Chapter 51
An Object Tracking Approach Based on Hu Moments and ABCshift

Xingye Wang, Zhenhai Wang and Kicheon Hong

Abstract Robust visual tracking has become an important topic in the field of computer vision. The integration of cues such as color, shape features has proved to be a promising approach to robust visual tracking. In this paper, an algorithm is presented which integrates Hu moments and color histogram. Moreover, this paper integrates the ABCshift algorithm to overcome color features drawbacks which easily lead to loss of target object when the color of object is similar to the color of background. The proposed algorithm has been compared with other trackers using challenging video sequences. Experimental work demonstrates that the proposed algorithm has strong robust and improves the tracking performance.

Keywords Object tracking · Hu invariant moment · ABCshift · Fusion

51.1 Introduction

Object tracking is a challenging problem in computer vision. It is widely applied in various fields, such as intelligence surveillance and monitoring [1], perceptual user interfaces [2], smart rooms, smart city [3], and video compression etc.

In recent years, many object tracking approaches have been proposed, for example, Mean shift [4], Camshift [5] and particle filter [6]. Among of these, Mean shift and Camshift have been widely adopted because of their relative simplicity and low computational cost. Color is used as a feature for histogram-based

X. Wang · Z. Wang (✉)
School of Informatics, Linyi University, Linyin, Shandong, China
e-mail: lywzhh@163.com

K. Hong
Department of Information & Telecommunications Engineering, The University of Suwon
Wau-ri, Bongdam-eup, Hwaseong-si, Gyeonggi-do 445-743, Korea
e-mail: kchong@suwon.ac.kr

W. E. Wong and T. Ma (eds.), *Emerging Technologies for Information Systems, Computing, and Management*, Lecture Notes in Electrical Engineering 236, DOI: 10.1007/978-1-4614-7010-6_51,

appearance representation in these methods. Color histogram is scale and rotation invariant and it can handle occlusion in a certain degree. As global statistic information, color histogram does not provide discriminative localization ability [7]. When the color of target is similar to the color of background, tracking will become unstable and sometimes even lead to loss of target object.

A good tracking algorithm should be able to work well in various difficult situations such as various illuminations, background clutter, and occlusion. There are two technique trends in the computer vision tracking community. One is to develop more inherently robust algorithms and another is to employ multiple cues to enhance tracking robustness. To increase the robustness and generality of tracking, various image features must be employed. Every single cure has its own advantages and disadvantages [8].

In this paper, we propose an object tracking method fusing color feature and shape feature that overcome the above mentioned drawbacks of Mean shift or Camshift. Hu invariant moments [9] are the one of region based features and they are a very popular shape measure. They are invariant to translation, scale change and rotation. Moreover, its computation is simply. To improve the robustness of tracking, we use adaptive background Camshift (ABCshift) [10] as a kernel tracker.

The paper is organized as follows. Section 51.2 introduces the Hu moments. The Camshift and ABCshift algorithm is presented in Sect. 51.3. The proposed object tracking approach is investigated in Sect. 51.4. The experimental results and conclusion are finally described in Sects. 51.5 and 51.6, respectively.

51.2 Hu Invariant Moments

51.2.1 Extraction for Hu Moments

For an image $f(x, y)$, size is $M \times N$. The central moment of order $(p + q)$ is defined as:

$$m_{pq} = \sum_{x=1}^{M} \sum_{y=1}^{N} x^p y^q f(x, y), \quad p, q = 0, 1, 2, \cdots \tag{51.1}$$

Central moments are defined as:

$$u_{pq} = \sum_{x=1}^{M} \sum_{y=1}^{N} (x - \bar{x})^p (y - \bar{y})^q f(x, y) \tag{51.2}$$

where $\bar{x} = m_{10}/m_{00}$ and $\bar{y} = m_{01}/m_{00}$. The normalized central moments are denoted by η_{pq}. η_{pq} is defined as $\eta_{pq} = u_{pq}/u_{00}^{\gamma}$, where $\gamma = (p + q)/2 + 1$, $p + q = 2, 3, \cdots$.

A set of seven invariants moments is defined as:

$$\phi_1 = \eta_{20} + \eta_{02} \tag{51.3}$$

$$\phi_2 = (\eta_{20} - \eta_{02})^2 + 4\eta_{11}^2 \tag{51.4}$$

$$\phi_3 = (\eta_{30} - 3\eta_{12})^2 + (3\eta_{21} - \eta_{03})^2 \tag{51.5}$$

$$\phi_4 = (\eta_{30} + \eta_{12})^2 + (\eta_{21} + \eta_{03})^2 \tag{51.6}$$

$$\begin{aligned}\phi_5 = {} & (\eta_{30} - 3\eta_{12})(\eta_{30} + \eta_{12})[(\eta_{30} + \eta_{12})^2 - 3(\eta_{21} + \eta_{03})^2] \\ & + (3\eta_{21} - \eta_{03})(\eta_{21} + \eta_{03})[3(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2]\end{aligned} \tag{51.7}$$

$$\begin{aligned}\phi_6 = {} & (\eta_{20} - \eta_{02})\left[(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2\right] \\ & + 4\eta_{11}(\eta_{30} + \eta_{12})(\eta_{21} + \eta_{03})\end{aligned} \tag{51.8}$$

$$\begin{aligned}\phi_7 = {} & (3\eta_{21} - \eta_{03})(\eta_{30} + \eta_{12})[(\eta_{30} + \eta_{12})^2 - 3(\eta_{21} + \eta_{03})^2] \\ & + (3\eta_{12} - \eta_{30})(\eta_{21} + \eta_{03})[3(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2]\end{aligned} \tag{51.9}$$

51.2.2 Hu Moments Similarity Measure

$S_o = \{S_{ok} | k = 1, 2, \cdots 7\}$ denotes the feature vector of query image. $S_i = \{S_{ik} | k = 1, 2, \cdots 7\}$ represents the feature vector of *i*th image in the image feature database. We compute the dissimilarity value between Hu moments of any two images by Euclidean distance:

$$dist(i) = \sqrt{\sum_{k=0\cdots7} (S_{ok} - S_{ik})^2} \tag{51.10}$$

51.3 Camshift Algorithm and ABCshift Algorithm

51.3.1 Camshift Algorithm

Camshift is an object tracking method which is a modification of Mean Shift tracking method. Mean Shift itself is a robust nonparametric technique for finding the peak in a probability distribution. Camshift can deal with dynamically changing color probability distribution which is taken from the video frames. Because RGB color models are much more sensitive to lighting changes, so this algorithm converts RGB color space to HSV color space in order to decrease illumination influence to tracking object. They use the HSV color system and using only hue component to make the object's color 1D histogram. This histogram is stored to convert next frames into corresponding probability of the object. The probability distribution image itself is made by back projecting the 1D hue histogram to the hue image of the frame. Camshift is then used to track the object based on this backproject image.

The Camshift algorithm is shown as below:

Step 1: Choose the initial region of interest, which contains the object we want to track.
Step 2: Make a color histogram of that region as the object.
Step 3: Make a probability distribution of the frame using the color histogram. As a remark, in the implementation, they use the histogram back projection method.
Step 4: Based on the probability distribution image, find the center mass of the search window using Mean shift method.
Step 5: Center the search window to the point taken from step 4 and iterate step 4 until convergence.
Step 6: Process the next frame with the search window position from the step 5.

Figure 51.1 shows the CAMSHIFT Algorithm [5].

51.3.2 ABCshift Algorithm

For each frame of an image sequence, the Camshift algorithm looks at pixels which lie within a subset of the image defined by a search window. The Camshift algorithm will be failed when the tracked object moves across regions of background with which it shares significant colors.

ABCshift is an Adaptive Background Camshift algorithm. It can be continuously relearned for every frame by using a background model.

The object location probabilities can now be computed for each pixel using Bayes' law as:

$$P(O|C) = \frac{P(C|O)P(O)}{P(C)} \tag{51.11}$$

where $P(O|C)$ denotes the probability that the pixel represents the tracked object given its color, $P(C|O)$ is the color model learned for the tracked object and $P(O)$ and $P(C)$ are the prior probabilities that the pixel represents object and has the color C respectively.

The denominator of Eq. (51.11) can be expanded as

$$P(C) = P(C|O)P(O) + P(C|B)P(B) \tag{51.12}$$

where $P(B)$ denotes the probability that the pixel represents background.

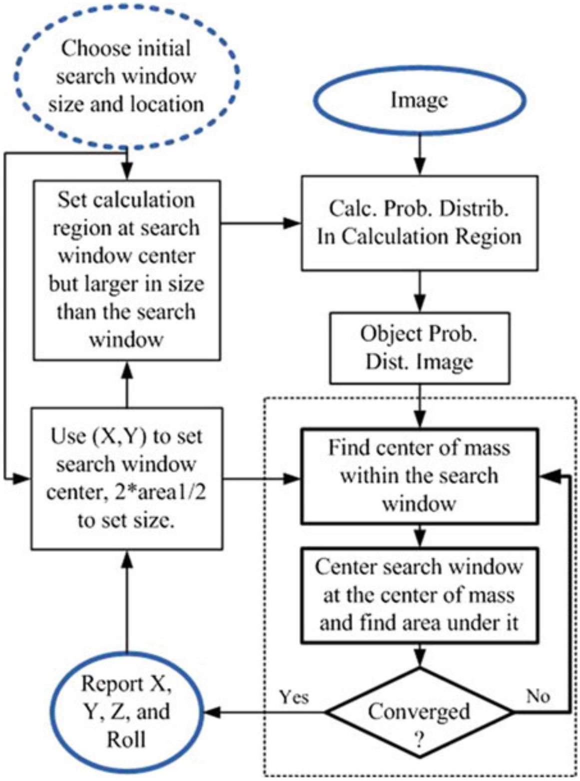

Fig. 51.1 The flowchart of CAMSHIFT algorithm

This paper assigns values to object priors in proportion to their expected image areas. If the search window is resized to be r times bigger than the estimated tracked object area, then $P(O)$ is assigned the value $1/r$ and $P(B)$ is assigned the value $(r - 1)/r$.

When object target enters into a background area which color is similar to a kind of color of the object, $P(C)$ values will increase. But $P(C|O)P(O)$ remains static. So, $P(C|O)$ values diminished.

The tracker will adaptively learn to ignore object colors which are similar to the background and instead tend to focus on those colors of the object which are most dissimilar to whatever background is currently in view.

51.4 The Proposed Object Tracking Approach

The proposed tracking algorithm combines the shape and color features through calculate the similarity between template region and candidate region. The likelihood function is defined as follow:

$$S = \alpha\rho_c + (1 - \alpha)d_{hu} \tag{51.13}$$

Color feather is scale and rotation invariant. It is more robust and stable than shape feature in tracking of colored objects. So, where $\alpha \in [0.5, 1]$, which is defined as the reliability factors for color features. ρ_c is the Bhattacharyya distance of ABCshift algorithm. d_{hu} is the distance of Hu moments. In this paper, the value of the factor α is set according to scene. It will be set smaller if the background includes clutter or the object appearance has geometric change, otherwise it will be set bigger.

The proposed algorithm is summarized as:

(1) Identify an object region in the first image and train the object model, $P(C|O)$. Computes the Hu moments.
(2) Center the search window on the estimated object centroid and resize it to have an area r times greater than the estimated object size.
(3) Learn the color distribution, $P(C)$ by building a histogram of the colors of all pixels within the search window. Calculate the Hu moments of the search window and the distance of Hu moments between template window and search window.
(4) Use Bayes' law, Eq. (51.11) to assign object probabilities, $P(O|C)$, to every pixel in the search window, creating a 2D distribution of object location.
(5) Estimate the new object position as the centroid of this distribution and estimate the new object size (in pixels) as the sum of all pixel probabilities (in pixels) as the sum of all pixel probabilities within the search window.
(6) Compute the Bhattacharyya metric between the distributions, $P(C|O)$ and $P(C)$. If this metric is less than a preset threshold then enlarge the estimated object size by a factor r.
(7) Calculate the likelihood between template windows with search window according to Eq. (51.13).
(8) Repeat steps 2–7 until the object position estimate converges.
(9) Return to step 2 for the next image frame.

51.5 Implementation and Experiments

To check the effectiveness of the proposed approach, we have implemented and tested it on a wide variety of challenging image sequences in different environments and applications. The experiments are performed on a computer which has an Inter(R) Core(TM) i3 processor (3.10 GHZ) with 3.00 GB memory. Our solution and other compared solutions are implemented in VC++ language.

In Fig. 51.2, the human head in a video sequence from http://vision.stanford.edu/~birch/headtracker/seq/ is moving to the left and right very quickly. The illumination on the face also changes. The background scene includes clutter and material of similar color to the face. Object turns around and is spatially occluded. As we can see in Fig. 51.2, at frame 33, the Camshift tracker fails to track because the background color is similar to the face. From frame 64, Camshift is trapped in

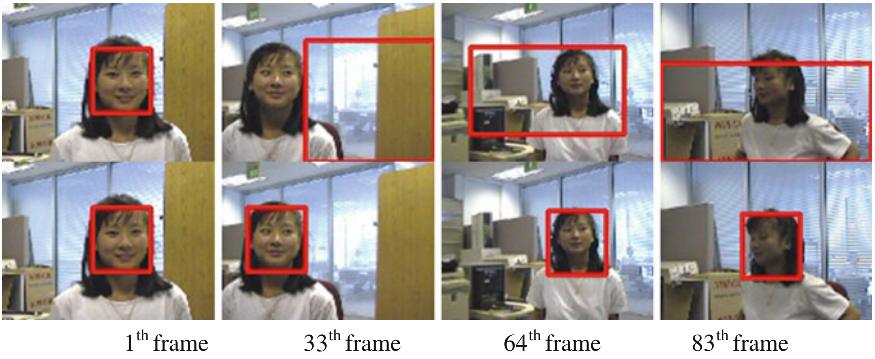

Fig. 51.2 Tracking results of the girl head moving sequence with Camshiftt tracker (*first row*) and our method (*second row*)

Fig. 51.3 The comparison of tracking result: Mean shift (*red rectangle*), our method (*blue rectangle*)

a false region. The tracker fails for most of the remaining frames because the object is distracted by similar color region. In our method, because fuse the shape features and colour features and ABCshift algorithm can adaptively adjust object probability distribution according to the background, the object never loses the target and achieves the most accurate results.

In Fig. 51.3, green rectangle shows the search window of Mean shift algorithm, red rectangle shows the result of Mean shift tracking. Blue rectangle shows the tracking result of proposed algorithm. Initial select object include two parts which have difference colours. When the person who wears a red shift enters to the region which includes red brick walls and doors, the Mean shift tracking lose the tracked person. Adaptive background method can successfully tracks throughout the sequence and is not distracted by red regions of background.

51.6 Conclusion

To increase tracking robustness and accurateness, the paper proposed an object tracking method based on combining Hu moments and ABCshift algorithm. It can not only overcome color features drawback, but also can adaptive background color changing. Experiments show the proposed method has better performance than Mean shift or Camshift. Similarly to Mean shift and Camshift, proposed algorithm will fail to rapidly move object. In future work, the research attempt to investigate integration with Kalman filter or particle filter and adaptive features fusion mechanism to improve tracking robustness.

Acknowledgments This work was supported by the GRRC program of Gyeonggi Province, Korea [(GRRC SUWON 2012-B3), Development of Multiple Objects Tracking System for Intelligent Surveillance]. This research was also supported in part by Science and technology development planning project (No. 2012YD01052), Shandong province, China. Their support is gratefully acknowledged.

References

1. Yilmaz, A., Javed, O., Shah, M.: Object tracking: a survey. ACM Comput. Surv. **38**(4), 1–45 (2006)
2. Ikizler, N.: Human action recognition with line and flow histograms. In: Proceedings of the Pattern Recognition, ICPR 2008, pp. 1–4 (2006)
3. Qi, W., Zhang, X., et al.: A robust approach for multiple vehicles tracking using layered particle filter. Int. J. Electron. Commun. **65**(7), 609–618 (2011)
4. Comaniciu, D., Ramesh, V., Meer, P.: Kernel-based object tracking. IEEE Trans. Pattern Anal. Mach. Intell. **25**(5), 564–577 (2003)
5. Bradski, G.: Computer vision face tracking for use in a perceptual user interface. Intel Technol. J. **2**(2), 13–27 (1998)
6. Isard, M., Blake, A.: CONDENSATION–conditional density propagation for visual tracking. Int. J. Comput. Vis. **29**(1), 5–28 (1998)
7. Wang, Z., et al.: Shape based appearance model for kernel tracking. Image Vis. Comput. **3**(3), 1–13 (2012)
8. Liu, H., Ze, Y., Zha, H., et al.: robust human tracking based on multi-cue integration and mean-shift. Pattern Recogn. Lett. **30**, 827–837 (2009)
9. Hu, K.M.: Visual pattern recognition by moment invariant. IEEE Trans. Inf. Theory **8**(2), 179–187 (1962)
10. Stolkin, R., Florescu, I., Kamberov, G.: An adaptive background model for camshift tracking with a moving camera. In: Proceedings of the 6th International Conference on Advances in Pattern Recognition. World Scientific Press, Kolkata (2007)