Moving Object Tracking via Hausdorff Distance and Particle Filter

Junqing Wang[1,2], Zelin Shi[1],Shabai Huang[1]

[1] Shenyang Institute of Automation, Chinese Academy of Sciences, Shenyang, 10016,China
{wangjunqing, zlshi, HSB}@sia.cn
[2] Graduate School of Chinese Academy of Sciences, Beijing, 100039, China

Abstract

Moving object tracking is widely applied in computer vision. A novel method for moving object tracking, which utilizes particle filter and Hausdorff distance is proposed in this paper. This algorithm consists of system model, measure model, the strategy of template update with adaptive tracking window and solution to occlusion in the particle filter framework. In system model, Hausdorff distance and edge information of target are applied to improve the robustness against variation of rotation, scale, translation and illumination of target. In measure model, this new similarity metric defined based on gray histogram not only enhances tracking fault-tolerant property, but its computational cost has also been greatly reduced. The strategy of update template of adaptive tracking window and solution to occlusion makes tracking more stable and robust. The experimental results also illustrate that this algorithm is stable and efficient to track deformable objects in image sequences.

1 Introduction

Object tracking is a challenging and important problem in computer vision. Due to rotation, scale, translation and illumination etc., adaptive tracking is appealing in tracking tasks in order to attain stable and precise tracking in most tracking application. Generally, object tracking utilizes template to match target. But this is true of template-matching methods that do not adapt to appearance changes, and it is true of motion-based tracking where the appearance model can change rapidly [1].

This paper proposes a robust, adaptive template for moving object tracking based on Hausdorff distance and particle filter under complex background. This method utilizes the information of gray level and edges of target to construct dynamic tracking model so that the problem of tracking is transformed into a Bayesian estimation of the state of a dynamic system. The state space approach is convenient for handling multivariate data and non-linear and non-Gaussian process [2]. Particle filter has been proven very sucessful for non-linear and non-Gaussian estimation problems and is reliable in the case of clutter and occlusions [3].

The remainder of this paper is organized as follows. In section 2 particle filter and Hausdorff distance are briefly described. The dynamic system which includes system model and measure model is described in section 3. Template update and solution to occlusion are explained in section 3,too. In section 4 experimental results are illustrated. Finally, we summarize our conclusions.

2 Hausdorff Distance and Particle Filter

2.1 Hausdorff Distance

Hausdorff distance measures the mismatch of the two sets [4]: Given two finite point sets $A=\{a_1,\ldots,a_m\}$ and $B=\{b_1,\ldots,b_n\}$,the Hausdorff distance D between A and B is defined as

$$D=\max\{d(A,B),d(B,A)\}. \quad (\mathbf{1})$$

Where $d(A,B)$ is the distance from set A to set B and $d(B,A)$ from set B to set A expressed as

$$\begin{aligned} d(A,B) &= \max_{a\in A}\min_{b\in B}\|a-b\| \\ d(B,A) &= \max_{b\in B}\min_{a\in A}\|b-a\| . \end{aligned} \quad (\mathbf{2})$$

Where $\|\bullet\|$ generally denotes a L_p norm. Obviously Hausdorff distance D is the maximum of $d(A,B)$and $d(B,A)$, which measures the degree of mismatch between two set A and B. Hausdorff distance defined in eq.(2) presents some problems in the presence of outliers. If one feature point of the model or image is an outlier, the resulting Hausdorff distance will be very large, even if all other points perfectly match. Therefore, it is preferable to using partial Hausdorff distance, which measures the difference between a portion of the model and the image [4]. However, the direct comparison method using Hausdorff distance is time-consuming, some speedup techniques for computation have developed. It is important that the distance transform used to reduce computational cost in matching algorithms should produce reasonably good approximation of the Euclidean distance as the matching measure is computed from the distance values. So, Chamfer-distance is used as approximation of Hausdorff distance [11].

2.2 Particle Filter

Particle filter provides a Bayesian inference framework for dynamic state estimation. It is a technique for implementing a recursive Bayesian filter by Monte Carlo simulations [5]. The key idea is to represent the required posterior density function (Pdf) by a set of random samples with associated weights and to compute estimates based on these samples and weights. As the number of samples becomes very large, this Monte Carlo characterization becomes an equivalent representation to the usual functional description of the Pdf

Let $\{x_k^i, w_k^i, i = 1 .... N\}$ denote a random measure that characterizes the Pdf $P(X_k|z_{1...k})$ at current time k, where $\{x_k^i, i = 1 .... N\}$ is a set of sample points with associated weights $\{w_k^i, i = 1 .... N\}$, X_k is the current object state at time k and $\{z_{1...k}\}$is all observations up to time k. The weights are normalized such that $\sum_i w_k^i = 1$. These sample points propagate from current state to next state according to a system model. Each sample is weighted in term of the observations with probability $w_k^i = P(X_k | z_{1...k})$. The mean state of an object is estimated at each time step according to

$$E(S) = \sum_{i=1}^{N} w_k^i x_k^i \tag{3}$$

3 Tracking Model

From a Bayesian perspective, the tracking problem is to recursively calculate some degree of belief in the state X_k at time k, taking different values, given the data $z_{1...k}$ up to time k. Thus, it is assumption that the initial Pdf of the state vector, $P(X_0|z_0)=P(X_0)$, also known as the prior, is available(z_0 being the set of no measurements).Then, in principle, the Pdf $P(X_k|z_{1...k})$ may be obtained recursively in two stages: prediction and update.

3.1 System Model

The prediction stage involves using the system model, also known as state transition model, to be obtained by $P(X_k|X_{k-1})$ which describes a Markov process of order one. This paper utilizes edges of target and inter-frame relation in images sequence to construct system model which is described by $P(X_{k+1}|X_k)$. Inter-frame relation illustrates the fact that the motion or deformation of the object between successive frames is not significantly large so that the object in the next frame is in the neighborhood of the object in the current frame [6]. At time k, centroid, width and height of the object are defined as state vector of the object, $X_k = \{(x_k,y_k),W_k,H_k\}$, where (x_k,y_k) denotes centroid of the object ,W_k denotes width of the object, H_k denotes height of the object. Edges of the object are regarded as observation data of the object state. According to the above analysis, it is a fact that the centroid, (x_k,y_k), of the object in current frame will appear in the neighborhood of it in the next frame. Here, the neighborhood of (x_k,y_k) in the next frame is 5*5. So there are 25 possible state transitions that are defined as $\{ X^i_{k+1}= (x^i_{k+1},y^i_{k+1}),W^i_{k+1},H^i_{k+1}, i=1…25\}$ from the current frame to the next frame .

Hausdorff distance of edge pixels is used to describe state transition model and is approximated by chamfer distance because it can achieve robustness against rotation, scale, translation and partial occlusion.

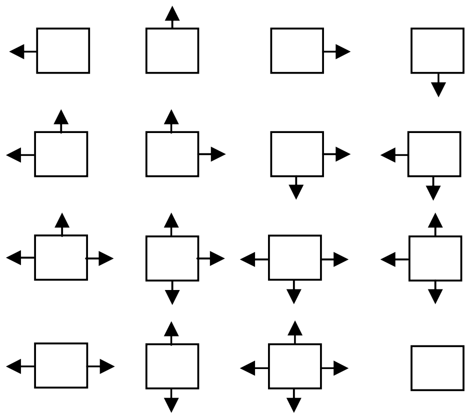


Figure 1 All kinds of zooming in tracking window

For the sake of simplicity, the system model is denoted by single direct Hausdorff distance, $D_T(p_1,p_2)$,which represents the distance of edge pixel p_1 of template to the nearest edge pixel p_2 of matching sub-region in image through chamfer distance. Due to probable object deformation, tracking window should also adapt to changing. All variations of tracking window are classified as zoom in and zoom out. All kinds of various cases of zooming in tracking window are shown in figure 1,where arrowhead denotes direction of zooming in tracking window, and the cases of zooming out tracking window are analogous to the cases of zooming in and omitted. Supposed that system model, $P(X_{k+1}|X_k)$, obeys Gaussian distribution. So $P(X_{k+1}|X_k)$ is defined as eq.(4).

$$P\left(X_{k+1}^{j} \mid X_k\right) = \max_i \left\{ \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{D_j^{ith}}{2\sigma^2}}, \quad i = 1...31 \right\} \quad j = 1...25$$
$$W_{k+1}^{j} = W_{k+1}^{j,i}$$
$$H_{k+1}^{j} = H_{k+1}^{j,i} \qquad \textbf{(4)}$$

Where D_j^{ith} denotes the j-th smaller Hausdorff distance which is computed through chamfer transformation from the state X_k to X^j_{k+1}, i denotes 31 kinds of possibly variable cases of tracking window. All of $P\left(X_{k+1}^{j} \mid X_k\right), j = 1...25$ need be normalized to obtain 25 transition probability weights defined as eq.(5).

$$w_j = \frac{P\left(X_{k+1}^{j} \mid X_k\right)}{\sum_{i=1}^{25} P\left(X_{k+1}^{i} \mid X_k\right)} \qquad j = 1\ldots\ldots 25 \tag{5}$$

3.2 Measure Model

The update stage applies measure model to obtain $P(z_k|X_k)$ which describes the likelihood through the observation z_k under the state X_k. Trackers using gray level reference models have been proved robust and versatile for a modest computational cost. They have in particular been proved to be very useful for tracking tasks, where the objects of interest can be of any kind, and exhibit in addition drastic changes of spatial structure through the sequence, due to pose changes, partial occlusion, etc. which arise in the context of video analysis and manipulation. For such application, most trackers based on a space-dependent appearance reference would break down very fast. In contrast, using gray level distribution within the region of interest is an appealing way to address such complex tracking tasks [7]. So the measure model, $P(z_k|X_k)$, is described based on gray level histogram distance which compares the gray level content of candidate regions to a reference gray level histogram.

We define a probabilistic metric based on the histogram distance between a candidate region and a reference model as eq.(6) which is different from [8] based on Bhattacharyya similarity coefficient.

$$P\left(z_k \mid X_k\right) = p(h_T, h_k) = \frac{1}{255}\sum_{i=1}^{255}\frac{\min\{h_T^i, h_k^i\}}{\max\{h_T^i, h_k^i\}} \tag{6}$$

Where $h_T = \left\{h_T^i, i = 1\ldots 255\right\}$ and $h_k = \left\{h_k^i, i = 1\ldots 255\right\}$ denote histograms of template and candidate region associated with a hypothesized state X_k at time k respectively. This probabilistic metric has advantage of fuzzy metric and less computational cost than Bhattacharyya similarity coefficient. Given the state X_k at time k, the probability of the object appearing at the state X^i_{k+1} is $P(X^i_{k+1}| X_k)$.Given the observation z^i_{k+1}at the state X^i_{k+1},the likelihood that the object appears at X^i_{k+1} is the distribution $P(z^i_{k+1}|X^i_{k+1})$.According to Bayesian rule, the Pdf is $P\left(X_{k+1}^i \mid z_{k+1}^i\right) \propto P\left(z_{k+1}^i \mid X_{k+1}^i\right) P\left(X_{k+1}^i \mid X_k\right)$. So the weight π_{k+1}^i associated with the state X^i_{k+1} may be defined as eq.(7)

$$\pi_{k+1}^i = w_{k+1}^i P\left(z_{k+1}^i \mid X_{k+1}^i\right) \tag{7}$$

Where w^i_{k+1} and $P(z^i_{k+1}| X_k)$can be obtained by eq.(5) and eq.(6), respectively. Therefore, the mean state X_{k+1}at time k+1 can be obtained through eq.(8).

$$x_{k+1} = \sum_{i=1}^{25} \pi_{k+1}^{i} x_{k+1}^{i}$$
$$y_{k+1} = \sum_{i=1}^{25} \pi_{k+1}^{i} y_{k+1}^{i}$$
$$W_{k+1} = \sum_{i=1}^{25} \pi_{k+1}^{i} W_{k+1}^{i} \quad \textbf{(8)}$$
$$H_{k+1} = \sum_{i=1}^{25} \pi_{k+1}^{i} H_{k+1}^{i}$$

3.3 Template Update and Solution to Occlusion

In order to stably track deformable moving object, the template need be updated by means of the updating rule. This paper defined the updating rule as eq.(9) which preserves certain edge points in the current template and select certain new edge pixel from the next frame. Suppose that T_k is the current template at time k and T_{k+1} is the template after updating at time k+1.

$$T_{k+1} = \begin{cases} p_1 \ replaced \ with \ p_2 & D_T(p_1, p_2) \le 3 \ and \ D^{fth}(T_k, I_{k+1}) \le th \\ T_k & D^{fth}(T_k, I_{k+1}) > th \end{cases} \quad \textbf{(9)}$$

Where $D_t(p_1,p_2)$ denotes Hausdorff distance between the template pixel p_1 and the pixel p_2 in the matching image I_{k+1}, and $D^{f\text{th}}(T_k,I_{k+1})$ denotes the f-th largest Hausforff measure between template T_k and I_{k+1},th is predefined threshold which is computed by averaging Hausdorff distance of all pixels of template.

In process of tracking, occlusion may happen. The template does not update and is remembered for a while such as [9] when the object is occluded. The occlusion is declared when Hausdorff measure $D^{f\text{th}}(T_k,I_{k+1})$ is more than some threshold. Let t_1 be the time of happening occlusion and L be the maximal duration of the occlusion. If the $D^{f\text{th}}(T_k,I_{k+1})$ is still more than a given threshold at the time t_1+L, it indicates that the occlusion does not end, then the tracking system need return initialized state.

4 Experimental results

We use this algorithm (HDPF) to test two image sequences containing 150 frames and 200 frames, respectively. Templates were manually initialized at the beginning as the bounding box of the object. Figure 2(b) and Figure 3(b) show some tracking results of the two image sequences. Some results of template update are presented in Figure (a) and Figure 3(a). In the two experiments, the moving objects can be reliably tracked.

To evaluate the performance of this algorithm, the same datasets are applied for the tracking algorithms based on Hausdorff distance (HD)[10] and the sum of squared linear differences (SSD). Tracking window using the algorithm of HD happens to drift towards background both at the 22th frame of sequence one and at the 61th frame of sequence two, and lost target at last. Tracking window using the algorithm of SSD happens to drift towards background both at the 15th frame of sequence one and at the 37th frame of sequence two, and lost target at last. The proposed algorithm always stably tracks the moving object in these two image sequences. Here, the tracking rate is defined as eq.(10). Figure 4(a) and (b) display the performance of these three algorithms for image sequences one and two. This shows that the proposed algorithm is superior over the other two algorithms.

$$\text{Tracking rate} = \frac{\text{Number of right matching frames}}{\text{Number of all frames}} \times 100\% \tag{10}$$

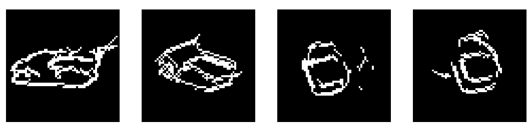


Figure 2(a) some update templates in experiment one

Figure 2(b) Some tracking results in experiment one

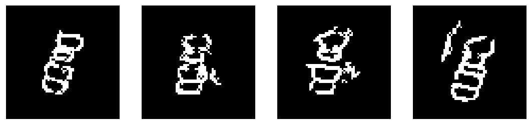


Figure 3(a) some update templates in experiment two

Figure 3(b) Some tracking results in experiment two

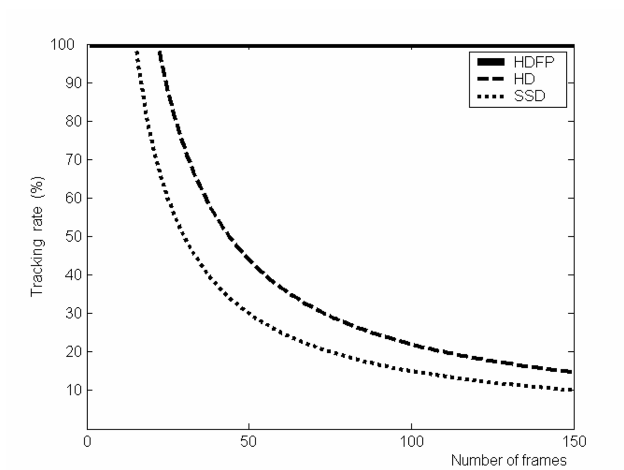


Figure 4 (a) Performance comparison of three algorithms in experiment one

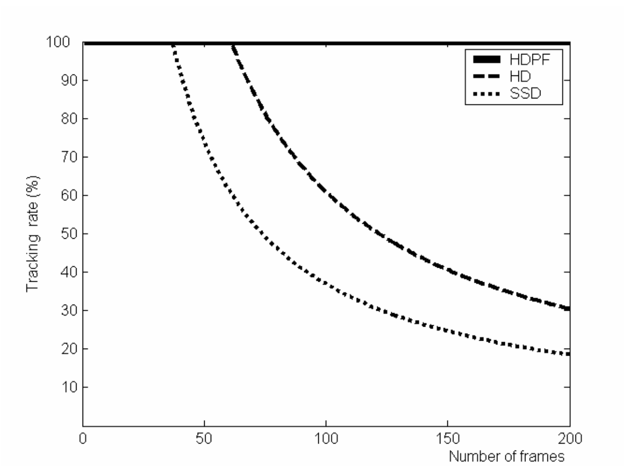


Figure 4 (b) Performance comparison of three algorithms in experiment two

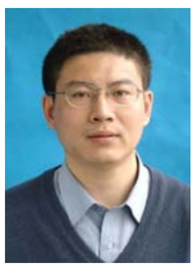
5 Conclusion

This algorithm, which consists of system model, measure model and strategy of template update and solution to occlusion, utilizes particle filter and Hausdorff distance measure to track moving object in image sequences. Because Hausdorff distance measure and edge information are integrated into tracking framework, this enhances stability for tracking. Because a novel similar metric based on histogram is defined in measure model, this not only enhances matching fault-tolerant property, but its calculation quantity has also been greatly reduced. Because the strategy of update template of adaptive tracking window and solution to occlusion is applied, this makes tracking more stable and robust. The experimental results also illustrate that this algorithm is stable and efficient to track deformable objects in image sequences.

References

1. A.D. Jepson, D.J. Fleet, and T.F. El-Maraghi. Robust online appearance models for visual tracking. IEEE Trans. Pattern Anal. Mach. Intell. 25(10): 1296-1311 (2003)
2. A. Doucet, N. De Freitas, and N.Gordon. Sequential Monte Carlo Methods in Practice. Spriner-Verlag, New York,2001.
3. M. Isard and A. Blake. Condensation-conditional density propagation for visual tracking. Int. J. Computer Vision, V.29(1), pp.5-28,1998.
4. D.P. Huttenlocher, G.A. Klanderman, and W.J. Rucklidge. Comparing images using the Hausdorff distance. IEEE Trans. Pattern Anal. Mach. Intell. 15(9):850 –863(1993).

5. Arulampalam, S. Maskell, N. Gordon and T. Clapp. A Tutorial on ParticleFilters for On-line Non-linear/ Non- Gaussian Bayesian Tracking. IEEE Trans. Signal Processing, 50(2): 174 -188 (2002).
6. Zong Y., Jain A. K. Object tracking using deformable templates. IEEE Trans. Pattern Anal. Mach. Intell. 22 (5): 544-549(2000).
7. P. Pérez, C. Hue, J. Vermaak and M. Gangnet. Color-Based Probabilistic Tracking. European Conference on Computer Vision, pp.661-675,2002.
8. K. Nummiaro, E. Koller-Meier, L. Van Gool. Color Features for. Tracking Non-Rigid Objects. ACTA Automatica Sinica, 2003.
9. HT Nguyen , M. Worring and R. van den Boomgaard. Occlusion Robust Adaptive Template Tracking. Proc. Int'l Conf. Computer Vision, V.1, pp. 678-683, 2001.
10. Thomax M., Ngan K. N. Automatic segmentation of moving objects for video object plane generation.IEEE Trans. Circuits and Systems for Video Technology,1998,8(5):525-538.
11. Borgefors G. Distance transformations in digital images. Computer Vision, Graphics, Image Processing, 34(3):344-371(1986).



Junqing WANG received the BS degrees in computer sciences from Jilin University in 1998.From 1998 to 2001,he worked in Jilin Company of China Telecom, Changchun, P.R. China. Now, he is a candidate for Ph.D in Shenyang Institute of Automation, Chinese Academy of Sciences. His research interest is image processing, target tracking, moving object detection.