings of the European Conference on Computer Vision (2000) 702–718.
[23] Y. Song, X. Feng, P. Perona, Towards detection of human motion, Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (2000) 810–817.
[24] N.T. Siebel, S. Maybank, Fusion of multiple tracking algorithm for robust people tracking, Proceedings of the European Conference on Computer Vision (2002) 373–387.
[25] T. Zhao, R. Nevatia, Tracking multiple humans in complex situations, IEEE Transaction on Pattern Analysis and Machine Intelligence 26 (9) (2004) 1208–1221.
[26] A. Senior, Tracking people with probabilistic appearance models, Proceedings of the International Workshop on Performance Evaluation of Tracking and Surveillance Systems (2002).
[27] Vance Faber, Clustering and the continuous *K*-means algorithm, Los Alamos Science (22) (1994) 138–144.
[28] R.C. Gonzalez, R.E. Woods, Digital Image Processing, Prentice-Hall, Englewood Cliffs, NJ, 2002.

Shiuh-Ku Weng received the B.S. degree from the Feng Chia University in information engineering, Taiwan, in 1991 and both M.S. and Ph.D. degrees were obtained in electronic engineering from Chung Yuan Christian University, Taiwan, in 1993 and Chung Cheng Institute of Technology, National Defense University, Taiwan, in 1997, respectively. In 1999, he joined the faculty of the Department of Electrical Engineering at Chinese Naval Academy for four years. In 2003, he was invited to join the Department of Information Management at the same school. His research interests include video object tracking, image segmentation, optimal estimation, and signal detection.

Chung-Ming Kuo received the B.S. degree from the Chinese Naval Academy, Kaohsiung, Taiwan, in 1982, the M.S. and Ph.D. degrees all in electrical engineering from Chung Cheng Institute of Technology, Taiwan, in 1988 and 1994, respectively. From 1988 to 1991, he was an instructor in the Department of Electrical Engineering at Chinese Naval Academy. Since January 1995, he has been an associate professor at the same institution. From 2000 to 2003, he was an associate professor in the Department of Information Engineering at I-Shou University. Since February 2004, he has been Professor at the same institution. His research interests include image/video compression, image/video feature extraction, understanding and retrieving, motion estimation and tracking, multimedia signal processing, and optimal estimation.

Shu-Kang Tu was born in Kaohsiung, Taiwan, R.O.C., in 1978. He received the B.E., and M.E. degree from the Department of Information Engineering, I-Shou University, Kaohsiung, R.O.C., in 2002, and 2004. He is working in Best Wise International Computing CO., LTD, currently. His research interests include video tracking and segmentation algorithms.