

# Real-Time Object Tracking and Segmentation Using Adaptive Color Snake Model

Kap-Ho Seo, Jin-Ho Shin, Won Kim, and Ju-Jang Lee

**Abstract:** Motion tracking and object segmentation are the most fundamental and critical problems in vision tasks such as motion analysis. An active contour model, snake, was developed as a useful segmenting and tracking tool for rigid or non-rigid objects. In this paper, the development of new snake model called “adaptive color snake model (ACSM)” for segmentation and tracking is introduced. The simple operation makes the algorithm runs in real-time. For robust tracking, the condensation algorithm was adopted to control the parameters of ACSM. The effectiveness of the ACSM is verified by appropriate simulations and experiments.

**Keywords:** Active contours, condensation algorithm, object tracking, image segmentation.

---

## 1. INTRODUCTION

In recent years, automatic controls have affected the world in various fields. Automation has been carried out in factories, military, services in hospitals, and so on. The controllers of high performance need large amounts of integrated information about systems and the environment for feedback. A vision system is a sensor that is capable of giving this kind of information. Vision sensors are necessary for sensing obstacles or targets in complex systems such as mobile robots, active and intelligent cruise controls. In a vision system, processing such as segmentation and tracking are critical tasks to be accomplished satisfactorily.

A considerable work has been done during the past decade in object tracking and motion analysis of nonrigid objects in the context of snake models. Active contour models have been developed as useful tools for segmenting and tracking rigid and nonrigid objects. Snake, one of the active contour models, was introduced by Kass *et al.* [1] in 1987. They defined snake energies such as internal energy, image energy

and external energy. Segmentation and tracking can be done by this energy minimization process. They tried to solve optimization problem for energy minimization by use of the variational approach. They have applied snake to track facial features such as lips in an image sequence. The estimated motion parameters of these features were used to explain facial expressions, etc. Leymarie and Levine [2] have used the snake model to track cells in biological image sequences and proved the convergence of snake's motion. DeCarlo and Metaxas [3] have proposed a deformable face model which includes both shape and motion parameters and have applied it to track human faces. Point distribution based active shape models were also proposed by Kervrann and Heitz [4] to track objects in long image sequences, where a point distribution is used to characterize the structure and variations in the object shape.

Besides the object boundaries, various information in an image or a sequence of images have been used for segmenting objects. Appearance information including grayscale or color have been used in conjunction with deformable shape models [5-8]. Bascle and Deriche [9] have combined texture correlation of the entire internal region of the object and B-spline contour along the object boundary. Color information has also been used to track nonrigid objects in real-time application [7,10,11].

While an active contour model is applied to track an object, stepping to the next image frame in the presence of a large motion difference between them may cause unpredictable effects on the tracking contour. If the variance of the object's location and the configuration between two successive images is large due to the abrupt increase of the object's speed or the low operation speed of the vision system, then the

---

Manuscript received February 28, 2005; revised October 25, 2005; accepted December 26, 2005. Recommended by Editorial Board member Sangdeok Park under the direction of Editor Keum-Shik Hong. This work was supported by the SRC/ERC program of MOST/KOSEF (grant #R11-1999-008).

Kap-Ho Seo, Won Kim, and Ju-Jang Lee are with the Department of Electrical Engineering and Computer Science, KAIST, 373-1 Guseong-dong, Yuseong-gu, Daejeon 305-701, Korea (e-mails: neoworld@kaist.ac.kr, kimwon@odyssey.kaist.ac.kr, jjlee@ee.kaist.ac.kr).

Jin-Ho Shin is with the Department of Mechatronics Engineering, Dong-Eui University, San 24, Gaya-dong, 995 Eomgwangno, Busanjin-gu, Busan 614-714, Korea (e-mail: jhshin7@deu.ac.kr).

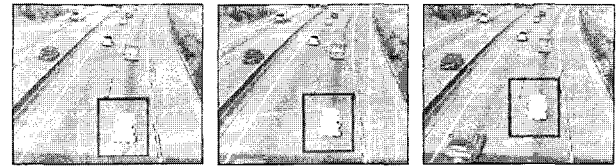
tracking of snake cannot be guaranteed. For the better tracking performance, some advanced algorithms were employed for motion estimation in the snake model.

Optical flow algorithm has been commonly applied to estimate the object's motion [3,12,13]. But, the computation of the optical flow field for the entire area of interest lead to a considerable computational complexity. It may be effective only for a static camera. As the camera moves, the system generates many variations between successive images. Optical flow-based approach is not suitable for these situations. Another approach is the Kalman filter, a powerful tool for motion analysis [14,15], which is an frequently used in dynamic estimations. Kalman filtering requires the system to be linear, with its observations be linear functions of the underlying state. The most obvious difficulties of the visual tracking lie in the modeling of the shape uncertainty and the probability density of the state. Most of the algorithms that use the state-space description are based on Kalman filtering. Kalman filtering has a mechanism that fuse the current noisy measurement information and the estimation based on the history of states. Kalman filtering deals with the uncertainty of the state by carrying the covariance matrices of the states and the measurements, and it works optimally when the noise is Gaussian. This indicates that Kalman filtering algorithm can fail when the uncertainty in the system has multi-modal distribution. This is exactly what happens in the noisy measurement data from an image with several visual clutter, which causes malfunction of the Kalman filter in such situations. The condensation algorithm has been employed in many researches [16-18]. One of the striking properties of condensation algorithm is its simplicity compared to the Kalman filter. In presence of camera motion, the condensation algorithm shows the robust tracking of object's motion. In following sections, the condensation algorithm will be described in detail.

In this paper, we propose a new snake model, which is capable of dealing with color information. And, we also present a real-time algorithm to track and segment a moving object using the condensation algorithm for better tracking performance. In Section 2, the problem statement is discussed. Section 3 and 4 review the conventional snake and condensation algorithm. The proposed snake model, ACSM and tracking scheme are presented in Section 5 and 6. Experimental results with ACSM on a real video sequence are presented in Section 7. The conclusions are given in Section 8.

## 2. PROBLEM STATEMENT

Visual tracking involves successive segmentation of



(a)  $(i-1)$ th frame. (b)  $i$ th frame. (c)  $(i+1)$ th frame.

Fig. 1. Example of visual tracking.

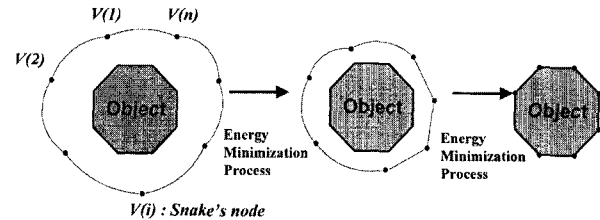
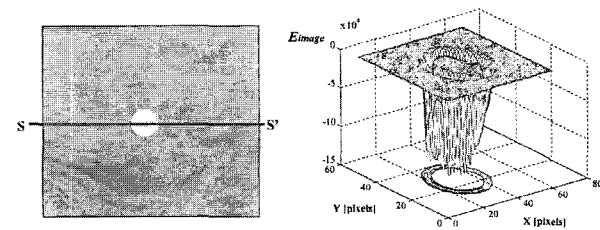


Fig. 2. The concept of segmentation process.



(a) Original image.

(b) Image energy.

Fig. 3. The profile of image energy.

the target object's boundaries in a sequence of images. Fig. 1 shows three sequential images for visual tracking of a particular object in them. Accurate visual tracking involves the successive segmentation of the object's boundary. Snake's segmentation process is related with energy minimization. Snake energies should be defined in such a way that include the boundary information as minimum energy states. If the minima of the energy surface is found accurately through the energy minimization process, snake can figure out the object's boundary. This is the segmentation process [5]. When the variance of the object's location and the configuration between two successive images is small, snake can also make segmentation of the succeeding image from the contour of the preceding image through the process of energy minimization. This is the tracking process of snake.

When the variance between successive images is small, the snake operates well by the variational solution [1] or dynamic programming method [19]. However, when there is the large variance, the convergence of snake motion cannot be guaranteed as the assumptions of variational approach is not effective in such situations. This is a major problem of snake in objects tracking. Furthermore, when the object moves in a complex background, using only image intensity information is not enough to separate the object from its background. This is another major

problem of snake in segmenting objects.

Three kinds of problems are discussed in this section: segmentation problem, tracking problem, and contraction problem.

### 2.1. Segmentation problem

Snake can find an object's boundary by the process of energy minimization in the energy space defined on the basis of the object's geometry and the intensity information of the image.

In Fig. 2, the segmentation process is explained conceptually. Snake contour may contract into the boundary of the target object from the initial location along the gradient of the energy related to the object's geometry. Snake's contour may be expressed by the set of node vectors in discrete cases. Snake can find the exact boundary of the object by searching the maximum gradient points. Kass defined image energy as a function of the gradient of image intensity as

$$E_{image}(x, y) = -|\nabla I(x, y)|^2. \quad (1)$$

Fig. 3 is a representation of the image energy. In Fig. 3(a), an original image is presented, where is a ball on green grass. By the gradient process of (1) the original image can be expressed as sets of energy values. In Fig. 3(b), the points that exist within the boundaries of the object can be identified, as they have minimum values in the image energy space. Therefore, snake can make segmentation of the object from an image. However, in this process, the conventional snake do not utilize the prior knowledge about color information, which often plays an important role in improving the detection performance in the presence of disturbing image features.

### 2.2. Tracking problem

When an image is grabbed by the image-grabbing unit, the tracking procedure finds the object's boundary, despite the variation of the object's location from the preceding image. If the variation is small, the solutions gained from the variational approach are effective, and the snake can find the object's boundary by the gradient of the image energy. If it is not the case, the effectiveness of the solutions of the snake's motion is not guaranteed. This concept is further explained in Fig. 4. The energy profile, which can be gained by cutting the object's image energy along the S-S' axis in Fig. 3(a), is expressed in Fig. 4. In this figure, the snake's nodes are in the segmentation state. Therefore, two representative nodes, black dots, are placed in the minimum energy states of the image energy. If the ball on the grass moves to the right, the energy profile of Fig. 4(a) also moves to the right. In Fig. 4(b) the result of motion is expressed when the variance of motion is small. As the motion is large, the positions of nodes may remain in the valley

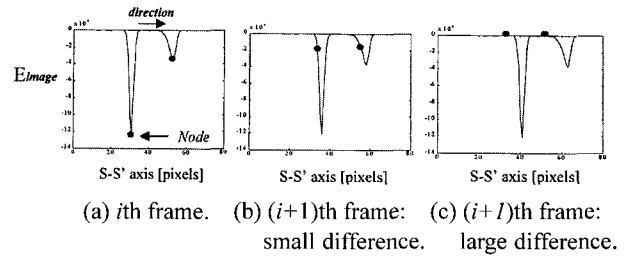


Fig. 4. The concept of snake's tracking.

created by the gradient of image energy. Each node can also move down to the minimum energy state along the gradient of image energy. This way, the tracking is possible in small variance cases. Large variance cases are conceptually explained in Fig. 4(c), where the degree of motion is large, and the nodes of snake may not exist in the valley of energy. Therefore, there may be no gradient of energy and no moving force can be generated. This is a tracking problem of Kass' snake in large variation cases. Consequently, the solutions are not guaranteed in large motion.

### 2.3. Contraction force problem

Kass proposed internal energy to make the contour of snake have smooth geometry. Snake's internal energy is composed of two parts, i.e., the first and second derivatives of the contour along the spline. The snake's contour can be expressed in discrete form as shown in Fig. 5(a), and the derivative operations are approximated by finite difference equations. The internal energy,  $E_{int}$ , at  $i$ th node can be expressed as

$$E_{int}(i) = \alpha_i \cdot |\vec{v}_i - \vec{v}_{i-1}|^2 + \beta_i \cdot |\vec{v}_{i-1} - 2\vec{v}_i + \vec{v}_{i+1}|^2. \quad (2)$$

In the initial segmentation mode, it is required to generate contraction forces into the center of the object. The contraction forces are related with the second term in (2). If we introduce  $\vec{u}_i$  such that

$$\vec{u}_i = \vec{v}_i - \vec{v}_{i-1} \quad (3)$$

then  $\vec{u}_{i+1} - \vec{u}_i$  can be expressed as follows:

$$\vec{u}_{i+1} - \vec{u}_i = \vec{v}_{i+1} - 2\vec{v}_i + \vec{v}_{i-1}. \quad (4)$$

From (4) it is evident that the minimization of  $E_{int}$  is along the direction of  $\vec{u}_{i+1} - \vec{u}_i$ . The contraction

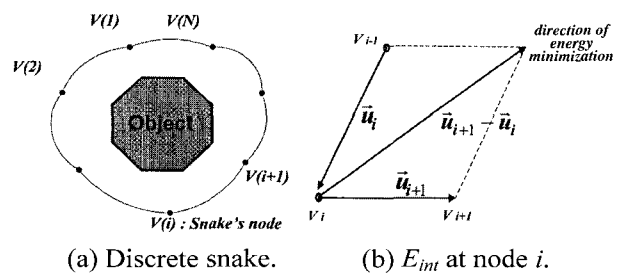


Fig. 5. Contraction force of  $E_{int}$ .

forces are generated by the second term of  $E_{int}$  in (2). The direction of the contraction force at node  $i$  is expressed in Fig. 5(b). If objects have rectangular shapes like automobiles for example, the  $E_{int}$  may have larger values at corners compared to the other points. Moreover, if objects have non-convex shapes, then the contraction forces may not be generated into the center of the object at the points in the vicinity of non-convex region. Therefore, a new energy has to be formulated to avoid effects from the shape of the object. New contraction energy designed on the basis of quadratic functions is proposed in Section 6.

### 3. CONVENTIONAL SNAKE

Kass introduced snake concept in 1987, where he proposed a contour, a deformable curve, and its energies and solutions based on variational approach. Consider a deformable curve  $v(s,t)$  with parameters  $s$  (spatial index) and  $t$  (time index), defined in given open intervals  $\Omega$  and  $T$ , where it is a function of two variables  $x$  and  $y$  with the parameterization

$$v(s,t) = (x(s,t), y(s,t)); \quad s \in \Omega, t \in T. \quad (5)$$

Kass defined energy terms, that are functions of the contour  $v(s,t)$ . Active contour models are the energy-minimizing spline guided by internal constraint forces and influenced by external image forces that pull it toward features such as lines and edges [20]. The conventional energy function in the snake is defined in discrete form as follows:

$$E_{snake} = \sum_{i=1}^n [E_{int}(i) + E_{image}(i)], \quad (6)$$

where  $n$  is the number of snake node (called *snaxel*  $v_i$ ), the internal spline energy  $E_{int}(i)$ , and the external image energy  $E_{image}(i)$ , at the  $i$ th position of the contour, are described as

$$E_{int}(i) = \alpha_i |v_i - v_{i-1}|^2 + \beta_i |v_{i-1} - 2 \cdot v_i + v_{i+1}|^2, \quad (7)$$

$$E_{image}(i) = -|\nabla f(v_i)|^2, \quad (8)$$

where  $f(v_i)$  is the original pixel value at  $v_i$ . The symbols  $\alpha_i$  and  $\beta_i$  are the weighting factors. The image energy is set to be the negative magnitude of the image gradient, so that the snake is attracted to the regions with low image energy, i.e., strong edges. Thus a snake model is actually a function with a compromise balance between internal and image forces.

### 4. ADAPTIVE COLOR SNAKE MODEL

Although conventional snake has proved to be a very attractive and an efficient method, it is not

suitable for complex situations, where fast motion or abrupt variations of object contours are encountered.

The critical disadvantage of conventional snake is not to utilize the prior knowledge such as color and motion information, which often plays an important role in improving the detection performance in the presence of some disturbing image features (e.g. shadows, shading and neighboring objects). Therefore our attention is directed towards the use of color and motion information. This consideration provides more accurate results compared to conventional algorithms, in which objects are tracked by exploiting motion information; this is more evident in sequences with complicated content, such as fast motions, complex backgrounds, special camera effects (zooming, panning, tilting), etc.

Approach of ACSM is somewhat different from other approaches that utilize the color information in active contours [6-8,12]. Most previous work has been based on the gradient of color intensity, in that a mere extension on gradient of gray image intensity is used. ACSM doesn't depend on gradient of color intensity, but it depends on the stochastic matching degree of color.

#### 4.1. Tracking problem

We have proposed *color snake model* in previous work [23,24]. The main idea was the redesign of the snake node, which can deal with color information. But, snaxels in previous research could not adapt color variations. In this paper, we propose a new formulation for ACSM.

The energy functions for ACSM are modified from the conventional snake. As shown in Fig. 6, we have designed a snaxel having two components: outside and inside color patch. These components are compared to the original image where each snaxel lays. This makes the feature/object segmentation process faster and more accurate. The newly proposed energy function is as follows:

$$E_{acs} = \sum_{i=1}^n [E_{int}(i) + E_{image}(i) + E_{cs}(i)], \quad (9)$$

$$E_{cs} = - \sum_{k=1}^{n_{p1} + n_{p2}} \{W(p_{i,k}) \cdot D(p_{i,k})\}, \quad (10)$$

where

$$W(p_k) = \begin{cases} [w_1 & w_2] & \text{if } p_k \in P_{in} \\ [w_2 & w_1] & \text{if } p_k \in P_{out} \end{cases}$$

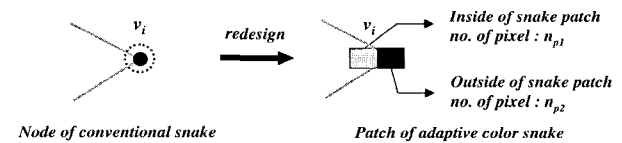


Fig. 6. Design scheme of adaptive color snake model.

$$D(p_k) = \begin{bmatrix} N(z_{p_k}; \bar{z}_{p_i}, \Sigma_1^2) N(\bar{z}_{p_i}; \bar{z}_{p_i, gen}, \Sigma_{1, gen}^2) \\ N(z_{p_k}; \bar{z}_{p_o}, \Sigma_2^2) N(\bar{z}_{p_o}; \bar{z}_{p_o, gen}, \Sigma_{2, gen}^2) \end{bmatrix}.$$

In  $W(\cdot)$ ,  $w_1$  and  $w_2$  are the user-determined weighting factors for each patch of the snake node. And, a color distribution  $N(\cdot)$  can be represented by a Gaussian distribution model as follows:

$$N(z; \bar{z}, \Sigma^2) = \frac{1}{\sqrt{2\pi}\Sigma} \exp\left(-[z - \bar{z}]^T \Sigma^{-2} [z - \bar{z}] / 2\right), (11)$$

where

$$z = [r \quad g]^T, \quad \bar{z} = \frac{1}{n_{p_1} + n_{p_2}} \sum_{i=1}^{n_{p_1} + n_{p_2}} [r_i \quad g_i]^T,$$

$$\Sigma = \begin{bmatrix} \sigma_{rr} & \sigma_{rg} \\ \sigma_{gr} & \sigma_{gg} \end{bmatrix}.$$

and  $r, g$  are the normalized color values.

The proposed energy function is composed of three components. First and second terms of (9) are identical to the corresponding terms in the conventional snake model. The first term is the internal energy, its modification will be mentioned in the next section. Third term is the color-fitting term so that ACSM is capable of shrinking until the contour of snake meets the desired color boundary.

In (10), the distribution of general color  $N(\bar{z}_i; \bar{z}_{i, gen}, \Sigma_{i, gen}^2)$  was multiplied to the distribution of each color, to prevent the color model from moving too far from the general color.

#### 4.2. Color adaptation

Most color-based systems are sensitive to change in viewing environment. In general, the color representation of an object viewed with a color camera is influenced by many parameters: ambient light, object movement etc. There are two groups of researches that address environment changes: adaptation and toleration. The adaptive approach provides an alternative to making a color model useful in a large range. Instead of emphasizing the recovery of the spectral properties of light sources and surfaces that combine to produce the reflected lights, the goal of adaptation is to transform the previously developed color model into the new environment [25]. We have adopted a method to adapt the color model. Based on the identification of the desired color histogram, the modified parameters of the model can be computed according to the following process.

In (11),  $\bar{z}$  is so noisy that we use the adaptation scheme for determining the color value. Each adapted color value is determined as follows:

$$(\hat{r}_k, \hat{g}_k, \hat{s}_k) = \sum_{i=0}^{m-1} (\alpha_{k-i} \bar{r}_{k-i}, \beta_{k-i} \bar{g}_{k-i}, \gamma_{k-i} \Sigma_{k-i}), (12)$$

where  $\alpha$ ,  $\beta$ , and  $\gamma$  are scale factors for updating each color.

#### 4.3. Operation flow of ACSM

The procedure of ACSM is summarized as the following pseudocode.

**begin**

initialize the number of iteration

assign manually general color value  $\bar{z}_{gen}, \Sigma_{gen}$

initialize snake node and set initial color  $\bar{z} = \bar{z}_{gen}$

**while** (not termination condition) **do**

update the iteration number

compute energy function using the greedy method

find and move to the point with minimum energy

calculate  $\bar{z}$  from the color value within patches

update  $\hat{r}_k, \hat{g}_k, \hat{s}_k$

**end**

## 5. OBJECT TRACKING USING CONDENSATION ALGORITHM

In the previous section, a new design scheme for the segmentation was discussed. In this section a new scheme for tracking is presented.

The condensation algorithm [16] has attracted much interest in the field of active vision as it offers a framework for dynamic state estimation, where the underlying probability density function is not required to be Gaussian. The algorithm is based on factored sampling but extended to apply iteratively to successive images in a sequence. As new information becomes available, the posterior distribution of the state variables is updated by recursively propagating these samples.

A new snake energy for tracking, operation modes for the proposed snake, and the application of the condensation algorithm are explained as follows:

#### 5.1. Design of contraction energy

In (2), the second term makes the contraction forces. Yet, this internal energy suffers problems explained in Section 2.3. Therefore, a new formulation is required for the internal energy to overcome those problems.

The contraction energy,  $E_{cont}$ , is proposed on the basis of quadratic function in (13). This function has parabolic characteristics with the focus being determined by the initial snake contour. The estimated focus can be calculated by using the first moment and the area of initial snake.

$$E_{cont}(i) = \beta_i \cdot K_{cont} \{(x_i - \hat{x}_c)^2 + (y_i - \hat{y}_c)^2\}, (13)$$

where  $\beta$  is a constant for  $E_{cont}$ ,  $K_{cont}$  is a energy

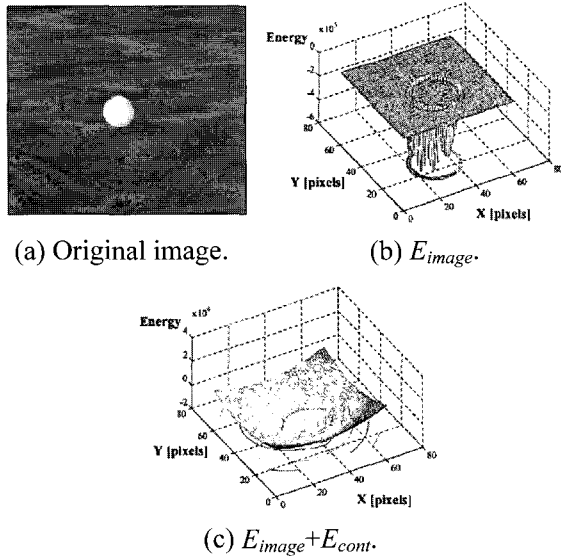


Fig. 7. The concept of contraction energy,  $E_{cont}$ .

constant and  $\hat{x}_c$ ,  $\hat{y}_c$  is the focus which can be estimated by the condensation algorithm. The concept of contraction energy,  $E_{cont}$ , is explained in Fig. 7. The original image can be converted into the image energy,  $E_{image}$ , by (1) as in Fig. 7(b). The sum of energies,  $E_{image} + E_{cont}$ , is shown in Fig. 7(c). The resulting energy has a property that contracts snake's nodes into the estimated focus of the target object along the gradient. Initial snake's nodes may be contracted by the contraction energy  $E_{cont}$ , and along the boundary of the object, the nodes can fall into the valley of the image energy  $E_{image}$ . According to this process, objects can be segmented successfully.

## 5.2. Operation modes

There are two kinds of operation modes in object tracking using ACSM: segmentation mode and jump mode.

**Segmentation mode:** Segmentation is to separate the boundary of the target object from the images. In the initial state, snake has to be laid roughly around the target object. From this initial position, snake will gradually shrink to the boundary of the target object by minimizing snake energy function. In most of the previous works, this initialization of snake had accomplished manually. However, if the residual motion of the moving object can be detected by processing the condensation algorithm for two successive images, the regional information about the moving object can be stochastically obtained. Thus, automatic initialization can be made possible by using this concept.

In this segmentation mode, total snake energy is comprised of three major terms  $E_{cont}$ ,  $E_{image}$ , and  $E_{cs}$ :

$$E_{acs} = \sum_{i=1}^n [E_{cont}(i) + E_{image}(i) + E_{cs}(i)]. \quad (14)$$

**Jump mode:** The jump mode is proposed to track the object's motion when the overlapped area of the object to be tracked doesn't exist between two sequential images [13]. This concept is explained in Fig. 8. This concept is based on the assumption that the position of the object can be obtained by processing the condensation algorithm. Snake's nodes from every previous image are jumped into the boundary of the object, whose location has moved to another region and they are re-initialized on the basis of the radius of the segmented object. Segmentation process and jumping based on the information of the condensation algorithm are repeated through successive images. In this mode total snake energy is comprised of three major terms such as  $E_{cont}$ ,  $E_{image}$ , and  $E_{cs}$ , the same way as the segmentation mode.

The operation flow for the jump mode is explained in Fig. 9. In the initial state, snake's nodes are initialized manually or automatically. Initially, snake contracts into the boundary of the object to be segmented and this segmentation process is continued to the minimal state of the total snake energy for segmentation. In the next frame, the object's location may have moved to another position, which is far from the corresponding position in the previous image. After calculating the condensation algorithm, snake

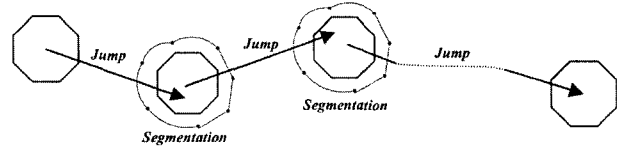


Fig. 8. The concept of jump mode.

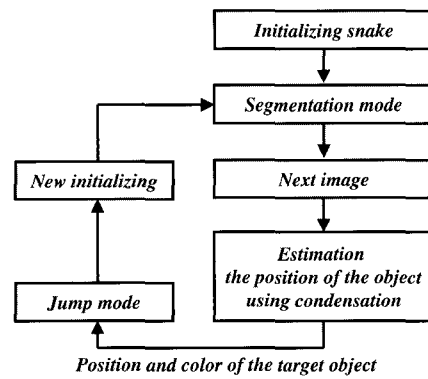


Fig. 9. The operation flow of jump mode.

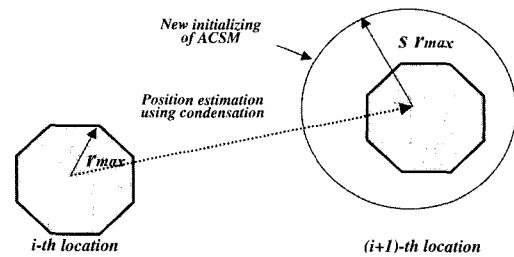


Fig. 10. The concept of re-initializing process.

nodes jump on the basis of the information of the condensation algorithm. After that, the snaxels are re-initialized as in Fig. 10. The boundary of the object in the next frame can be determined by the segmentation process based on the gradient of  $E_{cont}$ . Then, the jump mode system receives the succeeding image for finding the new location of the object. Re-initialization can be done by using the radius information of the object from the previous image. In the  $i$ th image, the maximum radius of the segmented object can be calculated. But this calculation by the condensation algorithm may not be so accurate that in the  $(i+1)$ th image, the calculated center may not coincide with the actual center. Therefore, the new radius for re-initialization has to be determined by the product with a scale factor,  $s$ .

## 6. EXPERIMENTAL RESULTS

### 6.1. System setup

We demonstrated the performance of ACSM by applying it to real image sequences. We have implemented the proposed algorithm on a Pentium-IV 2.4 GHz PC with 512MB RAM, and a Logitech QuickCam-Pro USB camera. Three sequential images were captured at the rate of 10 frames/sec. Each image has the resolution of 320x240 pixels and depth of 24-bit color. For simplicity of comparison, 16 snaxels were used for each snake model. For ACSM, initial color distribution of the object was manually assigned. In most object tracking problems, the appearance properties (especially, color information) of target object are known. In the following experiments, however, the color information for outside patch of ACSM are not known, and therefore the energy for outside patch is not required to be calculated. Thus, only inside energy of  $E_{cs}$  in (10) is used for experiments. From the second frame, newly adapted color is applied to ACSM. The size of snake patches are 5 pixels each, along the same and opposite directions of the estimated center, which is calculated using the condensation algorithm.

The state vector for Condensation algorithm has the following form:

$$x_t = [x_t \ \dot{x}_t \ y_t \ \dot{y}_t \ r_t \ \dot{r}_t \ g_t \ \dot{g}_t]^T, \quad (15)$$

where  $(x, y)$  locates the center of the object,  $r$  and  $g$  are the normalized color of the object. In the experiments, 100 samples were used at each iteration.

### 6.2. Experimental results

Experiments were performed on three cases. In the first case, a fast moving object on a simple background are segmented. Second experiment was carried out to test the proposed algorithm for segmenting a fast moving object on a complex background. In the third experiment, the performance

was tested in the presence of camera motion. To demonstrate the performance, ACSM was compared with a conventional snake, and a snake with optical flow. Through the experiments, the effectiveness of ACSM was verified.

**Case 1: Large motion variance and simple background**

In case 1, the performance of segmentation and tracking in the case of a fast moving object on simple background was evaluated. This experiment was implemented in order to demonstrate that the motion information is essential for object tracking using active contours. In figure, yellow dots are the final snake points.

Fig. 11 shows the results of conventional snake. A conventional snake was applied to object tracking without motion estimation such as optical flow or condensation algorithm. Since an object in this experiment moves fast, contour in the previous frame is not suitable for the initial contour of the present frame. It causes a mismatch between the object and the estimated contour.

Fig. 12 shows the results of the conventional snake with motion information using an optical flow. To

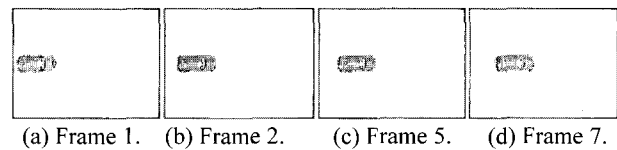


Fig. 11. Experimental results for conventional snake in case 1.

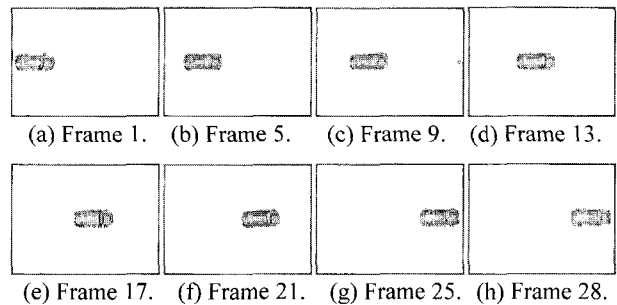


Fig. 12. Experimental results for conventional snake with optical flow in case 1.

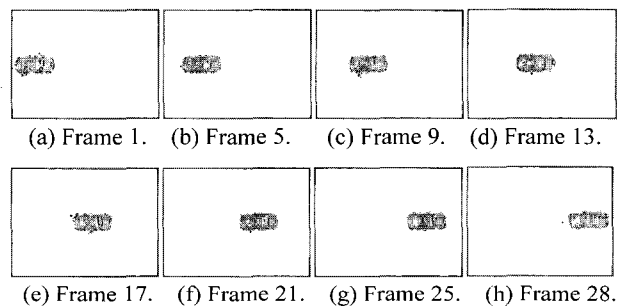


Fig. 13. Experimental results for ACSM in case 1.

guarantee the tracking in the presence of large motion, the estimation of the object motion using optical flow was applied to the conventional snake. Compared to result of the conventional snake, object tracking is guaranteed as shown in Fig. 12. But, there still exists a problem as mentioned in Sec. 2.1. In frame 13 and 21, in the presence of illumination change, approximation errors and mismatch of snake around the target object are visible. In this experiment, as the background is very simple, the effect on this problem is not serious.

Fig. 13 presents the results of ACSM. In this figure, blue dots are the samples by the condensation algorithm. From frame 1 to frame 28, The condensation algorithm tracks an object. Since ACSM considers the color information of the object stochastically, ACSM segments the object more accurately as shown in frame 13 and 21. Through all frames in a sequence, illumination condition was slightly changing. Though the change of illumination, color adaptation ability of ACSM has overcome the problem.

**Case 2:** Large motion variance and complex background

The performance of segmentation and tracking in the case of a fast moving object on a complex background is evaluated in case 2. In case 1, we have showed the effectiveness for using motion information. Experiments of case 2 showed that the color information is also essential for object tracking using active contours.

As shown in Figs. 14 and 15, the target object moves across other objects, which are of different colors. Background objects with black color are used, because color difference between background object and the floor is larger than that between target object and the floor, or between target object and background objects. In this experiment, target object moves forward and rotates considerably.

In general, optical flow-based approach to object tracking and segmentation needs the following assumptions [13].

- Object has a rigid body motion, and mainly translational motion.
- No pure rotation at one point.
- No divergence existence in the image plane.

The above assumptions are needed to filter noisy components in processing of optical flow. In the image sequences, there are many noise factors such as sensor noise, illumination changes, and so on. Though these factors are not real motion information, optical flow may consider them as real motion information. This interest on erroneous information may take more calculation time. Consideration on calculation time will be discussed in Sec. 6.3.

Fig. 14 shows the results of conventional snake with motion information using optical flow. From frame 1 to 5, conventional snake works well. But,

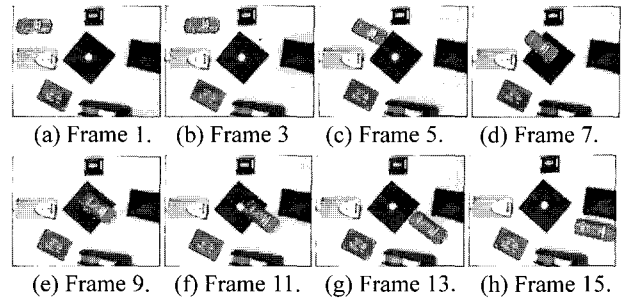


Fig. 14. Experimental results for conventional snake with optical flow in case 2.

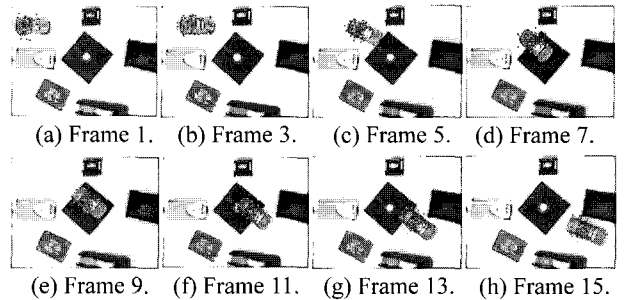


Fig. 15. Experimental results for ACSM in case 2.

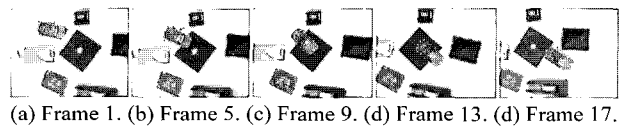


Fig. 16. Experimental results for ACSM in case 3.

target object encounters background objects with larger edge intensity from frame 6. Since edge intensity of the background object is larger than that of the target object, snaxels tend to be attracted to the boundary of the background object.

Fig. 15 presents the results of ACSM. In contrast to results of the conventional snake, ACSM is capable of tracking an object more accurately, since color information of the object is also used. Though the background object with high edge intensity is located near the target object,  $E_{cs}$  in (10) attracts snaxels to the object's boundary.

**Case 3:** With camera motion

Third experiment was carried out in order to show the capability of ACSM in real-time tracking while the camera is in motion. The computation of the optical flow field for the entire area of interest requires considerable computational complexity. Thus, it may be effective only for a static camera. As the camera moves, the system generates considerable variation between two successive images. Therefore, optical flow-based approaches to real-time tracking system need longer calculation time. Therefore, in case 3, conventional snake with optical flow fails to operate in real-time at frame rate of 10 frame/sec as shown Table 2.

We have tested the effectiveness of ACSM on a



moving camera system. We have implemented the proposed algorithm on a typical pan-tilt camera system. Tracking and segmentation was performed from frame 1 to 17. Fig. 16 presents the results of ACSM. With the color information of the object, ACSM has accurately segmented the object. The condensation algorithm has not been affected by the camera motion. It only considers the probabilistic distribution of desired color information.

### 6.3. Discussion

In this section, two considerations are described to evaluate the performance of ACSM compared to other approaches. Those are segmentation error and calculation time. In real-time application, those are the most important factors.

#### 6.3.1 Remarks on segmentation error

For comparing with other approaches, two kinds of error measure are accepted as measure for segmentation error. Following measures are the gauge to the accuracy of segmentation. First error measure is given as follows:

$$E_{SCB} = \frac{SCB(n)}{n}, \quad (16)$$

where  $n$  is the number of snaxels, and  $SCB(\cdot)$  refers to the number of snaxels on the correct boundary. This measure converges to 100% when correct segmentation is done. Second segmentation error [24] is given by

$$E_{card} = \frac{card(T \cap B^c) + card(T^c \cap B)}{card(T)}, \quad (17)$$

where  $T$  is the target object mask,  $B$  is the background object mask, and  $card(\cdot)$  is the cardinality (i.e., number of pixels) of a set.

Table 1 shows the comparison between conventional snake and ACSM. In order to compare on the same condition, condensation algorithm for tracking method and 50 nodes for snaxels were used to evaluate measures on each frame. And, same initial positions for snaxels were used at each frame. Table 1 says that ACSM is less disturbed than the conventional method, in the presence of background

Table 1. Results of segmentation error.

	Segmentation method	$E_{SCB}$ (%)	$E_{card}$ (%)
Case 1	Conventional Snake	97.32	3.43
	ACSM	100.00	0.00
Case 2	Conventional Snake	70.70	69.31
	ACSM	98.58	3.29
Case 3	Conventional Snake	63.02	115.85
	ACSM	98.23	2.94

Table 2. Results of calculation time.

	Segmentation method	Tracking method	Time (ms/frame)
Case 1	CS	OF	86
	ACSM	OF	98
	ACSM	CA	63
Case 2	ACSM	OF	157
	ACSM	CA	89
Case 3	ACSM	OF	373
	ACSM	CA	94

\* CS: conventional snake, OF: optical flow, CA: condensation

object with high edge intensity and camera motion.

#### 6.3.2 Remarks on calculation time

In real-time systems, calculation time for the proposed algorithm is of great importance. The computational burden in the real-time implementation of the algorithm for object tracking system is the motion estimation. For the better performance of tracking, several advanced algorithms for motion estimation has been employed in the snake model. Optical flow algorithm has been applied to estimate the object's motion. But, the computation of the optical flow field for the entire area of interest requires considerable computational complexity. It may be only effective for a static camera. As the camera moves, the system generates a considerable variation between two successive images. In presence of camera motion, Condensation algorithm shows the *robust tracking of object's motion*.

It is hard to compare the performance between the proposed algorithm and other approaches directly. The proposed system is designed to operate at the frame rate of 10 frame/sec. The same parameters of the snake such as the number of node, greedy search space in each iteration is used for conventional snake. Table 2 shows the average calculation time for processing each frame. Tests with conventional snake only in case 2 and 3 are included, because it doesn't satisfy the segmentation performance in case 1. This table includes the comparison between tracking methods: optical flow and condensation algorithm. As shown in case 2 and 3, in the case of considerably rotating object or camera motion, calculation time for extracting motion vector is too large to operate in real-time. But, ACSM with condensation algorithm is suitable for real-time application.

## 7. CONCLUSION

In this paper, we have introduced a real-time object tracking scheme using a new snake model, called "*adaptive color snake model (ACSM)*" which can be applied to color image segmentation and object tracking. Through experiments, the effectiveness of the proposed algorithm was verified.

Considerable works have been done during the past decade in tracking and motion analysis of objects in the context of snake models. Active contour models have been developed as useful tools for segmenting and tracking objects. Snake provides high-level information related to the contour shape and image features. These image features are usually based on intensity edges. When using intensity edges as image features, the image segmentation results obtained by use of snake may be negatively affected by the disturbances (e.g. shadows, shading and neighboring objects). Because intensity-based snake cannot distinguish between various transition types, our attention is directed towards the use of color and motion information. This consideration provides more accurate results compared to conventional algorithms, in which objects are tracked by exploiting motion information; this is more evident in sequences with complicated content, where there are fast motion, complex background, special camera effects (zooming, pan-tilting), etc.

When active contour model is applied under above conditions, advanced motion estimation algorithm should also be used for better performance. Usually, optical flow method is used for motion estimation. But, as shown in our experiments, it takes much calculation time in sequences where complex background and camera motion exist. Especially, in real-time applications, calculation time as well as segmentation performance are the most important factors.

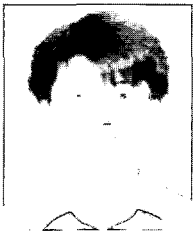
Experimental results show the potential of the proposed ACSM and tracking scheme. Despite the fast motion, complex background, special camera effects, ACSM can also operate in real time. Therefore, proposed tracking scheme is more suitable than intensity-based or optical flow-based approaches.

## REFERENCES

- [1] M. Kass, A. Witkin, and D. Terzopoulos, "Snake: Active contour models," *Int. J. Computer Vision*, vol. 1, pp. 321-331, 1988.
- [2] D. Terzopoulos and K. Waters, "Analysis and synthesis of facial image sequences using physical and anatomical models," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 15, no. 6, pp. 569-579, 1993.
- [3] D. DeCarlo and D. Metaxas, "The integration of optical flow and deformable models: Applications to human face shape and motion estimation," *Proc. of IEEE Computer Vision and Pattern Recognition*, pp. 231-238, 1996.
- [4] C. Kervrann and F. Heitz, "Robust tracking of stochastic deformable models in long image sequences," *Proc. Int'l Conf. Image Processing*, vol. 3, pp. 88-92, 1994.
- [5] A. Blake and A. Yuille, *Active Vision*, MIT Press, Cambridge, 1992.
- [6] T. Gevers, S. Ghebreab, and A. W. M. Smeulders, "Color invariant snakes," *Proc. of Ninth British Machine Vision Conf.*, pp. 659-670, 1998.
- [7] S. Lefevre, J. P. Gerard, A. Piron, and N. Vincent, "An extended snake model for real-time multiple object tracking," *Proc. of Int'l Workshop on Advanced Concepts for Intelligent Vision Systems*, pp. 268-275, 2002.
- [8] G. Sapiro, "Color snakes," *Computer Vision and Image Understanding*, vol. 68, no. 2, pp. 247-253, 1997.
- [9] B. Bascle and R. Deriche, "Region tracking through image sequences," *Proc. of Fifth IEEE Int'l Conf. Computer Vision*, pp. 302-307, 1995.
- [10] P. Fieguth and D. Terzopoulos, "Color-based tracking of heads and other mobile objects at video frame rates," *Proc. of IEEE Conf. Computer Vision Pattern Recognition*, pp. 21-27, 1997.
- [11] B. Heisele, U. Krebel, and W. Ritter, "Tracking nonrigid, moving objects based on color cluster flow," *Proc. of IEEE Conf. Computer Vision Pattern Recognition*, pp. 257-260, 1997.
- [12] Y. Fu, A. T. Erdem, and A. M. Tekalp, "Tracking visible boundary of objects using occlusion adaptive motion snake," *IEEE Trans. on Image Processing*, vol. 9, no. 12, pp. 2051-2060, 2000.
- [13] W. Kim, C. Lee, and J. Lee, "Tracking moving object using snake's jump based on image flow," *Mechatronics*, vol. 11, pp. 199-226, 2001.
- [14] N. Peterfreund, "Robust tracking of position and velocity with kalman snakes," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 21, no. 6, pp. 564-569, 1999.
- [15] M. Sonka, V. Hlavac, and R. Boyle, *Image Processing, Analysis, and Machine Vision*, 2nd edition, PWS Publishing, 1998.
- [16] M. Isard and A. Blake, "Condensation-Conditional density propagation for visual tracking," *International Journal of Computer Vision*, vol. 29, no. 1, pp. 5-28, 1998.
- [17] G. Jang and I. Kweon, "Robust object tracking using an adaptive color model," *Proc. of IEEE Int'l Conf. Robotics & Automation*, vol. 2, pp. 1677-1682, 2001.
- [18] E. B. Meier and F. Ade, "Tracking multiple objects using the condensation algorithm," *Robotics and Autonomous Systems*, vol. 34, pp. 93-105, 2001.
- [19] A. Amini, T. Weymouth, and R. Jain, "Using dynamic programming for solving variational problems in vision," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 12, pp. 855-867, 1990.
- [20] M. Kirby and L. Sirovich, "Application of the Karhunen-Loeve procedure for the characteri-

zation of human faces," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 16, pp. 689-700, 1994.

- [21] K. Seo, W. Kim, C. Oh, and J. Lee, "Face detection and facial feature extraction using color snake," *Proc. of the 2002 IEEE International Symposium on Industrial Electronics*, vol. 2, pp. 457-462, 2002.
- [22] K. Seo and J. Lee, "Object tracking using adaptive color snake model," *Proc. of IEEE/ASME International Conference on Advanced Intelligent Mechatronics*, vol. 2, pp. 1406-1410, 2003.
- [23] K. K. Sung and T. Poggio, "Example-based learning for view-based human face detection," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 2, pp. 39-51, 1998.
- [24] A. Doulamis, N. Doulamis, K. Ntalianis, and S. Kollias, "An efficient fully unsupervised video object segmentation scheme using an adaptive neural-network classifier architecture," *IEEE Trans. on Neural Network*, vol. 14, no. 3, pp. 616-630, 2003.



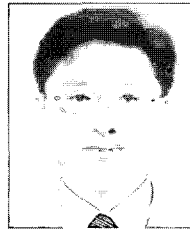
**Kap-Ho Seo** was born in Pusan, Korea, in 1974. He received the B.S. degree from Korea University, Korea, and the M.S. degree in Electrical Engineering from Korea Advanced Institute of Science and Technology (KAIST), Korea in 1999 and 2001, respectively. He is currently working toward a Ph.D. degree in KAIST. His current research

interests include neural network, vision-based mobile robot, and rehabilitation robot system.



**Jin-Ho Shin** received the B.S. degree in Electronic Engineering from Hanyang University, Seoul, Korea, in 1991 and the M.S. and the Ph.D. degrees in Electrical Engineering from KAIST, Daejeon, Korea, in 1993 and 1999, respectively. From 2000 to 2002, he was a Postdoctoral Fellow at the Department of Mechano-Informatics,

The University of Tokyo, Tokyo, Japan. Since 2002, he has been with the Department of Mechatronics Engineering, Dong-Eui University, Busan, Korea, where he is currently an Assistant Professor. His research interests include robust adaptive control and intelligent control of robotic systems, control theory and applications, object motion tracking, sensing and vision systems, and intelligent robotics.



**Won Kim** was born in Seoul, Korea in 1967. He received the B.S. degree in Electronic Engineering from Hankook Aviation University in 1990 and M.S. degree from Electrical & Electronic Engineering of KAIST in 1999. He worked as a Researcher from 1990 to 1997 in the field of system engineering at Agency for Defense Development, Taejeon, Korea. He is now pursuing a Ph.D. in electrical & electronic engineering of KAIST. His research interests include visual tracking, design technique of active contours and motion estimation.



**Ju-Jang Lee** (M'86-SM'98) was born in Seoul, Korea, in 1948. He received the B.S. and M.S. degrees, both in Electrical Engineering, from Seoul National University in 1973 and 1977, respectively, and the Ph.D. degree in Electrical Engineering from the University of Wisconsin in 1984. From 1977 to 1978, he was a Research

Engineer at the Korean Electric Research and Testing Institute, Seoul. From 1978 to 1979, he was a Design and Processing Engineer at G. T. E. Automatic Electric Co., Waukesha, WI. For a brief period in 1983, he was the Project Engineer for the Research and Development Department of the Wisconsin Electric Power Co., Milwaukee, WI. He joined the Department of Electrical Engineering, Korea Advanced Institute of Science and Technology (KAIST), Daejeon, Korea, in 1984, where he is currently a Professor. In 1987, he was a Visiting Professor at the Robotics Laboratory of Imperial College Science and Technology, London, U.K. From 1991 to 1992, he was a Visiting Scientist at the Robotics of Carnegie Mellon University, Pittsburgh, PA. His research interests are in the areas of intelligent control of mobile robots, service robotics for the disabled, space robotics, evolutionary computation, variable structure control, chaotic control systems, electronic control units for automobiles, and power system stabilizers. Dr. Lee is a Senior Member of IEEE and a Member of the IEEE Robotics and Automation Society, the IEEE Evolutionary Computation Society, the IEEE Industrial Electronics Society, KIEE, KITE, and KISS. He is also a President of ICASE and a Counselor of SICE in Japan.