Fast Reference Region Adjustment Using Sizing Factor Generation in Correlation-Based Image Tracking

Si-Hun Sung and Sung-Il Chien

Abstract

When size and shape of moving object have been changed, a correlator often accumulates walk-off error. A success of correlation-based tracking largely depends on choosing suitable window size and position and thus transferring the proper reference image to the next frame. For this, we propose the Adaptive Window Algorithm with Four-Direction Sizing Factors (AWA-FSF) for fast adjusting a reference region to enhance reliability of correlation-based image tracking in complex cluttered environments. Since the AWA-FSF is capable of adjusting a reference image size more rapidly and properly, we can minimize the influence of complex background and clutter. In addition, we can finely tune the center point of the reference image repeatedly after main tracking process. Thus we have increased stability and reliability of correlation-based image tracking. We tested performance of the AWA-FSF using 45 real image sequences made of over 3400 images and had the satisfied results for most of them.

I. Introduction

Correlation-based tracking [1-4] acquires a reference image from the previous frame and looks for the most similar area to the reference image for estimating the object position in the current frame. Although the correlator is said to be robust against the cluttered noise, its real application has some limitations, too. Usually, it is desirable that searching area should be chosen to be small due to its large computation involved. Another one is its tendency to accumulate walk-off error especially when the object of interest is changing in size, shape, or orientation from frame to frame. Actually, the human operator can not exactly initialize a tracking window to the boundary of a target but more largely than a target size in the field. It is one of causes that a correlation-based tracking increases a walk-off error.

If walk-off error accumulation proceeds beyond a certain critical point, correlation-based tracking can be unsuccessful. It is quite important that the size and position of a window should be precisely determined to guarantee that a proper reference image could be transferred to the next frame. For increasing correlation reliability, correlation-based tracking is usually expected to have high occupancy rate of an object, which means that the window encloses the object properly. It is quite desirable that the window can adjust its size as circumstances of the object have been under change.

Concept of an adaptive window could be found in stereo matching applications, too. The disparity boundaries are sharp for the smaller window but the computed disparity becomes noisy. The larger window means that the computed disparity becomes smoother but the disparity boundaries can be blurred. So, noise and boundary sharpness have a relationship of trade-off. Lotti and Giraudon [5] made four non-centered adaptive windows associated to each image point in the thresholded edge image. Kanade and Okutomi [6] determined the adaptive window size using local intensity and disparity patterns to minimize uncertainty in the disparity computed at each point.

To automatically adapt the reference area, Hughes and Moy [3] designed edge walking algorithm for the boundary detection to operate on a segmented binary image. This algorithm scans the binary image in a raster fashion looking for illuminated pixels. Once an illuminated pixel has been found, the edge walking algorithm searches for further illuminated pixels connected to the initial pixel. They used the boundary detection for estimating the size of an object and thus the size of window that will enclose the object.

Manuscript received October 15, 1997; accepted January 13, 1998.

S.H.Sung and S.I.Chien are Department of Electronic Engineering, Kyungpook National University.

Similarly, Montera et al. [1] determined an object region though expanding from the inner point of an object to outer in the image. To yield the boundary of the object, they searched for the areas where pixel values vary from above the threshold to below the threshold. However, we expect that both methods are difficult to be applied to tracking non-homogeneous cluttered background and large objects sometimes having internal edges.

Adaptive window without a proper sizing magnitude can hardly accommodate itself to environment variations when a window size is much larger or smaller than an object size or an object size is abruptly changed. To adjust a window size more rapidly and efficiently, this paper proposed the Adaptive Window Algorithm with Four-Direction Sizing Factors (AWA-FSF). The AWA-FSF defines eight districts: four side districts and four corner districts, and determines four window sizing directions and four sizing factors using the extracted mutual information obtained from the relation of a corner district and a side district.

The proposed AWA-FSF has been adapted to our modified correlation method intended for accomplishing image tracking in the highly cluttered environments, which we call the Centroid-Compensated Correlator (CCC). Here, we append the centroid method to finely tune the result of the correlation when clutter components as well as object components are simultaneously contained in the tracking window and centroid compensation moves the matched point further to the location of an object which occupies usually larger area than the clutter components inside the window we consider. Additionally, as basic tracking image set, we adopt the use of the Positive Difference of Edges (PDOE) [7] modified from a conventional difference procedure using the absolute value. As we know, some edge operators can extract valuable information useful for successful tracking. We employed the Robinson 3-Level masks to generate edge images used in our tracking. The detailed description of the CCC and the PDOE is beyond the viewpoint of this paper and thus we briefly introduce the CCC and the PDOE in section 2. In section 3, we present the structure and the procedure of the proposed AWA-FSF. Then in section 4 we provide experimental results that clearly demonstrate the advantage of the AWA-FSF. Finally, section 5 gives a summary along with discussion.

II. Applied Image Tracker Architecture

The image tracking block diagram we propose is described in Fig. 1. The overall system is largely divided into the PDOE, the CCC, the AWA-FSF, and a recursive updating loop. First, we acquire the background-reduced image using the PDOE as tracking feature and then track the object by applying the CCC. Finally, the AWA-FSF generates a reference image region with tightly enclosing the object for tracking in the next frame. For the consecutive tracking the size and position of the reference image will be suitably updated.

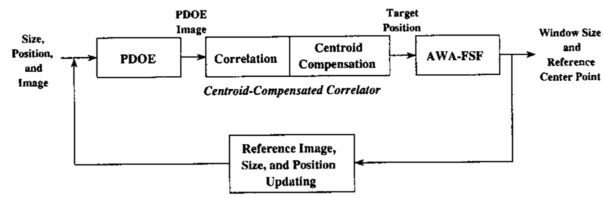


Fig. 1. Overall tracking block diagram.

1. *Positive Difference of Edges (PDOE) for Feature Extraction*

1) *Edge Finding Procedure of the PDOE*

There are many kinds of edge operator such as Sobel, Prewitt, Laplacian, and so on [8]. The Sobel operator usually produces many minute edge components, but in case of image tracking, the minute edge found in background can bring undesirable effects. We thus selected the Robinson 3-Level operator that produces little minute edge components and its local averaging tends to reduce also the effects of noise. We can further reduce computation time through implementing only with summation.

2) *Difference Procedure of the PDOE*

The PDOE conceptually has spatial and temporal motion components as shown in Fig. 2. We can express the PDOE image in the current frame such that

$$DE_n(i,j) = \begin{cases} E_n(i,j) - E_{n-1}(i,j), & \text{if } E_n(i,j) > E_{n-1}(i,j) \\ 0, & \text{Otherwise} \end{cases} \qquad \textbf{(1)}$$

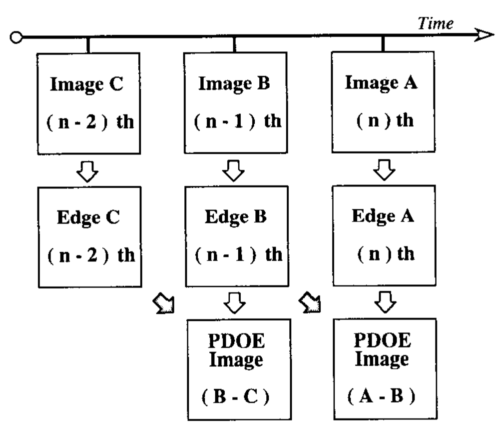


Fig. 2. Block diagram of the PDOE.

where $E_n(i,j)$ is an edge image at point (i,j) in the current frame and $E_{n-1}(i,j)$ is one in the previous frame. The PDOE removes negative components from the difference of two edge images. As shown in Fig. 3, the PDOE can detect motion components much better than the difference method just using an absolute value.

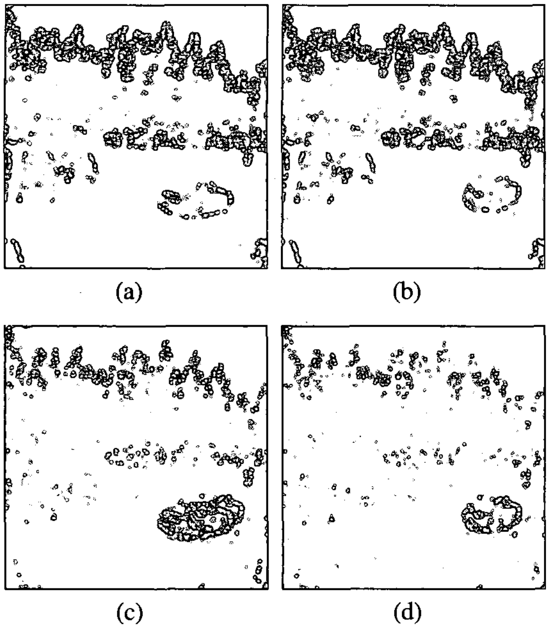


Fig. 3. Image information extracting using two real images in parking lot: (a) Edge image in the previous frame, (b) Edge image in the current frame, (c) Absolute difference of edge images, and (d) PDOE image.

2. Centroid-Compensated Correlator (CCC)

The CCC is a robust image tracker against complex cluttered environment. Correlation-based tracking is a method that looks for the most similar area to the reference image acquired from the previous frame for estimating the object position in the current frame but often accumulates the walk-off error. To compensate the error of correlation-based tracking, we sequentially append the centroid method to refine output of correlation.

In case that clutter components are partially included in a reference image, the correlation peak position might be deviated from the center of the object to the clutter part and this is found to be harmful for the correlator. However, the clutter components are usually weak and occupy small region inside the matched window determined by a previous correlator. Thus the centroid position calculated from the window region can be correctly shifted toward the center of object that might not be located at the center of the window.

The CCC is found to be quite useful for overcoming strong cluttered environments and occlusion effects and to prevent the tracker from accumulating the walk-off error severely. Meanwhile, the time consumption added due to the centroid compensation is not so prohibitive because the centroid method is only applied to the small region already decided by the correlator.

III. Adaptive Window Algorithm with Four-Direction Sizing Factors (AWA-FSF)

Fig. 4 describes an overall block diagram of our proposed AWA-FSF architecture. A feature set used for adjusting an adaptive window is the PDOE image previously described. The AWA-FSF needs mainly several steps: the adaptive window setting, the direction of sizing estimation in corner and side, the sizing factor determination, and the window size determination and the reference center point relocation. Here, the adaptive window can expand or shrink independently in four sizing directions, each side having a sizing magnitude derived from each sizing factor. In Fig. 4, we present several sizing parameters: DS_S is a direction of sizing in the side, DS_C is a direction of sizing in the corner, and DS_{CS} is a direction of sizing obtained from converting a corner effect to the corresponding side direction. Additionally, $\triangle SF$ indicates a sizing factor that will be used for determining final window size change in pixel unit.

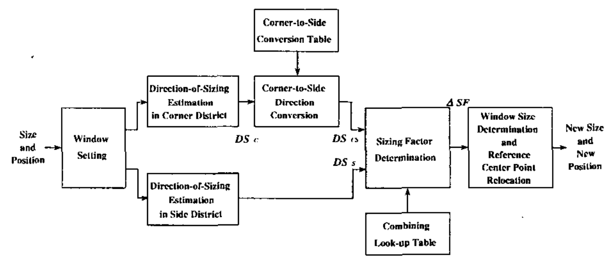


Fig. 4. Overall block diagram of the AWA-FSF.

1. Setting Adaptive Window

For adaptively controlling window size and relocating the center point, we now construct three regions: inner region, middle region, and outer region. First, we define the outside boundary of the middle region as the previous window given by the previous Centroid-Compensated Correlator. We design the inner region needed for extracting information within an object boundary and the extended outer region for obtaining useful clues about background information. The extracted information from a middle region will be used as criterion for determining whether pixels in the middle region are part of the object or not. Thus in order to extract more accurate information near the object boundary, the area of the middle region is fixed to be smaller than those of the others.

Second, eight overlapped districts consisting of four side districts and four corner districts are then defined. The direction of sizing in a side, DS_S, will act as a dominant parameter in finally determining a sizing direction of the window. A corner district will evaluate the edge distribution to provide a relevant corner sizing direction, which will be decomposed to the two corresponding side directions. For this, its area is designed to be similar to that of a side district. Finally, each district has been divided into three sub-districts: inner zone (I), middle zone (M), and outer zone (O), which are not overlapped. Actually, the statistical information from these zones will be used to identify various situations leading to the determination of the suitable directions of sizing, which will be detailed in the following sections. Fig. 5 describes a layout of the left side district of four side districts and a layout of the top right corner district of four corner districts. The remaining side and corner districts will be defined similarly.

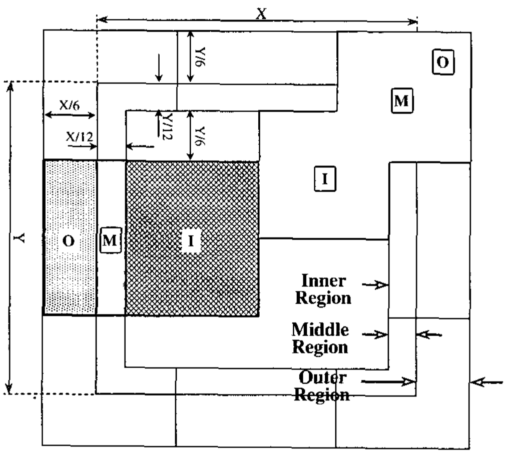


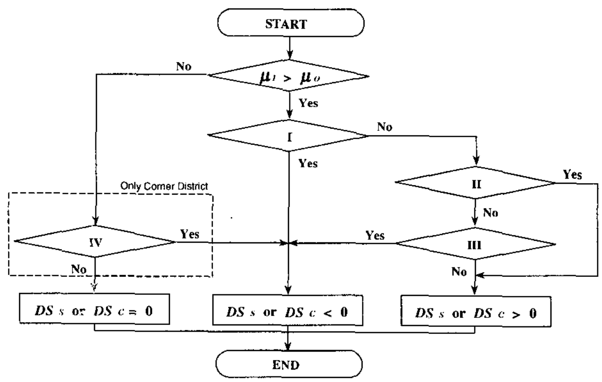
Fig. 5. The layout of the left side district and that of the top right corner district.

2. *Determining Direction of Sizing*

Here, we should estimate a direction of sizing by comparing the extracted statistical information such as means and variances from three zones in each district. A direction of sizing is not the permanent direction but the temporary direction for finally generating the sizing factor $\triangle SF$. The detailed flow chart of determining direction of sizing in a district is shown in Fig. 6, where μ_I, μ_M, and μ_0 are means of gray levels in inner zone, middle zone, and outer zone. σ_I^2, σ_M^2 , and σ_O^2 are variances of those zones. A positive direction of sizing denotes that the window size should be increased and a negative direction of sizing denotes that the window size should be decreased. The determining procedure of sizing magnitude will be detailed later in section 3.4. The AWA-FSF can perform independently for the corner and the side in adjusting a window size.

We heuristically have made four conditions to control the direction of sizing. Condition I represents the situation of decreasing a window size in which an absolute difference between a mean of inner zone and that of middle zone is larger than an absolute difference between a mean of middle zone and that of outer zone and Condition II represents the opposite situation. In addition, for reducing the boundary sensitivity around the middle zone, we set a marginal range α as 20% of μ_M in Conditions I and II. Condition III plays a role in providing an additional information when the mean value is insufficient to determine a direction of sizing. In the side district, we found that they do not need the marginal range. At later stage, DS_S will dominantly determine a window direction of sizing at later stage and thus we append Condition IV for fine adjustment.

Two corner districts select a sharing side as a sizable region only when the neighbor corner districts have the same signs, that is, DS_Cs. Table 1 is the corner-to-side conversion table which determines the sizable sides by the corner districts. On the contrary, a side district determines a direction of sizing DS_S without considering DS_Ss of neighboring side districts.



Conditions :

$$\text{I} : |\mu_M - \mu_I| > |\mu_M - \mu_O| + \alpha\mu_M$$

$$\text{II} : |\mu_M - \mu_O| > |\mu_M - \mu_I| + \alpha\mu_M$$

$$\text{III} : \sigma_I^2 > \sigma_0^2 \text{ or } \sigma_m^2 > \sigma_0^2$$

$$\text{IV} : \sigma_1^2 > \sigma_0^2 \geq \sigma_M^2$$

Fig. 6. Flow chart of determining direction of sizing.

3. *Determining Sizing Factor*

The AWA-FSF should act adaptively for size variation of a moving object in the complex circumstances. In order to perform sizing more efficiently, a sizing factor that will determine the final window size in pixel unit by also considering the size of the previous window, which will be

Table 1. Corner-to-side conversion table. We choose a DS_{CS} by the corner districts with respect to mutual relation with two neighbor corner districts only when their DS_Cs are the same signs. '-' denotes 'don't care'.

		Top Left	Top Right	Bottom Left	Bottom Right
Location of Corner District	Top Left	-	Top Side	Left Side	-
	Top Right	Top Side	-	-	Right Side
	Bottom Left	Left Side	-	-	Bottom Side
	Bottom Right	-	Right Side	Bottom Side	-

detailed in section 3.4, is estimated here for each side of the window.

1) Based on the Window Sizing Direction

Determining a sizing factor also considers the directions of window sizing. If the window fails to enclose the whole part of an object by lavishly reducing its size, the tracker might lose some valuable information and this leads to quite undesirable situation. On the other hand, increasing a window size is more tolerant against sizing error, since the window still encloses a whole object information. This consideration is reflected on Table 2. The magnitude of positive $\triangle SF$ representing an increasing window is fixed to be twice as large as that of negative $\triangle SF$ representing a decreasing window. This fact means that our window system is designed to be generous in expanding, but somewhat cautious in shrinking.

2) Based on Mutual Relation with Neighbor Districts

We have to consider the mutual relation between DS_{CS} which is the converted direction of sizing from corner information and DS_S which is the direction of sizing only from side information. If potentially determined directions of sizing differ from each other, we assume that the window boundary is located on the border between the object and background components. We then conclude the DS_S as a window sizing direction by ignoring the result of a corner district, since we found that an object exists more frequently in a side district rather than in a corner district. Meanwhile, if the DS_{CS} potentially determined by the two neighbor DS_C s is the same as the DS_S, we assume that an object size is much larger than the window size or is much smaller than the window size. To rapidly adjust a window size in case of the same directions of sizing, the sizing factor is generated twice as large as that in the other cases. Table 2 is the combining look-up table for determining the sizing factor based on mutual relation with the DS_S and the DS_{CS} and also on inward and outward weighting discussed in previous section 3.3.1.

Table 2. Combining look-up table. Sizing factor determination is based on mutual relation with the DS_S of a side district and the DS_{CS} obtained from the two neighboring corner districts.

Direction-of-Sizing By Side District	Direction-of-Sizing By Neighbor Corner Districts	Sizing Factor $\triangle SF$
$DS_S < 0$	$DS_{CS} > 0$	-1
$DS_S < 0$	$DS_{CS} < 0$	-2
$DS_S > 0$	$DS_{CS} < 0$	+2
$DS_S > 0$	$DS_{CS} > 0$	+4

4. Evaluating Sizing Magnitude with Sizing Factor

Here, we also consider the size of the previous window, since it is desirable that the final sizing magnitude should be to some degree proportional to the size of the previous window. First, we determine that the basic sizing unit S_{B_k} is given by

$$S_{B_k} = \max(1, \frac{S_k}{50}) \quad \text{for } k = V, H \tag{2}$$

where S_k shown in Fig. 7 is a vertical (V) or horizontal (H) size of a window size in the previous frame and we selected fifty pixels as a size threshold for a small object heuristically. We now put the center of coordinates at the center of window. Then we eventually evaluate a sizing magnitude $\triangle S_i$ for varying the window size in each window sizing direction of top (T), bottom (B), left (L), and right (R) side. $\triangle S_i$ now defined in pixel unit is evaluated by

$$\Delta S_i = \Delta SF_i \times S_{B_k} \quad \text{for } i = T, B, L, R; \quad \text{for } k = V, H \tag{3}$$

where

$$\triangle SF_i \in \{-2, -1, +2, +4\} \tag{4}$$

Finally, we can obtain new window center point and new window sizes in four side directions from the center point. New window positions S_T', S_B', S_L', and from a center point are

$$S_i' = S_i + \mathrm{sgn}(S_i) \times \Delta S_i \quad \text{for } i = T, B, L, R \tag{5}$$

where

$$\mathrm{sgn}(x) = \begin{cases} +1, & \text{if } x > 0 \\ -1, & \text{if } x < 0 \end{cases} \tag{6}$$

and S_t is the coordinate value of the window position before the window sizing procedure. Thus we can find the finely tuned new window center point for acquiring the reference image that will be used for the next frame.

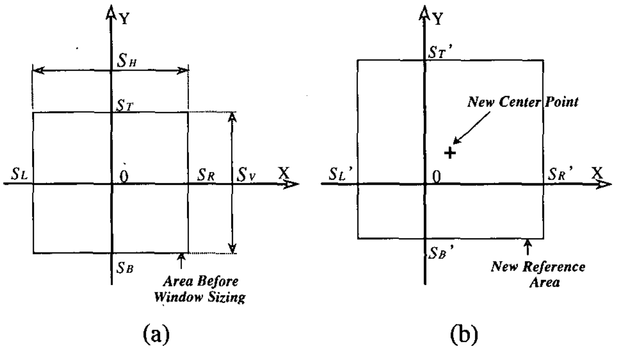


Fig. 7. Reference window boundary and center point relocation for the more finely tuned reference image: (a) Before window sizing procedure and (b) After window sizing procedure by the AWA-FSF.

5. Conceptual Example of the Proposed AWA-FSF

For easy understanding, we explain a conceptual example of the AWA-FSF operation in Fig. 8. First, we determine each direction of sizing from four side districts and four corner districts. The dotted arrows show directions of sizing in each district. Fig. 8(a) shows that a corner district determines the DS_C without considering the results of neighbor corner districts. Actually, we show the DS_{CS} contributed from corner districts to the relevant side directions in Fig. 8(b). For instance, the DS_C of the top left corner district is the same as that of the top right corner district and we can conclude that the window should be increased upward for the top side.

In the meantime, one side district does not care about the results of the others. The DS_S by each side district is now available as shown in Fig. 8(c). Then we determine the window sizing factors using directions of sizing generated by side districts and by neighbor corner districts based on Table 2. We obtain the window sizing directions as well as the sizing factors in four sides, as is described in Fig 8(d). In this example, sizing factors are +4 on the top side, -2 on the bottom side, -1 on the left side, and +2 on the right side. The center point is relocated to a new position as in Fig. 8(e). In this way, the AWA-FSF can correct the walk-off error introduced by a tracking process and provide a new window that follows and encloses an object efficiently.

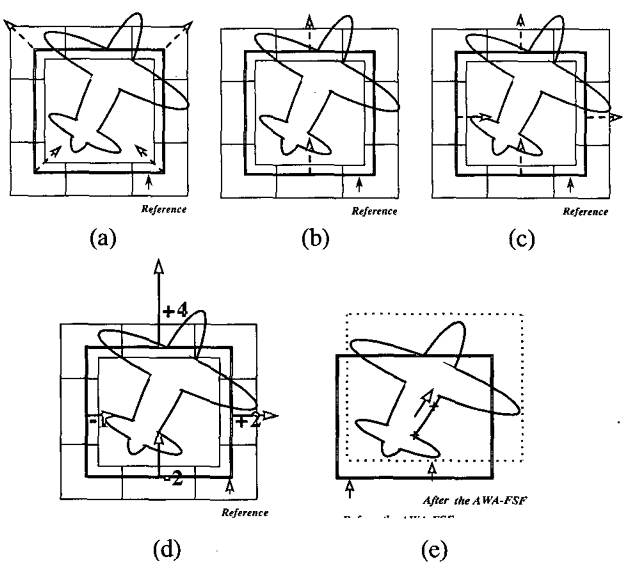


Fig. 8. Conceptual example of the proposed AWA-FSF: (a) DS_Cs without considering neighbor corner districts in corner districts, (b) DS_{CS}s in corner districts, (c) DS_Ss in side districts, (d) Window sizing directions and sizing factors based on combining DS_{CS} and DS_S as shown in (b) and (c), and (e) Fine tuning effect of the center point of window by the AWA-FSF.

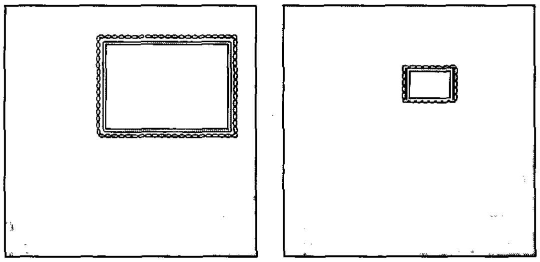
IV. Experimental Results

We have applied our proposed method based on the AWA-FSF presented in this paper to real image sequences. We compare the performance of our method having sizing factors described so far with that of adaptive window tracker whose window size is variable by just 1 pixel, $\triangle S_i \in \{-1, +1\}$ in (3), and that of another variant adaptive window tracker without sizing factors in which $\triangle S_t$ is determined as $\triangle S_i \in \{-S_{B_k}, +S_{B_k}\}$, i.e., $\triangle SF \in \{-1, +1\}$. When we actually performed tracking experiments based on fixed size window method for many cases in which the size of an object undergoes rapid change, the tracking failed so often that we did not include these experimental results. For objective comparison, the initial position and size of the adaptive windows were set to be the same. We present the results in Fig. 9 and Fig. 10. These experimental results show that the tracker with the AWA-FSF performs better than the others.

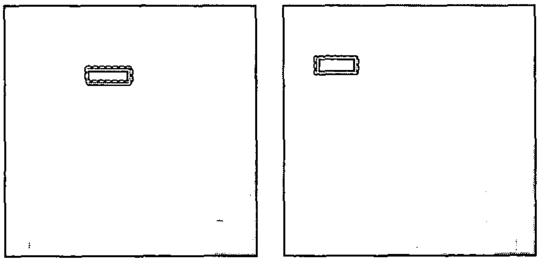
The dotted rectangle in these figures is the tracking result obtained from the Centroid-Compensated Correlator before window adjusting by the AWA-FSF. Meanwhile, the solid rectangle is the reference image region adjusted by the AWA-FSF that will be passed to the next frame. We can present some advantages: capability to correct walk-off error by suitably shifting the center point, enhancing stability and reliability by rapidly and tightly adjusting the reference window and so forth. In all test sequences, we found that our method needs fewer required frames than the others for enclosing a target tightly.

1. Object in the Air

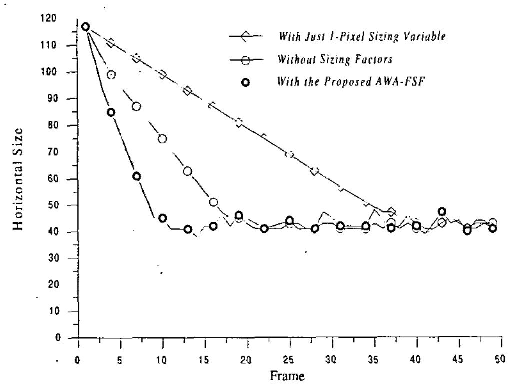
Fig. 9 shows a tracking result that our method is applied to an aircraft flying above a skyline of city in dark sky with low contrast. The test image sequence is consisted of 48 real images and an initial window size is intentionally chosen to be much larger than the object size to verify an initial window sizing speed. All methods succeeded in tracking a target over whole frames but their windowing performances were different. As shown in Fig. 9(e) and Fig. 9(f), the AWA-FSF could tightly adjust a vertical window size in 13 frames, an adaptive window without sizing factors in 27 frames, and 1-pixel variable window in 39 frames.



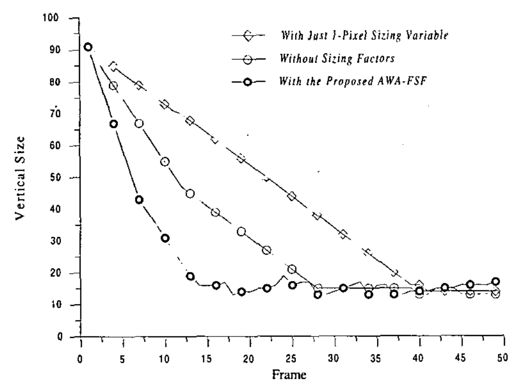
(a) Frame 1 (b) Frame 10



(c) Frame 17 (d) Frame 48



(e) Size variation in horizontal direction

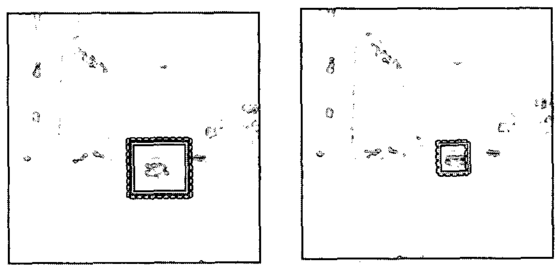


(f) Size variation in vertical direction

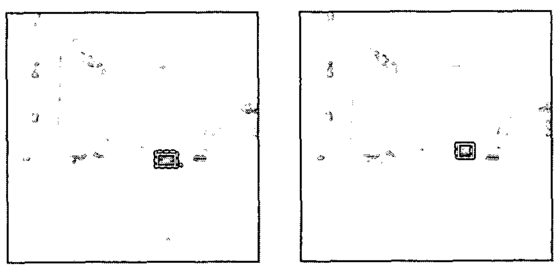
Fig. 9. Test image sequence that an aircraft flies in dark sky with low contrast.

2. Object on the Ground

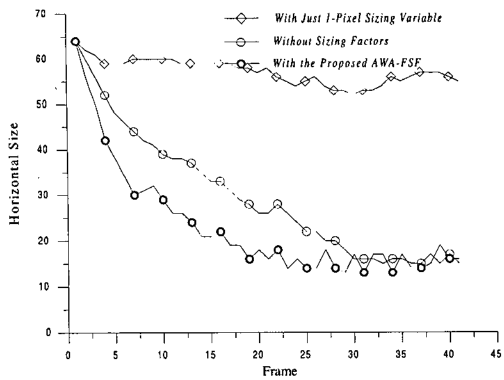
Fig. 10 shows the experimental results for an image sequence of the car went away from an observer to a lot of clutters. There are many clutter components far from an observer, i.e., the parked car and buildings in the background. So, the tracking window is needed to adjust compactly its size to object size as soon as possible. Referring to Fig. 10(e) and Fig. 10(f), the adaptive window without sizing factors could not properly adapt the window vertical size to the target vertical size and the adaptive window with just 1-pixel sizing variable could not adjust its size to target size in the both directions, i.e., vertical and horizontal direction. Their tracking window centers did not match up to the approximate target centers owing to failing in adjusting a proper window size. The AWA-FSF, however, accomplished tightly adjusting the window size to the target size. We found that at the beginning of image tracking, a low occupancy rate of an object in the tracking window mainly causes a basic correlator to accumulate walk-off error. Also, we have similar results from experiments applied our method to partially occluded object.



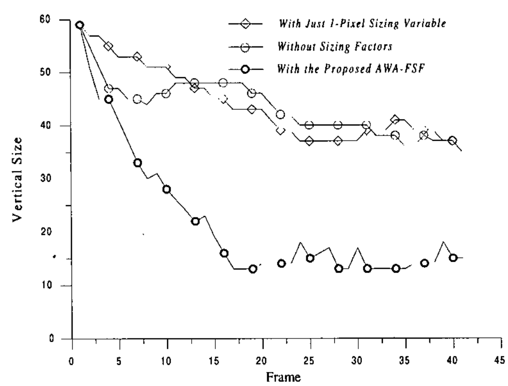
(a) Frame 1 (b) Frame 7



(c) Frame 16 (d) Frame 40



(e) Size variation in horizontal direction



(f) Size variation in vertical direction

Fig. 10. Test image sequence that a car fast goes away from an observer to clutters.

V. Conclusion

This paper presented an image tracking architecture employing the centroid compensation and the AWA-FSF for enhancing stability of correlation-based tracking in the cluttered surroundings. The centroid compensation provides our tracker with the capability of modifying the result of the correlation-based tracking to the real center of an object when weak clutter components exist within a window. When the initial window size happens to be much larger than the object size at the beginning of tracking, the AWA-FSF could generate enough sizing magnitudes to remove background components and to increase the occupancy rate of the object components in the reference image. Furthermore, a window with larger area exponentially spends larger computation time of correlation task. Generally, the window sizing speed of the AWA-FSF is found to be over three times as fast as that of the just 1-pixel sizable adaptive window.

By virtue of the proposed AWA-FSF capable of adjusting window size more rapidly and properly by generating sizing factors in four directions, we could achieve benefit of minimizing influence of complex background and clutter in correlation-based tracking process. In addition, this method can more finely tune the size of the reference image and the reference position repeatedly after the main tracking routine has been terminated. Since the AWA-FSF can tightly adjust window size around object, computation time for correlation could be optimized. We can process about several frames a second due to a CPU load of a PC, and thus expect to enhance the processing speed enough to process with real-time if we use a special purpose hardware. Our experimental results have demonstrated a clear advantage of the AWA-FSF over the adaptive window without sizing factors on 45 sequences composed of more than 3400 real images.

References

[1] D. A. Montera, S. K. Rogers, D. W. Ruck, and M. E. Oxley, "Object tracking through adaptive correlation," *Optical Engineering*, Vol. 33, No. 1, pp. 294-302, Jan. 1994.

[2] R. L. Brunson, D. L. Boesen, G. A. Crockett, and J. F. Riker, "Precision trackpoint control via correlation track referenced to simulated imagery," *Proc. SPIE Acquisition, Tracking, and Pointing VI*, Vol. 1697, pp. 325-336, Apr. 1992.

[3] A. D. Hughes and A. J. E. Moy, "Advances in automatic electro-optical tracking systems," *Proc. SPIE Acquisition, Tracking, and Pointing IV*, Vol. 1697, pp. 353-366, Apr. 1992.

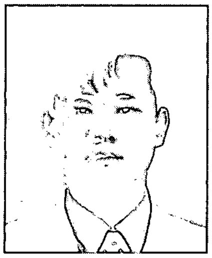
[4] G. J. Vanderbrug and A. Rosenfeld, "Two-stage template matching," *IEEE Trans. on Computer*, C-26, No. 4, pp. 384-393, Apr. 1977.

[5] J. L. Lotti and G. Giraudon, "Adaptive window algorithm for aerial image stereo," *12th International Conference on Pattern Recognition*, Vol. 1, pp. 701-703, Oct. 1994.

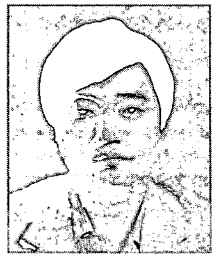
[6] T. Kanade and M. Okutomi, "A Stereo Matching Algorithm with an Adaptive Window: Theory and Experiment," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, Vol. 16, No. 9, pp. 920-932, Sep. 1994.

[7] S. H. Sung, S. I. Chien, M. G. Kim, and J. N. Kim, "Image Tracking of Moving Object Based on Multi-Step Sizable Adaptive Window and Sequential Joint Method," *Proc. the 9th KSPC*, Vol. 9, Part 1, pp. 517-520, Oct. 1996.

[8] E. R. Davies, *Machine Vision: Theory, Algorithms, Practicalities*, 2nd Ed., Academic Press, pp. 103-130, 1997.



Si-Hun Sung received the B.S. and M.S. degrees in electronic engineering from the Kyungpook National University, Taegu, Korea, in 1995 and 1997, respectively. He is currently working toward the Ph.D. degree in electronic engineering at the Kyungpook National University as a research assistant. His research interests include the areas of the field application of computer and machine vision, image processing, pattern recognition, and networks. He is a member of the Institute of Electronics Engineering of Korea.



Sung-Il Chien received the B.S. degree from Seoul National University, Seoul, Korea, in 1977, and the M.S. degree from the Korea Advanced Insitute of Science and Technology,. Seoul, Korea, in 1981, and Ph.D. degree in electrical and computer engineering from Carnegie Mellon University in 1988. Since 1981, he has been with the department of electronic engineering, Kyungpook National University, Taegu, Korea, where he is an associate professor. His research interests include pattern recognition, computer vision, and neural networks.