AN OBJECT TRACKING METHOD USING ADAPTIVE TEMPLATE UPDATE IN IR IMAGE SEQUENCE

PyeongGang Heo, HyungTae Lee, JungYoup Suk, SangHun Jin, and HyunWook Park

Dept. of Electrical Engineering, KAIST,
373-1 Guseong-dong, Yuseong-gu, Daejeon 305-701, Korea
hwpark@athena.kaist.ac.kr

ABSTRACT

In object tracking, the template matching methods have been developed and frequently used. It is fast enough, but not robust to an object with the variation of size and shape. In order to overcome the limitation of the template matching method, this paper proposes a template update technique. After finding an object position using the correlation-based adaptive predictive search, the proposed method selects blocks which contain object's boundary. It estimates the motion of boundary using block matching, and then updates template. We applied it to IR image sequences including an approaching object. From the experimental results, the proposed method showed successful performance to track object.

Keywords: block matching, IR image, object tracking, template update

1. INTRODUCTION

Currently, object tracking is researched actively for many application areas including robot vision, defense, surveillance system, and navigation. Some applications require its tracking performance should be maintained even in a case that object's shape is varied largely and suddenly. Thus, object tracking method is developed toward estimating not only object's movement but also object's variation.

Feature-based object tracking method finds tracking point using features extracted from an object and a target image. Its performance may be different as characteristic of the feature. It usually needs huge computational costs to generate invariant features, which makes it hard to be applied in a real-time tracker. KLT (Kanade-Lucas-Tomasi) [1], [2], and SIFT (Scale-Invariant Feature Transform) [3] tracker are typical feature-based trackers. KLT is the tracking method that extracts feature windows expressing an object, and finds each position having smallest intensity difference between feature windows. SIFT tracking method is to extract scale- and rotate-invariant features and match those features between two adjacent frames.

Secondly, template matching method finds the place having highest similarity between the template representing an object and a target image. Measures of similarity are usually SAD (sum of absolute difference), SSD (sum of squared difference) [4], [5], or correlation [6], [7]. This method needs relatively a little computational costs, but has disadvantage that it is not robust to object's shape variation.

Among various similarity measures, the normalized correlation needs relatively large computational costs, but tracks a target with accuracy and stability. The CAPS (Correlation-based Adaptive Predictive Search) [8] algorithm reduces the computational costs while maintaining the accuracy and stability. It determines skip-width and range under the assumption that auto-correlation and cross-correlation have a similar trend around the matching position.

In this paper, FLIR (Forward Looking Infrared) image sequence is used. The FLIR image sensor makes an image visible at night and during light rain, fog, haze, or dusty. The proposed method is applied to an approaching object in FLIR images. Section 2 presents the proposed method of the adaptive template update. Section 3 shows experimental result. Conclusion is given in Section 4.

2. THE PROPOSED METHOD

In image sequence, the object's size and shape are usually changed because of its motion and rotation. If the template is not frequently updated when the object shape is changed, the tracking may have a problem in accuracy. At first, the template is decided to cover the entire object in manual. Being tracked, the object may change its size and shape and differ from initial appearance in the image. When an object grows bigger and bigger in the image, object tracker with a fixed-size template may track a local part of the object having maximum similarity coefficient. As shown in fig. 1, the final template may indicate different object.

For correct tracking, the tracker should consider the variation of object's size and shape since the distance between the camera and the object changes and the object moves. However, it is difficult for the template matching method to track a shape or size changed object. Thus, the proposed method finds the motion of object's boundary and estimates a change of an object's shape or size.

2.1 Block Selection

When an object size changes, the boundary of the object also changes. The proposed method can detect change of the object size by estimating movement of blocks in the object boundary. Firstly, the method checks blocks along 4 boundaries (top, bottom, left and right) to estimate changes of object's shape or size. Figure 2 shows the position of the checked blocks.

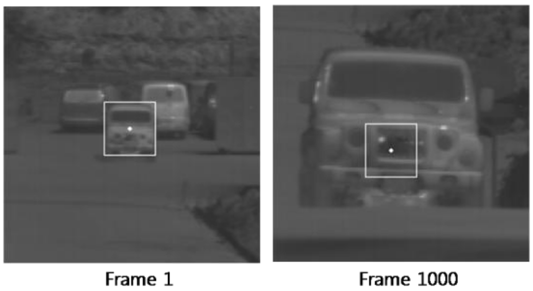

Figure 1. Tracking example with a fixed-size template

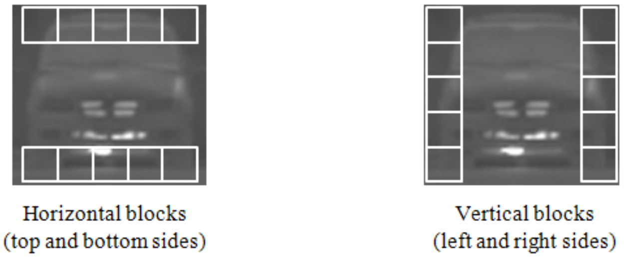

Figure 2. Position of checked blocks in horizontal and vertical sides.

The method selects blocks having edge information from Sobel gradient operation and thresholding. The checked blocks are divided into two types (horizontal type and vertical type) according to their position. Considering the type of blocks, then, every pixel of the blocks is convoluted with Sobel operator [9] as follows,

$$\begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} \text{ for vertical blocks and}$$

$$\begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix} \text{ for horizontal blocks.}$$

The block size is proportional to the template size. Figure 3 shows the template size of the experimental results on various ratios of the template to the block size when the object's size increases. It also shows the updating of the template size is unstable when the ratio is large. Thus the result of block matching is not reliable when the block size is small. Figure 4 shows the template size of the experimental results on various threshold values when the object's size increases. The threshold value influences the number of selected blocks. The information is not enough to detect variation of object's size when the number of selected blocks is small. Thus, the result of updating template size does not follow the trend of the variation of the object's size when the threshold value is too large. In this experiment, we selected the ratio of 4 and the threshold value of 30.

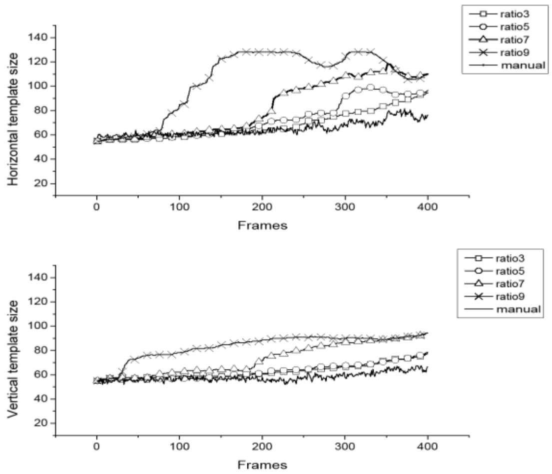

Figure 3. Results of updating template size on various ratios.

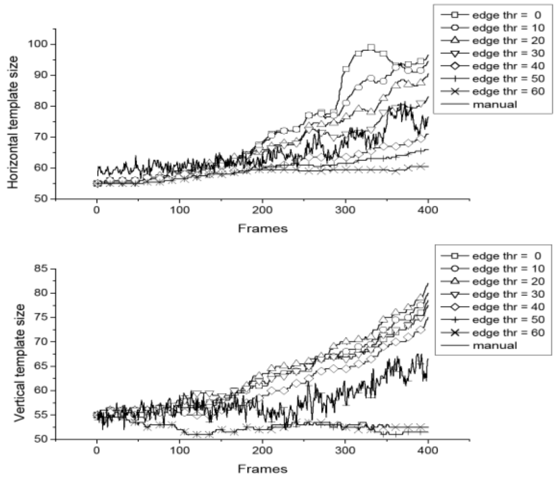

Figure 4. Results of updating template size on various threshold values.

2.2 Block Matching

The proposed method finds the best matched position of each block. Blocks of a small template have narrow search ranges because object size changes a little. On the contrary, those of big template have wide search ranges. In addition, the method considers that the block matching is sensitive to noise if blocks are relatively small. Thus, every block is smoothed, which results in reducing the speckle noise in IR image.

2.3 Template Size Update

Using motion vector, from the block matching, the proposed method updates the template's size. The method defines two modes: an "increasing" mode if a block motion vector directs to outer side of the template, and a "decreasing" mode if a block motion vector directs to inside of the template. For deciding a mode of boundary, we chooses the median of motion vector set which is ordered by the edge energy of the selected blocks in previous step. If the mode of boundary is decided as an 'increasing' mode, the motion vector directing the furthest position becomes the motion of boundary because the template of next frame should include the matched blocks. On the other side, the median of motion vector set becomes the motion of boundary when the mode of boundary is 'decreasing' mode. Finally, the method updates template after deciding every motion of four boundaries.

3. EXPERIMENTAL RESULTS

Image sequence used in the experiment is a FLIR image sequence provided from an image tracker in real application field. While a car comes closer, image tracker keeps to track it in a given image sequence. This paper compares CAPS tracker, SIFT tracker and the proposed method. The experiments are all done on an Intel core 2 2.13 GHz PC computer. Table 1 shows the processing time of three methods. As template size increases, the processing time also increase. It proves that the proposed method is faster than SIFT tracker.

The proposed method reduces computational costs while maintaining accuracy and reliability. Since it matches a template and a target image, and finds tracking position in just local search area, the proposed method reduces the computational costs. Fig. 5 compares three methods. First method is to expand all frames and process full-search method. Second method is to expand all frames and process 2-step CAPS algorithm. Third method is the proposed method. The proposed method is much faster than others.

Table 1. Comparison of processing time of three different methods

	Processing time (ms)		
Frame	CAPS tracker	SIFT tracker	Proposed method
1~120	101.51	2572.43	194.50
121~240	107.41	3430.41	210.87
241~360	111.46	4485.97	257.34
361~480	106.48	5018.16	345.69
481~600	114.35	4850.42	472.37

4. CONCLUSION

This paper presented an object tracker dealing with template size update in FLIR image sequence. The proposed method firstly found tracking point using the correlation-based adaptive predictive search, and selected blocks having high edge energy along the template boundary. Then, it searched the motion of an object's boundary through block matching. Finally, the proposed method updated the template size.

The experimental results showed high performance of the proposed method in terms of template size update and tracking accuracy. In addition, the proposed method is also faster than other tracking methods considering a size-changed object.

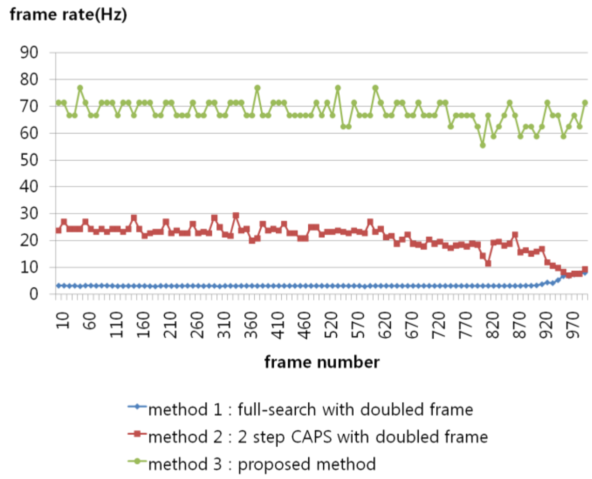

Figure 5. Comparison of frame rate of three different methods

ACKNOWLEDGEMENTS

We would like to thank LIG Nex1 Co. Ltd. for supports including FLIR image sequences.

REFERENCE

[1] C.Tomasi and T.Kanade, "Shape and Motion from Image Streams: a Factorization Method-Part 3 Detection and Tracking of Point Features," *Technical Report MU-CS-91-132*, Carnegie Mellon University, Pittsburgh, PA, Apr. 1991.

[2] J.Shi and C.Tomasi, "Detection and Tracking of Point Features," *IEEE Conference on Computer Vision and Pattern Recognition*, Seattle, Jun. 1994.

[3] D.Lowe, "Distinctive Image Features from Scale-Invariant Keypoints," *International Journal of Computer Vision*, vol.60, issue 2, pp.91-110, 2004.

[4] J.Jain and A.Jain, "Displacement Measurement and Its Application in Interframe Image Coding," *IEEE Transactions on Communications*, vol.29, pp.1799-1808, 1981.

[5] ISO/IEC MPEG-4 Video verification model version 18.0 ISO/IEC JTC1/SC29/WG11 N3908, 2001.

[6] A.Rosenfeld, "Picture Processing by Computer," New Yotk: Academic Press, 1969.

[7] W.Pratt, "Correlation Techniques of Image Registration," *IEEE Transactions on Aerospace and Electronic Systems*, vol.10, pp.353-358, 1974.

[8] S.Sun, H.W.Park, D.R.Haynor, and Y.Kim, "Fast template matching using correlation-based adaptive predictive search," *International Journal of Imaging Systems and Technology*, vol. 13. Issue 3, pp169-178, 2003.

[9] A. K. Jain, "Fundamentals of Digital Image Processing", Prentice Hall, Englewood Cliffs, 1989.

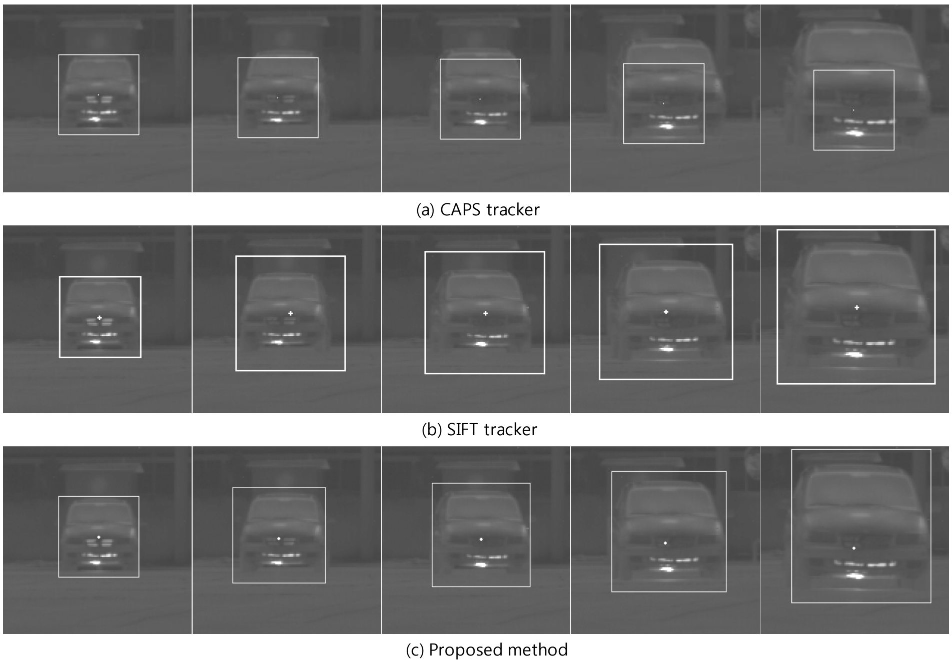

Figure 6. Tracking results of an image sequence 1 for three methods. (a) Tracking using fixed-size template on integer-pixel accuracy. (b) Tracking using SIFT. (c) The proposed method on half-pixel accuracy.

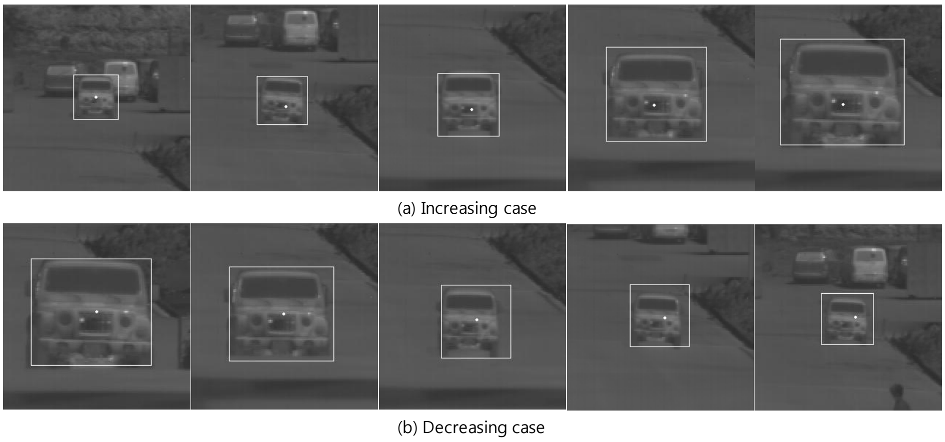

Figure 7. Tracking results of an image sequence 2 using the proposed method. (a) Case which the object increases. (b) Case which the object decreases.