

Optical Engineering

SPIDigitalLibrary.org/oe

Scale-adaptive object tracking using color centroids

Suk-Ho Lee
Euncheol Choi
Moon Gi Kang

Scale-adaptive object tracking using color centroids

Suk-Ho Lee,^a Euncheol Choi,^b and Moon Gi Kang^b

^aDongseo University San 69-1, Churye-2-dong, Sasang-Gu, Busan, 617-716, South Korea

^bYonsei University School of Electrical and Electronic Engineering, Institute of TMS Information Technology, 134 Shinchon-dong, Seodaemun-Gu, Seoul, Seoul 120-749 South Korea

E-mail: mkang@yonsei.ac.kr

Abstract. We propose a stable scale-adaptive tracking method that uses the centroids of the target colors in the target localization and scale adaptation. Because of the spatial information inherent in the centroids, a direct relationship can be established between the centroids and the scale of the target region. After the zooming factors are calculated, the unreliable zooming factors are filtered out to produce a reliable zooming factor that determines the new scale of the target. © 2011 Society of Photo-Optical Instrumentation Engineers (SPIE). [DOI: 10.1117/1.3633648]

Subject terms: object tracking; centroid shifts; scale adaptation; histograms.

Paper 110553LRR received May 22, 2011; revised manuscript received Aug. 16, 2011; accepted for publication Aug. 17, 2011; published online Sep. 15, 2011.

1 Introduction

Scale-adaptive tracking using only color information is a difficult and important problem. One of the most successful approaches using only color information is the kernel-based approach, which has become popular due to its tracking speed.^{1,2} However, due to the instability of color-based tracking, research to stabilize the tracking has been done by introducing the scale space,³ changing the kernel function,^{4,5} using multiple kernels or patches,⁶⁻⁹ incorporating temporal filtering,¹⁰ or discriminating local reliable regions.¹¹ Most color-based scale-adaptive tracking methods have utilized the histogram to determine the target size.^{1-4,12} However, in certain cases, the histogram fails to provide a good estimate of the target size (e.g., in cases of occlusion or the appearance of colors in the background similar to the target colors). This is due to the fact that the histogram is related to the numbers of pixels corresponding to the target colors. Therefore, sometimes colors in the background similar to the target colors cause the target region to spread out to the background region. In other cases, the target window sometimes shrinks too much, because small partial regions of the target show a similar distribution of colors as the entire target region.

In this letter, we propose a stable scale-adaptive tracking algorithm that combines the color centroids-based tracking algorithm in Ref. 13 with a new scale-adaptation algorithm also based on the use of centroids. The advantages of

using centroids in the computation of the scale are as follows: first, centroids have a direct relationship with the scale of the target, which makes the estimation of the scale simple and fast. Second, centroids can be filtered by filtering algorithms, leaving only the reliable centroids to be used in the rescaling. The proposed tracking algorithm produces stable scale-adaptive tracking results even in difficult cases when the background is cluttered or contains target colors, or when occlusion occurs.

2 Scale-Adaptive Centroid-Based Tracking Algorithm

In this section, we first introduce the color centroids-based target localization algorithm. Then, we propose the scale-adaptation algorithm, which is also based on the use of the color centroids. The combination of the two algorithms will be explained at the end of this section.

2.1 Target Window Shifting Using Centroids of the Target Colors

In Ref. 13, we proposed a stable target-shifting algorithm that shifts the current target location to the next target location based on the use of centroids of the target colors. The target position in the current frame (\hat{y}_0) is obtained by the area-weighted mean of the color centroids in the current frame, i.e.,

$$\hat{y}_0 = \frac{\sum_{u=1}^m \hat{q}_u \mathbf{C}_u^n}{\sum_{u=1}^m \hat{q}_u}, \quad (1)$$

where \mathbf{C}_u^n represents the centroid of the color bin u in the current frame, and $\{\hat{q}_u\}_{u=1, \dots, m}$ represents the m -bin histogram of the target model obtained in the initial frame. Here, the weights are $\{\hat{q}_u\}_{u=1, \dots, m}$, which correspond to the areas in the initial frame that the color bins $\{\hat{q}_u\}_{u=1, \dots, m}$ cover.

The location of the target in the next frame (\hat{y}_1) can be calculated in the same way as \hat{y}_0 using \mathbf{C}_u^{n+1} , the color centroids in the next frame, instead of \mathbf{C}_u^n in Eq. (1). Then, the shifting vector \hat{y}_{shift} which shifts the location of the target in the current frame to the location in the next frame, is computed as $\hat{y}_{\text{shift}} = \hat{y}_1 - \hat{y}_0$.

2.2 Scale Adaptation Based on Centroid Difference Vectors

Although most scale-adaptation algorithms utilize histograms of the target colors, we utilize the centroids of the colors for the rescaling of the new target region. The reason for this is based on the following facts:

1. The centroids contain the spatiality information and have a direct relationship to the scale of the target region.
2. Centroids that deviate much from their real positions due to the background colors can be easily detected and eliminated from the estimation of the scale.
3. The rescaling based on the centroids is not very sensitive to the appearance of background colors similar to the target colors as that based on histogram similarity.

For scale adaptation, we first divide the target region (which is centered at \hat{y}_1) into four equal subregions, as shown in Fig. 1. Figures 1(a) and 1(b) show the target in the current

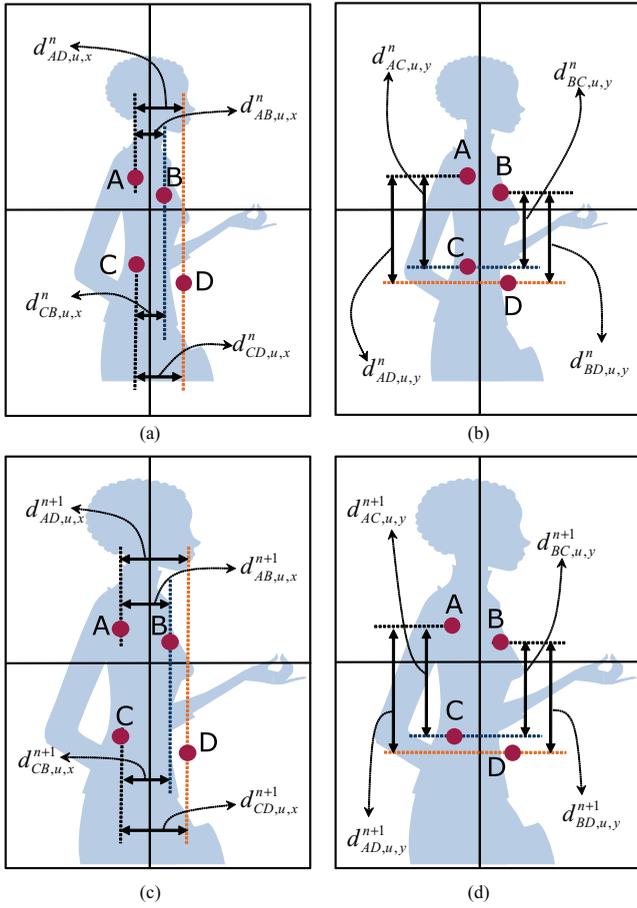


Fig. 1 Distances between the centroids in different subregions: (a,b) Corresponding to the current frame, and (c,d) corresponding to the next frame. (a,c) Distance in the x-coordinates and (b,d) distance in the y-coordinates.

frame and Figs. 1(c) and 1(d) show the target in the next frame. The scale of the target changes in the next frame. After dividing the target region into subregions, we calculate the centroids in each subregion, independently. Figures 1(a) and 1(b) show the calculated centroids in the current frame, and Figs. 1(c) and 1(d) show those in the next frame. As shown in Figs. 1(a) and 1(c) or in Figs. 1(b) and 1(d), the distances between the centroids in different subregions corresponding to the same colors have a direct relationship to the size of the target region. For example, in Figs. 1(a) and 1(c), the ratio between $d_{AD,u,x}^n$ and $d_{AD,u,x}^{n+1}$ can be used as an estimate of the ratio of the target's width in the next frame to that in the current frame.

We first make some notations. We denote by $d_{A,B,u,x}^{n+1}$ the absolute difference in the x-coordinates between the centroids in subregions A and B corresponding to the color u :

$$d_{A,B,u,x}^{n+1} = |C_{A,u,x}^{n+1} - C_{B,u,x}^{n+1}|, \quad (2)$$

where $C_{A,u,x}^{n+1}$ and $C_{B,u,x}^{n+1}$ represent the x-coordinates of the centroids corresponding to the color u in subregions A and B, respectively, and the superscript $n+1$ denotes the next frame. Likewise, we can define $d_{A,B,u,x}^{n+1}$, $d_{A,D,u,x}^{n+1}$, $d_{C,B,u,x}^{n+1}$, and $d_{C,D,u,x}^{n+1}$. Then, we compute the weighted average of the distances $d_{A,B,u,x}^{n+1}$, $d_{A,D,u,x}^{n+1}$, $d_{C,B,u,x}^{n+1}$, and $d_{C,D,u,x}^{n+1}$ to obtain $d_{u,x}^{n+1}$, where the weight is related to the numbers of pixels in

the color bin u in the following subregions:

$$d_{u,x}^{n+1} = W_{A,B,u} d_{A,B,u,x}^{n+1} + W_{A,D,u} d_{A,D,u,x}^{n+1} + W_{C,B,u} d_{C,B,u,x}^{n+1} + W_{C,D,u} d_{C,D,u,x}^{n+1} \quad (3)$$

where

$$W_{A,B,u} = \frac{N_{A,u} N_{B,u}}{N_{A,u} N_{B,u} + N_{A,u} N_{D,u} + N_{C,u} N_{B,u} + N_{C,u} N_{D,u}}, \quad (4)$$

and likewise for $W_{A,D,u}$, $W_{C,B,u}$, and $W_{C,D,u}$. Here, $N_{A,u}$, $N_{B,u}$, $N_{C,u}$, and $N_{D,u}$ denote the numbers of pixels corresponding to the color u in the subregions A, B, C, and D.

The zooming factor corresponding to the color u is then defined as

$$z_{u,x}^{n+1} = \frac{d_{u,x}^{n+1}}{d_{u,x}^n}. \quad (5)$$

Here, $d_{u,x}^n$ represents the weighted average of the distances $d_{A,B,u,x}^n$, $d_{A,D,u,x}^n$, $d_{C,B,u,x}^n$, and $d_{C,D,u,x}^n$ in the current frame, which is computed in the same way as $d_{u,x}^{n+1}$ using the centroids in the current frame. The zooming factor $z_{u,x}^{n+1}$ is an estimate of the resizing ratio of the target's width with respect to the color u . The zooming factor with respect to the y-axis can be obtained in the same way. The pair of zooming factors $z_{u,x}^{n+1}$ and $z_{u,y}^{n+1}$ can be used as the estimates of the resizing ratios of the target, where $z_{u,x}^{n+1}$ can be used as the resizing ratio of the width and $z_{u,y}^{n+1}$ as the resizing ratio of the height.

However, it is not guaranteed that the centroid that corresponds to a certain color u is reliable. Therefore, we perform filtering on the estimated zooming factors to obtain a pair of reliable zooming factors. The filtering first trims off all the extreme values because the size of the target cannot change dramatically. After the extreme zooming factor values have been trimmed off, the remaining zooming factor values are averaged. The filtering suitable for this kind of problem is the α -trimmed mean filtering,¹⁴ which is performed by first sorting the remaining zooming factor values to get

$$\{z_x^n(1), \dots, z_x^n(k)\}, \{z_y^n(1), \dots, z_y^n(k)\}, \quad (6)$$

where $z_x^n(1)$ and $z_y^n(1)$ represent the minimum values, and $z_x^n(k)$ and $z_y^n(k)$ represent the maximum values. Then, the α -trimmed mean value is obtained by

$$z_x^n = \frac{1}{k - 2[\alpha k]} \sum_{j=[\alpha k]+1}^{k-[\alpha k]} z_x^n(j), \quad (7)$$

where $[\cdot]$ represents the greatest integer part and $0 \leq \alpha < 0.5$.

2.3 Scale-Adaptive Tracking Algorithm

The principle steps of the tracking algorithm that combines the centroid-based target shifting with the centroid-based scale adaptation can be described as follows:

1. Initialize the target window in the initial frame by any motion detection algorithm.
2. Set a search window having the same center as the target window but with a larger size in the next frame and compute the color centroids within this window.



Fig. 2 Experimental results with a cluttered background: (a) Result with the Camshift algorithm, (b) result with the mean-shift-particle filtering, (c) result with the mean-shift blob algorithm, (d) result with Kwon-Lee (Ref. 9) algorithm, and (e) result of the proposed algorithm.

3. Using the computed centroids, estimate the location of the target in the next frame and shift the target window to the estimated location.
4. Partition the target window into four subregions with respect to the center and calculate all the zoom factors in the four subregions.
5. Filter the zoom factors to obtain a pair of reliable zoom factors, and resize the target window according to the zoom factors.
6. Acquire the next frame and repeat steps 2–5.

3 Experimental Results

We performed several experiments and compared the results to the Camshift,¹ the meanshift blob,³ the mean-shift particle filtering,¹⁰ and the Kwon-Lee algorithm,⁹ which we chose as a representative of patch-based tracking. For all the algorithms, the background colors were not removed when obtaining the initial target colors. Figure 2 shows the experimental results when the background is cluttered. This is a very difficult case for color-based tracking, because colors in the background similar to the target's colors are included

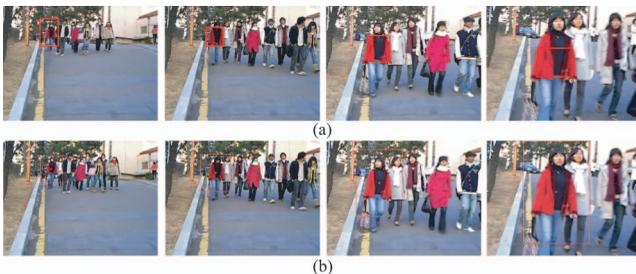


Fig. 3 Experimental results with fast enlargement: (a) Result with Kwon-Lee (Ref. 9) algorithm and (b) result with proposed algorithm.

Table 1 Summarization of the localization and scaling errors.

Sequence	Algorithm	Localization error (distance)	Scale error (%)
rainy	Camshift	181.98	367.2
	Mean-shift particle	70.34	70.82
	Mean-shift blob	71.98	73.2
	Kwon-Lee method ^a	68.98	69.4
	Proposed	21.42	15.86

^aReference 9.

in the target window. In this case, all the algorithms except the proposed one fail in the tracking. Figure 3 shows the case where the object experiences fast enlarging with a complex background. The mean-shift blob and mean-shift particle-filtering tracking fail in this case and are not shown here. The Kwon-Lee algorithm can track the object with a small initial target window, including not too much of the background colors. In comparison, the proposed scheme succeeds in the tracking even with a rather large target window. Table 1 shows the quantitative error results for the rain sequence.

Acknowledgments

This work was supported by the National Research Foundation of Korea grant funded by the Korea government (MEST) (Grant No. 2011-0000096) and The “Dongseo Frontier Project” Research Fund of 2009 by Dongseo University.

1. G. R. Bradski, “Computer vision face tracking for use in a perceptual user interface,” *Intel Technol. J.* **2**, 1–15 (1998).
2. D. Comaniciu, V. Ramesh and P. Meer, “Kernel-based object tracking,” *IEEE Trans. Pattern Anal. Mach. Intell.* **25**, 564–575 (2003).
3. R. Collins, “Mean-shift blob tracking through scale space,” presented at *IEEE Conf. on Computer Vision and Pattern Recognition* (2003).
4. A. Yilmaz, “Object tracking by asymmetric kernel mean shift with automatic scale and orientation selection,” presented at *IEEE Conf. on Computer Vision and Pattern Recognition* (2007).
5. C. Yang, R. Duraiswami, and L. Davis, “Efficient mean-shift tracking via a new similarity measure,” presented at *IEEE Conf. on Computer Vision and Pattern Recognition* (2005).
6. G. Hager, M. Dewan, and C. Stewart, “Multiple kernel tracking with SSD,” presented at *IEEE Conf. on Computer Vision and Pattern Recognition* (2004).
7. J. Jeyakar, R. Venkatesh Babu, K. R. Ramakrishnan, “Robust object tracking using local kernels and background information,” presented at *IEEE Int. Conf. on Image Processing* (2007).
8. F. Porikli and O. Tuzel, “Multi-kernel object tracking,” in *Proc. of IEEE Int. Conf. Multimedia and Expo*, pp. 1234–1237 (July 2005).
9. J. Kwon and K. Lee, “Tracking of a non-rigid object via patch-based dynamic appearance modeling and adaptive basin hopping monte carlo sampling,” presented at *IEEE Conf. on Computer Vision and Pattern Recognition* (2009).
10. A. Naeem, S. Mills, and T. Pridmore, “Structured combination of particle filter and kernel mean-shift tracking,” in *Proc. Int. Conf. Image and Vision Computing*, New Zealand (2006).
11. J. Fan, Y. Wu, and S. Dai, “Discriminative spatial attention for robust tracking,” presented at *Euro. Conf. on Computer Vision* (2010).
12. H. Wang, D. Suter, K. Schindler, and C. Shen, “Adaptive object tracking based on an effective appearance filter,” *IEEE Trans. Pattern Anal. Mach. Intell.* **29** (9), 1661–1667 (2007).
13. S. H. Lee and M. G. Kang, “Motion tracking based on area and level set weighted centroid shifting,” *IET Comput. Vis.* **4** (2), 73–84 (June 2010).
14. R. Oten and R. Figueiredo, “Adaptive alpha-trimmed mean filters under deviations from assumed noise model,” *IEEE Trans. Image Process.* **13**, 627–639 (2004).