Journal of Imaging Science and Technology® 54(2): 020505–020505-9, 2010.

Robustly Adaptive Moving Thermal Object Segmentation Using Background Modeling Based on Runtime-Weighted Features

Changhan Park
SIAT, Samsung Thales Co., Ltd., Chang-Li 304, Namsa-myun, Cheoin-gu, Yongin City, 449-885 Gyeonggi-do, South Korea

Jik-Han Jung
School of EECS, KAIST, 373-1, Guseong-Dong, Yuseong-Gu, 305–701 Daejeon, South Korea

Kyung-Hoon Bae▲
SIAT, Samsung Thales Co., Ltd., Chang-Li 304, Namsa-myun, Cheoin-gu, Yongin City, 449-885 Gyeonggi-do, South Korea
E-mail: khbae.bae@samsung.com

Abstract. *Moving object segmentation plays an important role in a complex object tracking system. This system decides whether the current block belongs to the object region or not. In this article, a scheme using background modeling based on runtime-weighted features for robustly adaptive moving object segmentation in infrared (IR) image sequence is proposed. Proposed background modeling for an open hardware (H/W) architecture design decreases the size of the search area to construct a sparse block template of search area in infrared images. The authors also compensate for motion compensation when the image moves in previous and current frames of IR imaging system. The method of separation of background and objects applies to adaptive values through time analysis of pixel intensity. The proposed method uses more feature information such as intensity, deviation, block matching error, and velocity. The weighting values give a higher weight to feature information which has a large difference between the object and the background region. Based on experimental results, the proposed method showed real-time moving object segmentation through background modeling in the proposed embedded system. © 2010 Society for Imaging Science and Technology.*
[DOI: 10.2352/J.ImagingSci.Technol.2010.54.2.020505]

INTRODUCTION

In the field of computer vision, moving object segmentation plays a crucial role as a preliminary step for high-level image processing. To understand an image, one needs to isolate the objects in it and find relationships among them.[1] For typical real-time applications oriented to the analysis of visual scenes in order to identify events and actions, simplifications are needed. For these, motion is a key factor aiding the segmentation process.[2] As an active research topic in computer vision, visual surveillance in dynamic scenes attempts to detect, recognize, and track certain moving objects from image sequences, and more generally to understand and describe object behaviors.[3] The problem of nonrigid object tracking (OT) and recognition in video sequences of the objects' actions is of increasing importance to many applications.[4–6] Intelligence surveillance systems using infrared (IR) and charge coupled device (CCD) images have been utilized in object detection, tracking, and recognition.[7]

Several powerful algorithms for OT have been developed in the last two decades. For a stationary object tracking system, frame differencing analysis was preferred; it can be generalized to situations where the video data can be easily stabilized.[8,9] Modern appearance-based tracking schemes such as the mean-shift algorithm use histogram-based object appearance models, so they are robust to nonrigid pose changes.[10] Kalman filtering and particle filtering also contribute to enhance the tracking performance.[11] Collins reviewed these tracking schemes and emphasized that tracking success or failure depends primarily on how distinguishable an object is from its surroundings.[12] In addition, he also noticed that tracking features need to be used adaptively since both foreground and background appearance can be changed as the object moves from place to place.

In this article we propose an open hardware (H/W) architecture that a hardware structure designed using the graphic processing board with an embedded processor and host personal computer to perform the algorithms required for a large computational load, such as background modeling and block matching-based OT algorithms. Moreover, we verify our proposed structure by applying moving object segmentation algorithm to IR images. We also present a tracking scheme based on block matching in the proposed system and employ several features such as intensity, deviation over time duration, and matching error to classify each pixel into the object region or the background region as

▲IS&T Member.

Received Feb. 25, 2009; accepted for publication Dec. 3, 2009; published online Feb. 26, 2010.

1062-3701/2010/54(2)/020505/9/$20.00.

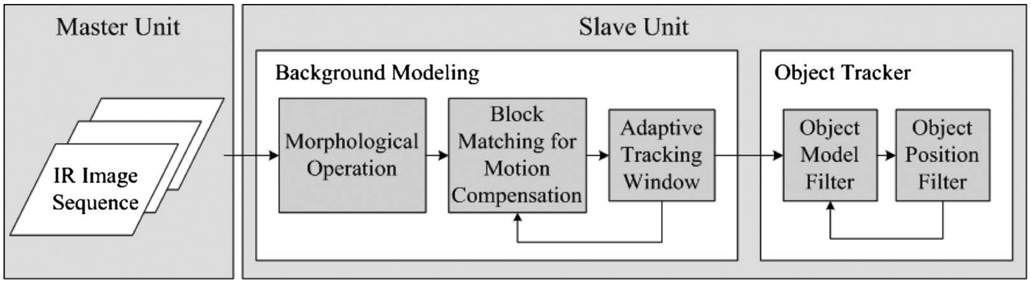

Figure 1. Block diagram of the proposed scheme for detection and tracking based on our system.

developed in previous research.[13] Each feature is weighted individually according to separability.

ROBUSTLY ADAPTIVE MOVING OBJECT SEGMENTATION

The block diagram of the proposed adaptive moving object segmentation (AMOS) scheme for background modeling and tracking based on an open H/W architecture is shown in Figure 1. First, we use morphological operation to extract roughly for edges and an object. Second, for each frame, motion compensation for background modeling calculates IR imaging system motion in order to stabilize an image. After the region of interest in the adaptive tracking window step is set, feedback to the block matching step yields the object model filter (OMF) and the object position filter (OPF). The OMF predicts the next object model at same-positional pixel and gives feedback to the background modeling step. Simultaneously, a predicted object position is provided for next position by the OPF.

Object-Based Motion Estimation Using Morphological Operation

Detecting motion in an IR image wastes much H/W resources in the course of background modeling for object detection. So, we define a prior meaning region to use minimum resources. The proposed scheme estimates motion roughly at thermal edges and the object based on morphological operations. The morphological operations are dilation, erosion, opening operation, closing operation, and top-hat operation, which are performed by convolving the structuring element with the image.[14] We use the morphological TopHat operation for thermal edge and object detection. The morphological TopHat operation for grayscale images is part of the basic toolbox of morphological operations. Its function is to detect contrasting objects on nonuniform backgrounds. Depending upon whether we are dealing with a thermal edge or objects in IR images, the operation is defined as

$$\mathrm{TopHat}(A,B) = A - (A \circ B) = A - \max_B(\min_B(A)), \quad (1)$$

where A and B represent a thermal IR image and a structuring element, respectively. $\circ$ represents the gray-scale opening of A by a structuring element B.

Block Matching for Motion Compensation

The goal of the block matching algorithm for motion compensation is to find the most similar block to the reference block in a tracking region, and it has been widely used for visual tracking and antishake. Most block matching techniques minimize the cost function such as sum of absolute differences (SAD), mean absolute error, and mean squared error. In this article we used SAD as the cost function. To find the best matching position, the conventional SAD criterion is evaluated as

$$(\hat{u},\hat{v}) = \arg\min_{(u,v)\in R^t} \sum_{i=0}^{S_x-1} \sum_{j=0}^{S_y-1} \left| I^t(u+i,v+j) - I^M(i,j) \right|, \quad (2)$$

where R^t is the size of sampling template in the restricted sparse region, S_x and S_y represent the width and the height of the restricted sparse region, respectively. I^t and I^M represent the current frame and the thermal object model, respectively.

In real applications, since there is jitter in an IR imaging system caused by observer or platform, image stabilization is required. Image stabilization can reduce the block matching error induced by IR imaging system jitter. In this article, we use global motion compensation as an image stabilization algorithm. Let I^{t-1} and I^t be the previous frame and the current frame, respectively. Global motion (G_x, G_y) can be estimated according to

$$(G_x,G_y) = \arg\min_{(dx,dy)} \sum_{i=G_W}^{W-1-G_W} \sum_{j=G_H}^{W-1-G_H} \left| I^{t-1}(i+dx,j+dy) - I^t(i,j) \right|, \quad (3)$$

where W and H are the width and the height of the current frame and dx and dy are restricted to $-G_W \le dx \le G_W$ and $-G_H \le dy \le G_H$. Here, G_W and G_H should be larger than maximum IR imaging system movement. Root mean squared error (RMSE) for detection of global motion is defined as

$$\mathrm{RMSE} = \sqrt{\frac{1}{N}\sum \mathrm{error}^2}, \quad (4)$$

where N and error represent frame number of image and accumulated error values between previous and current frames, respectively.

Adaptive Tracking Window for Region Restriction

We presented the region restriction scheme of moving object segmentation in previous research.[13] We propose an adaptive

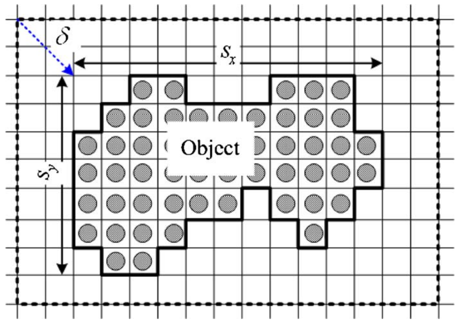

Figure 2. Scheme of adaptive tracking window for restricted region in IR image sequences.

tracking window for region restriction of a thermal object to implement background modeling. A scheme which restricts the signal processing region to the area surrounding the object is required in order to prevent clutter injection and heavy computational burden.

Let $(\hat{x},\hat{y})$ be the left-top position of the predicted object region in the current frame. Then the processing region P_r^t is restricted as

$$P_r^t = \{I^t(x,y) | \hat{x} - \delta \leq x - GM < \hat{x} + s_x + \delta, \hat{y} - \delta \leq y - GM < \hat{y} + s_y + \delta\}, \tag{5}$$

where (s_x, s_y) is the size of the thermal object and $\pm\delta$ are the maximum and minimum velocities of the object in the current frame. Global motion (G_x, G_y), estimated in Eq. (3), is considered in Eq. (5). Figure 2 shows the adaptive tracking window scheme for a restricted region in the current frame.

OMF for Background Modeling

For every pixel in the restricted region for the adaptive tracking window, the OMF determines whether each pixel is object or not by using the proposed process is described as follows:

(1) Check global motion values in block matching to compensate for motion.
(2) Divide the current frame I^t into two regions. $A(t)$ is a region overlapping both object and background regions, and $B(t)$ is a region not overlapping object and background regions in I^t.
(3) If region of $B(t)$ is a new background in the next frame, the background model is used; and if x is coordinate of an individual pixel, the average of intensity is $B_\mu^t(x) = I^t(x)$.
(4) If Eq. (6) is satisfied, each block is a candidate for the object model through background modeling.

$$|B_\mu^{t-1}(x) - I^t(x)| \leq 2.5 B_\sigma^{t-1}(x). \tag{6}$$

OPF for Object Tracking

The conventional method of object detection is to filter a single frame image using a high pass filter to gain a set of some number of candidate objects, then filter and track the true objects using a Kalman filter. This method is not very effective for a real-time based IR imaging system, because it is difficult to find a kind of algorithm that can adaptively detect and track both large moving objects and small objects. The object tracker step in proposed OPF provides two kinds of measurement. One is the new position of the object and the other is mean absolute difference (MAD). If MAD is relatively small, the object position moves to the current matching position in the current frame. We also propose a scheme in which a coast tracking algorithm keeps tracking the object while the object hides behind other objects or merges into background as shown in Figure 3. In the coast tracking technique, a criterion for determining whether the coast tracking is performed or not is needed. We consider motion estimation, which extracts average velocity of three frames when we assume a uniform velocity of an object in IR images.

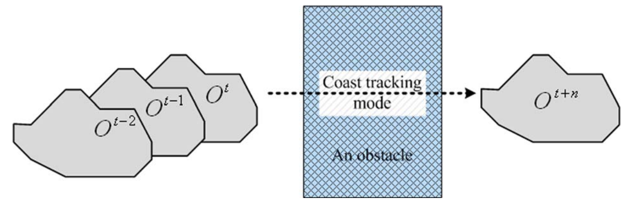

Figure 3. Detection and tracking algorithm keeps tracking the object while the object is hidden behind other objects or into background.

PROPOSED THERMAL OBJECT TRACKER

We implemented an open H/W architecture to verify the performance in an IR imaging system of the scheme proposed in previous research.[15] The proposed hardware architecture consists of the master unit and the slave unit for AMOS. The master unit consists of data storage, host processor, and graphic user interface (GUI) displayer. The slave unit consists of the input module of the IR or CCD camera, frame grabber, tracking part, and image output part. In this architecture, first the IR or CCD image sequences and the parameters involved with segmentation and tracking algorithms are transmitted to the tracking part. The tracking part can process image sequences captured by the frame grabber in the slave unit or stored in the master unit. The parameters are selected by the user and transmitted to the tracking part form the master unit.

Next, the tracking part in the slave unit performs segmentation, and tracking algorithms operate on the transmitted image sequences according to the user-selected parameters. After the segmentation and tracking algorithms are terminated, results such as runtime-weighted feature error, block matching error, object position, calculation time, object velocity, and direction are returned to the master unit. Finally, the master unit displays image sequences and the results. The block diagram of the proposed architecture is shown in Figure 4. Line A shows the processing flow, when using the image sequence captured by frame grabber. Line B shows the processing flow when using the image sequence stored in the master unit.

The software of the master unit consists of GUI and image sequence display software. The hardware of the slave

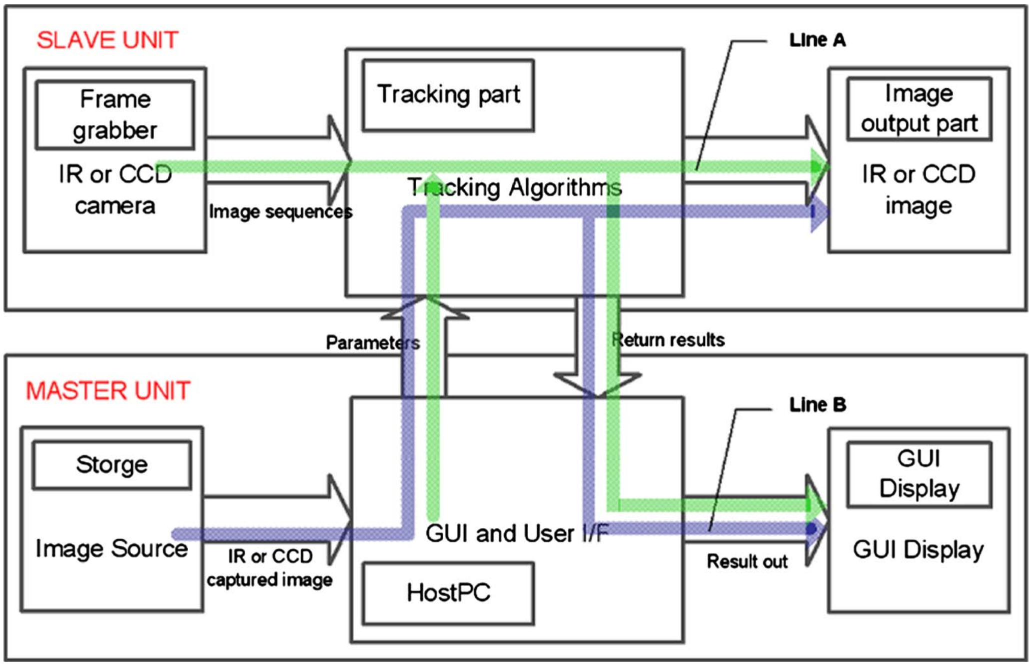

Figure 4. Block diagram of the master-slave H/W architecture.

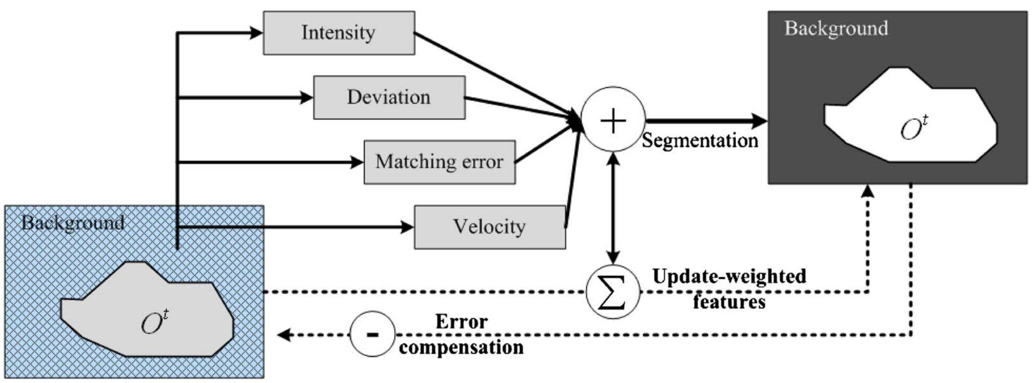

Figure 5. Features: intensity, deviation, matching error, and velocity.

unit is a graphic processing board. In the slave unit, the AMOS and tracking algorithms such as morphological top-hat operation, adaptive tracking window for background modeling, block matching, OMF, and OPF are implemented. The master unit and the slave unit communicate through a 100 MHz clock PCI-X bus. The master unit transmits parameters selected by a user and IR image sequences to the embedded processor in the tracking part through PCI-X bus. The embedded processor performs segmentation and tracking algorithms for the IR image sequences in accordance with parameters received by the master unit.

We presented the OMF scheme of moving object segmentation in previous research.[13] The proposed OMF updates shape, size, velocity, and each pixel's intensity of the object model in the current frame. At first, it determines the priorities of each feature. Then, with the priority information, the weighted-sum of every feature is evaluated pixel by pixel. Figure 5 shows this segmentation scheme. The proposed scheme consists of three stages. First, separability evaluation effects separation of object and background regions in the current frame. Second, we apply binarization to segment an object region in the same-positioned pixels. The

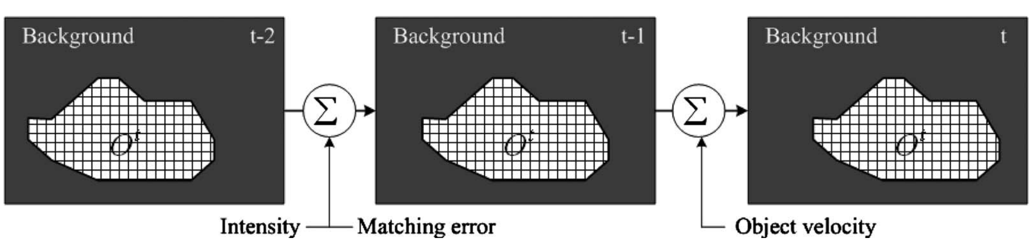

Figure 6. Illustration of separability: (a) intensity and (b) deviation of object.

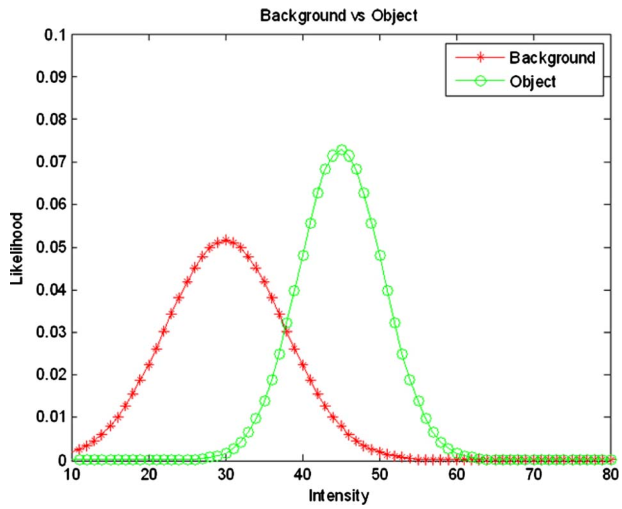

Figure 7. Binarization to separate for object and background regions.

$$NF^i(x,y) = \frac{F^i(x,y) - \left(\dfrac{\mu_T^i + \mu_B^i}{2}\right)}{|F_{MAX}^i - F_{MIN}^i|},$$

$$O(x,y) = \sum_{i=1}^{N} W^i \times NF^i(x,y), \qquad (8)$$

where $NF^i(x,y)$ is the normalized ith feature information at pixel (x,y) and $O(x,y)$, the objective function, is the weighted sum of each normalized feature information such as intensity, deviation, matching error, and velocity. The conventional Otsu method[16] is used to find an optimal threshold, and every pixel can thereby be classified into object or background. Figure 7 shows a histogram of object and background regions using the Otsu method.

Figure 8. Cars image sequence: (a) 50th frame, (b) 60th frame, (c) 65th frame, and (d) 75th frame.

last step we apply is to update the object model. Figure 6 shows features of intensity, matching error, and object velocity from the object model.

Separability Evaluation for an Object and Background Region

We presented the separability evaluation scheme for moving object segmentation in previous research.[13] Here we propose a scheme which classifies object and background regions through background modeling. Let the ith feature information at pixel (x,y) be $F^i(x,y)$. Then we can find two means, μ_T^i and μ_B^i, for each feature information, where ith is the mean of the ith feature information over the object region, T^t, and μ_B^i is the mean of the ith feature information over the background modeling, B^t. For each feature, the separability is defined by

$$W^i = \frac{\mu_T^i - \mu_B^i}{|F_{MAX}^i - F_{MIN}^i|}, \qquad (7)$$

where F_{MIN}^i and F_{MAX}^i represent the minimum and maximum value of the ith feature information space, respectively.

Binarization to Segment for an Object Region

We also presented a binarization scheme for moving object segmentation in previous research.[13] Here we propose a scheme to segment for object region through background modeling. We also define a new objective function to determine whether each pixel is an object pixel or not,

Update for Object Model Filter

The proposed scheme needs to be updated periodically with respect to the object and background regions because changes occur in the object's position, velocity, and acceleration. In the binarization stage after the separation stage for object and background regions in the current frame, each pixel is classified into the object or background region. After removing some noise pixels, we can enclose object pixels with a rectangle using position, velocity, and acceleration of the previous object. The new object size is determined by the

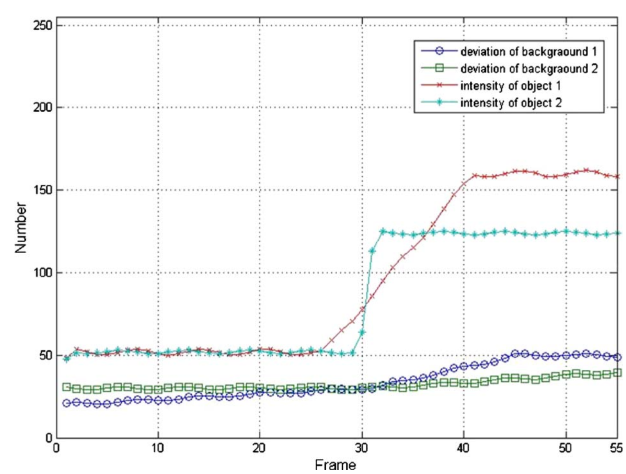

Figure 9. Statistic of a car: (a) intensity and (b) deviation of object and background.

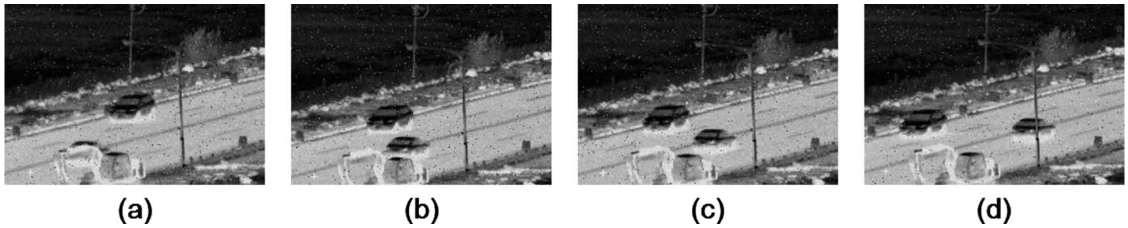

Figure 10. Cars image sequence with image shake, Gaussian and salt/pepper noises: (a) 50th frame, (b) 60th frame, (c) 65th frame, and (d) 75th frame.

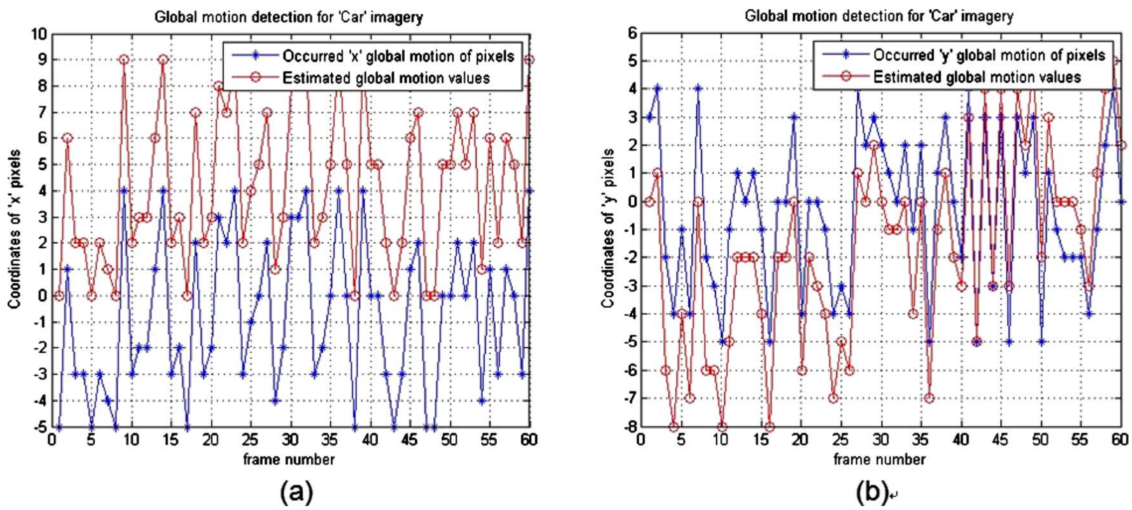

Figure 11. Global motion detection for Car imagery.

Figure 12. Segmentation image sequence of "Cars:" (a) 50th frame, (b) 60th frame, (c) 65th frame, and (d) 75th frame.

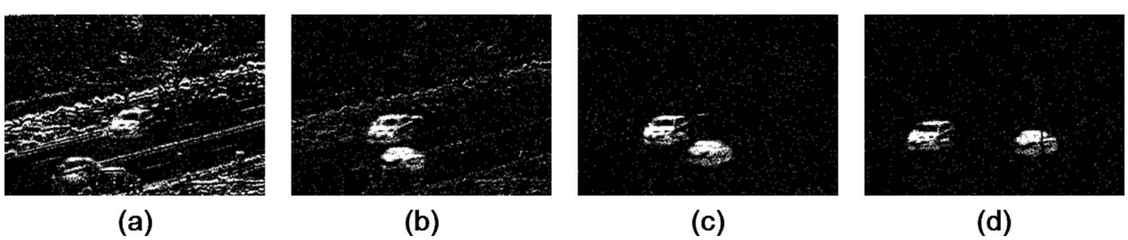

Figure 13. Median filter based segmentation image sequence of Cars with image shake and noise: (a) 50th frame, (b) 60th frame, (c) 65th frame, and (d) 75th frame.

Figure 14. Median and mean filter based segmentation image sequence of Cars with image shake and noise: (a) 50th frame, (b) 60th frame, (c) 65th frame, and (d) 75th frame.

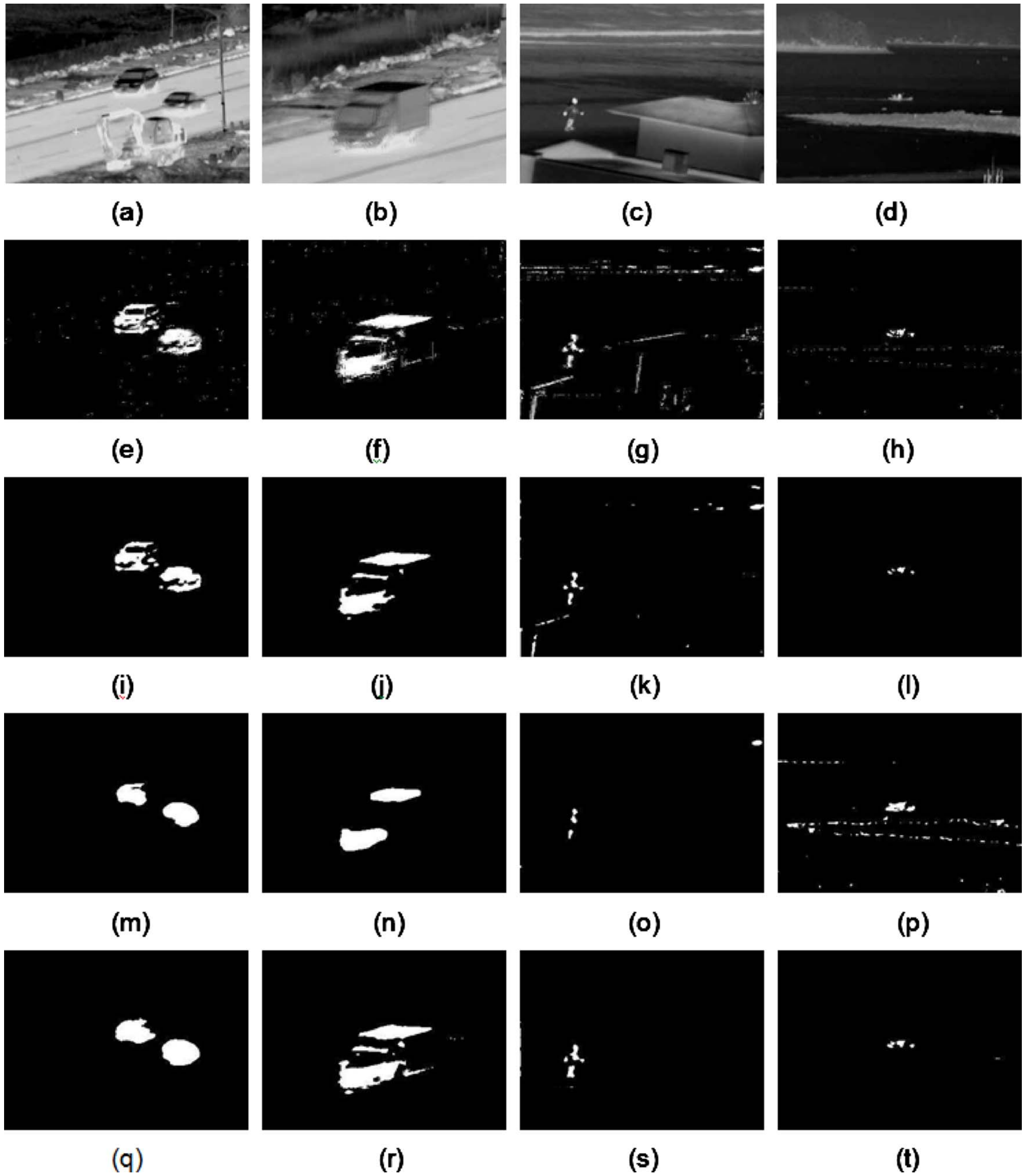

Figure 15. Segmentation results using 65th frame of Car, 110th frame of Truck, 41st frame of Person, and 50th frame of Ship: (a)–(d) original frame; (e)–(h) pixel average implementation; (i)–(p) two and four coefficients of the median filter for noise elimination respectively; (q)–(t) proposed method.

width and height of the rectangle in the current frame. The update of OMF is defined by

$$U_{\mathrm{OM}} = w_1 \times I_c + w_2 \times R_d + w_3 \times P_m + w_4 \times P_d + w_5 \times D_c, \tag{9}$$

where w_1–w_5, I_c, R_d, P_m, P_d, and D_c represent weighted constant values, intensity of current frame, difference between current frame and reference frame, mean of previous frame, variance of previous frame, and center distance of current rectangle, respectively.

EXPERIMENTAL RESULTS

The performance of moving object segmentation using background modeling based on runtime-weighted features is affected by noise, especially zero-mean Gaussian noise. To mitigate the noise effect, various details can be used. Under the assumption that an object is brighter than the background, the size of an object approaching the IR imaging system is larger than the previous object. Figures 8(a) and 8(d) show “Cars” original sequence from 50th to 75th frame as detected by the IR imaging system. The statistics of the Cars is shown in Figure 9. In this case, the intensity differ-

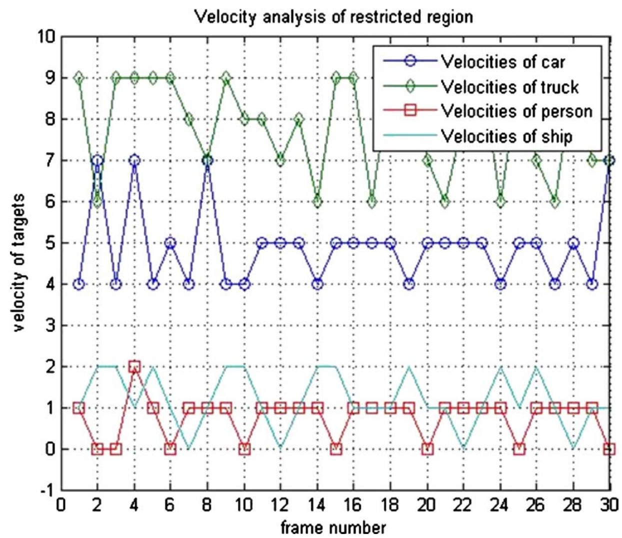

Figure 16. Velocity analysis of restricted region from adaptive tracking window.

ence between the object and background regions is much larger than deviation difference. Therefore, the intensity is the dominant discriminative feature. We did not update the object model until the frame at which the intensity change exceeded the uncertainty characteristic of the IR imaging system. So an abrupt discontinuity is shown.

Figures 10(a)–10(d) show Cars sequence from 50th to 75th frame captured by the IR imaging system with image shake; movement is ± 10 pixels, and zero-mean Gaussian noise (standard deviation: 10), also salt/pepper noise (probability: 0.01). Figure 11 shows estimated global motion values when applied to the ± 10 pixels of global motion from Fig. 10. We get x and y axis RMS pixel errors of 3.02 from Figs. 11(a) and 11(b), respectively.

Figures 12(a)–12(d) illustrate the segmented result using runtime-weighted features from Fig. 8 when the IR imaging system does not incur motion. Figures 13(a)–13(d) illustrate the segmented result using runtime-weighted features from Fig. 10 when IR imaging incurs motion; the results used not only median filter for removing noise but also global motion compensation for stabilization of image shake. Object extraction results for background modeling using the proposed scheme from the 60th frame are as shown in Fig. 13(b). Figures 14(a)–14(d) illustrate the segmented result using runtime-weighted features from Fig. 10 to heighten robustness against noise; the results is used not only median and mean filter but also global motion compensation for stabilization of image shake. In the simulation result, it is found that the 75th frame's segmented image results are better than for the 50th, 60th, and 65th frames. Segmentation image sequences from the simulation are shown in Fig. 14.

Figure 15 shows, from left to right, the segmentation result for 65th frame of Car sequence, 110th frame of "Truck" sequence, 41st frame of "Person" sequence, and 50th frame of "Ship" sequence. Figures 15(a)–15(d) show the original frame; Figs. 15(e)–15(h) show the results from the pixels average implementation; and Figs. 15(i)–15(p) show results obtained with two and four coefficients of the median filter for noise elimination, respectively. Figs. 15(q)–15(t) show results by the proposed method.

Figure 16 illustrates the results for object velocities in a restricted region using the adaptive tracking window from Fig. 15. Estimate of motion of an object is acquired from the object's position and velocity between the previous frame and the current frame. For motion estimation results we got an average of 4.9 pixels for Car, an average of 7.8 pixels for Truck, an average of 1.2 pixels for Person, and an average of 1.0 pixel for Ship. Figures 17 compares the tracking results for the proposed method with ground truth and Fig. 9; we estimate average RMS errors along x and y axes as 1.7 and 0.1 pixels, respectively. Figure 18 illustrates the object detection and tracking results using the proposed scheme with image-shake; its moving deviation is ± 10 pixels. Figs. 18(b) and 18(c) show results for coast tracking mode.

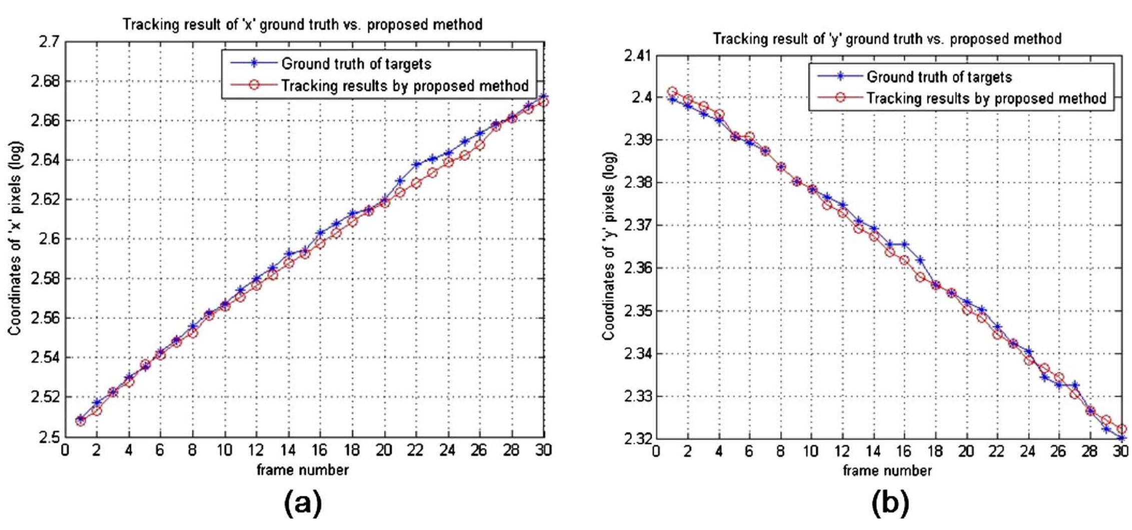

Figure 17. Tracking results of ground truth vs. proposed method: (a) x coordinate and (b) y coordinate.

Figure 18. Detection and tracking results in our system: (a) 35th frame, (b) 41st frame, (c) 51st frame, and (d) 70th frame.

CONCLUSIONS

In this article, a scheme using runtime-weighted, features-based, robustly adaptive moving object segmentation for an infrared image sequence is proposed. Proposed background modeling for an open H/W architecture design decreases the size of the search area in order to construct a sparse block template of the search area in the IR images. We also compensate for motion when the object moves between previous and current frames captured by the IR imaging system. The method of separation between background and objects leads to adaptive values through time analysis of pixel intensity. The proposed method uses more feature information such as intensity, deviation over time duration, block matching error, and velocity. The weighting values give a higher weight to the feature information which exhibits a large difference between object and background regions. Based on experimental results, the proposed method showed successful real-time moving object segmentation through background modeling using the proposed embedded system.

REFERENCES

[1] K. Deshmukh and G. Shinde, "An adaptive color image segmentation", Electronic Letters on Computer Vision and Image Analysis **5**, 12–23 (2005).

[2] L. Teixeira, J. Cardoso, and L. Real, "Object segmentation using background modeling and cascaded change detection", J. Multimedia **2**, 55–64 (2007).

[3] W. Hu, T. Tan, L. Wang, and S. Maybank, ""A survey on visual surveillance of object motion and behaviors", IEEE Trans. Syst. Man Cybern., Part C Appl. Rev. **34**, 334–352 (2004).

[4] I. Haritaoglu, D. Hartwood, and L. Davis, "Real-time surveillance of people and their activities", IEEE Trans. Pattern Anal. Mach. Intell. **22**, 809–830 (2000).

[5] S. McKenna, Y. Raja, and S. Gong, "Tracking colour objects using adaptive mixture models", Image Vis. Comput. **17**, 225–231 (1999).

[6] R. Plankers and P. Fua, "Tracking and modeling people in video sequences", Comput. Vis. Image Underst. **81**, 285–302 (2001).

[7] C. Park, K. Bae, and J. Jung, "Object recognition in infrared image sequences using scale invariant feature transform", Proc. SPIE **6968**, 1–9 (2008).

[8] C. Stauffer and W. Grimson, "Adaptive background mixture models for real-time tracking", *Proc. IEEE Computer Vision and Pattern Recognition II* (IEEE, Piscataway, NJ, 1999) pp. 246–252.

[9] M. Irani and P. Anandan, "A unified approach to moving object detection in 2D and 3D scenes", IEEE Trans. Pattern Anal. Mach. Intell. **20**, 577–589 (1998).

[10] D. Ramesh and P. Meer, "Real-time tracking of non-rigid objects using mean shift", *Proc. IEEE Computer Vision and Pattern Recognition II* (IEEE, Piscataway, NJ, 2000), pp. 142–149.

[11] M. Isard and A. Blake, "Condensation-conditional density propagation for visual tracking", Int. J. Comput. Vis. **29**, 5–28 (1998).

[12] R. Collins and Y. Liu, "Online selection of discriminative tracking features", *Proc. IEEE International Conference Computer Vision* (IEEE, Piscataway, NJ, 2003), Vol. **1**, pp. 346–352.

[13] J. Jung, H. Lee, D. Park, C. Park, and J. Lee, "Adaptive target segmentation using runtime-weighted features", Proc. SPIE **6567**, 1–7 (2008).

[14] C. Xiaochun, H. Yihua, H. Guilan, T. Xiaohong, and L. Wuhu, "Morphology based adaptive preprocessing method of infrared image sequence", Proc. SPIE **6279**, 1–6 (2007).

[15] J. Lee, J. Youn, and C. Park, "PowerPC-based system for tracking in infrared image sequences", Proc. SPIE **6737**, 1–9 (2008).

[16] N. Otsu, "A threshold selection method from gray level histograms", IEEE Trans. Syst. Man Cybern. **SMC-9**, 62–66 (1979).