Adaptive Multi-part Target Representation for Tracking

Mingqi Luo^{1,a}, Tuo Wang² and Bin Zhou¹

¹The school of Electronic and Information Engineering, Xi'an Jiaotong University, Xi'an, Shanxi, China

²Xi'an Jiaotong University Suzhou Academy, Suzhou, Jiangsu, China

Abstract. In this paper we propose and evaluate an effective approach based on multiple colour histograms. The target is adaptively divided into non-overlapping regions. The proposed partition does not weaken the robustness of the colour histogram representation; it can be used to any class targets. Experimental results show that the proposed representation improve the tracking accuracy and decrease the number of iterations.

1 Introduction

Visual localization and tracking plays a central role for many applications like intelligent video surveillance, smart transportation monitoring etc. Localization and tracking algorithms aim to find the most similar region to the tracking in image sequence. One of the most crucial difficulties in robust tracking is the construction of representation models that can accommodate the illumination variations, deformable appearance changes, partial occlusions, etc.

Colour histograms have been widely used to represent, analyse, and characterize images. They allow for significant data reduction, and can be computed efficiently; moreover colour histograms are robust to noise and local image transformation. In the target tracking domain, colour histogram are a popular form of target representation, because of their independence from scaling and rotation, and robust to partial occlusions [1]. Nevertheless the robustness of such a model is weakened in challenging tasks due to the lack of spatial information. This problem can be limited by computing more than one histogram on different parts of the target [2], but there is no generally accepted solution for a generic division.

In this paper we propose and evaluate an effective approach based on a multi-part model. The target is divided into non-overlapping regions in order to increase the tracking sensitivity to rotations and anisotropic scale changes. The proposed partition does not weaken the robustness of the colour histogram representation; it can be used to any class targets.

2 Mean shift tracking algorithm

Mean shift algorithm war first proposed by Fukunage and Hostetler [3] in 1975. Coumaniciu[4]employed the MS algorithm successfully in feature space analysis. MS

algorithm was also used in image smooth and image segmentation, and got good performance.

2.1 Target representation

In object tracking domain, a target is usually defined as a rectangle or an ellipsoidal region in image. Currently, a widely used target representation is the color histogram because of its independence of scaling, rotation and partial occlusion. Denote by $\{x_i\}_{i=1,\dots,n}$ the normalized pixels in target region. The probability of the feature $u(u = 1, \dots, m)$ in the target model is computed as [1]:

$$\begin{cases} q = \{q_u\}_{u=1,2,\dots,m} \\ q_u = C_h \sum_{i=1}^n k\left(\left\|\frac{x_i}{h}\right\|^2\right) \delta(b(x_i) - u) \end{cases} \quad (1)$$

Where q is the target model, q_u is the probability of the u th element of q , δ is the Kronecker delta function, $b(x_i)$ associates the pixel x_i to the histogram bin and $k(x)$ is an isotropic kernel profile. Constant C_h is a normalization function defined by:

$$C_h = \frac{1}{\sum_{i=1}^{n_h} k\left(\left\|\frac{x_i}{h}\right\|^2\right)} \quad (2)$$

Similarly, the probability of the feature u in the target candidate model from the candidate region centered at position y is given by:

$$\begin{cases} p(y) = \{p_u(y)\}_{u=1,2,\dots,m} \\ p_u(y) = C_h \sum_{i=1}^n k\left(\left\|\frac{y - x_i}{h}\right\|^2\right) \delta(b(x_i) - u) \end{cases} \quad (3)$$

^a Corresponding author: lmq232x@163.com

$$C_h = \frac{1}{\sum_{i=1}^{n_h} k \left(\left\| \frac{y - x_i}{h} \right\|^2 \right)} \quad (4)$$

Where $p(y)$ is the target candidate model, $p_u(y)$ is the probability of the u th element of $p(y)$, $\{x_i\}_{i=1, \dots, n_h}$ are pixels in the target candidate region centred at y , h is the bandwidth and C_h is a normalization function.

In order to calculate the likelihood of the target model and the candidate model, a metric based on the Bhattacharyya coefficient [5] is defined by using the two normalized histograms q and $p(y)$ as follows

$$d(y) = \sqrt{1 - \rho[\hat{p}(y), \hat{q}]} \quad (5)$$

$$\rho(y) \equiv \rho[\hat{p}(y), \hat{q}] = \sum_{b=1}^m \hat{p}_b(y) \hat{q}_b \quad (6)$$

2.2 Mean shift

Minimizing the distance $d(y)$ in (5) is equivalent to maximizing the Bhattacharyya coefficient $\rho[\hat{p}(y), \hat{q}]$ in (6). The optimization process is an iterative process and is initialized with the target position, denoted by y_0 , in the previous frame. By using the Taylor expansion, the linear approximation of the Bhattacharyya coefficient $\rho[\hat{p}(y), \hat{q}]$ can be obtained as:

$$\rho(y) \approx \frac{1}{2} \sum_{u=1}^m \sqrt{p_u(y_0) q_u} + \frac{C_h}{2} \sum_{i=1}^n \omega_i k \left(\left\| \frac{y - x_i}{h} \right\|^2 \right) \quad (7)$$

$$\omega_i = \sum_{u=1}^m \delta(b(x_i) - u) \sqrt{\frac{q_u}{p_u(y_0)}} \quad (8)$$

Since, the first term in (7) is independence of y , to minimize the distance in (5) is to maximize the second term in (7). In the mean shift iteration, the estimation target moves from y to a new position y_1 , defined as

$$y_1 = \frac{\sum_{i=1}^n \omega_i x_i g \left(\left\| \frac{y - x_i}{h} \right\|^2 \right)}{\sum_{i=1}^n \omega_i g \left(\left\| \frac{y - x_i}{h} \right\|^2 \right)} \quad (9)$$

3 Adaptive target representations

3.1 Adaptive partition

Since the colour histogram is the whole statistical characteristics of the target, its lack of spatial information, as it depends on the colour feature, it cannot give good

performance when an object and its background have similar colours.

In the case of rectangle-based trackers the adaptive partition is as follows.

$$k = \frac{w}{h} \quad (10)$$

Where w h are the width and height of the rectangle, respectively. Defined $k_0 = 2$. The adaptive partition is as Fig1:

Fig.1 Multi-part representation

When $k > k_0$, that is $w > 2h$, the moving target is partition into A1, A2 and A3; when $0.5k_0 < k < k_0$, the moving target is partition into B1, B2, B3, B4, B5, B6 and B7; when $0 < k < 0.5k_0$, the moving target is partition into C1, C2 and C3, where B6 have half width and height of the original rectangle, B7 have double width and height of the original rectangle.

3.2 The weighted Bhattacharyya coefficient

The Bhattacharyya coefficient is used to calculate the likelihood of the target model and the candidate model. Emilio Maggio and Andrea Cavallaro [6] using the average Bhattacharyya coefficient as the finally coefficient. They do not consider that the different parts have difference impact to the whole likelihood. In this paper, we adopt that the more likelihood of the same sub-part of the rectangle, the bigger weights. A commonly used weight is calculated as follow:

$$\omega'(i) = \frac{BC(p_i(y), q_i)}{\sum_{j=1}^C BC(p_i(y), q_i)} \quad (11)$$

$$\rho(y) = \sum_{i=1}^C \hat{\rho}(p_i(y), \hat{q}_i) \omega'(i) \quad (12)$$

Where $\omega(i)$ denote the weight of i -th sub-part, $BC(p_i(y), q_i)$ is the Bhattacharyya coefficient between $p_i(y)$ and q_i , C is the number of the sub-part, $p_i(y)$ and q_i are the probability distribution of target model and the candidate model, respectively, $\rho(y)$ is

the finally Bhattacharyya coefficient between target model and candidate model.

3.3 The algorithm

Given:

The target model $\{q_u^i\}_{u=1,2,\dots,m}$, $i = 1, 2, \dots, C$, and its location y_0 in the previous frame. The width w and height h of the external rectangle.

Step1. Using the width and height decide the adaptive partition, and calculate the sub-part target model.

Step 2. Initialize the location of the target in the current frame with y_0 , computer $\{p_u^i(y)\}_{u=1,2,\dots,m}$, $i = 1, 2, \dots, C$, and evaluate the Bhattacharyya coefficient $\rho[q, p(y_0)]$ between each same sub-part by(10) and (11).

Step 3. Derive the weights $\{\omega(i)\}_{i=1,2,\dots,m}$ according to (8).

Step 4. Find the next location of the target candidate according to (9).

Step 5. Compute $\{p_u^i(y)\}_{u=1,2,\dots,m}$, and the Bhattacharyya coefficient $\rho[q, p(y_1)]$ of each sub-part between $\{p_u^i(y)\}_{u=1,2,\dots,m}$ and $\{q_u^i\}_{u=1,2,\dots,m}$.

Step 6. While $\rho[q, p(y_1)] < \rho[q, p(y_0)]$

Do $y_1 = 0.5(y_1 + y_0)$

Evaluate the $\rho[q, p(y_1)]$

Step 7. If $\|y_1 - y_0\| < \varepsilon$, then stop.

Otherwise set $y_0 = y_1$ and go to step3.

4 Implementation of the algorithm

In order to select the most appropriate color feature, Emilio Maggio [6] described and evaluated eight representations. The eight representations are derived from seven color spaces: RGB, rgb, rg, CIELab, XYZ, YCbCr, and two representations based on HSV (HSV-D, HSV-UC) [7]. After the evaluating eight different color, they conclude that the RGB-based representation outperforms the others. In this paper, the RGB feature space is selected.

The parameters are the same for all test sequences, and are described in the following. The Epanechnikov profile is accepted in these experiments. To satisfy the low-computational cost imposed by read-time processing discrete densities, $16 \times 16 \times 16$ -bins histograms used in RGB feature spaces. Maximum number of iterations is 10. The results presented in this section are based on a dataset of targets extracted from 6 different test sequences, including faces, vehicles and pedestrian. All

of these test sequence is download on [8] and [9]. Fig2 showed the target initialization.

Fig.2. Target initialization

Fig.3. The tracing results

Fig.3. showed the car's tracking results. (a), (b) and (c) are the results of the traditional single-histogram tracking algorithm. (d), (e) and (f) are the results of the adaptive multi-part representation tracking results; all of the results are initialization with the same. They are the 24th, 39th and 51th, respectively. By analyzing these results, it is possible to notice that the multi-part representation improves in average the performances. Apparently, compare the frame of 39th, the single-histogram tracking algorithm does not contain the whole car, but the multi-part representation does. However, this algorithm is still some problems as single-histogram algorithm. Apparently, the scale is remaining the same throughout the tracking process.

Fig4 showed the curve of the number of iterations of different multi-part representation on faces sequences as showed in the middle of top row of the Fig2.

Fig.4. The number of iterations

In the left of the top row is the single-histogram, the right is the two-part partition representations, the left bottom is the seven-part partition representations, the right is the two-part partition representations as the series of the A,B and C showed in Fig1, respectively. The average number of iterations is 3.96, 3.15, 3.18 and 2.91, obviously, the number of iteration is decrease.

5 Conclusions

To summarize, we have proposed an adaptive multi-part target representations methods for tracking. Its adaptive decides the ways that divided the target by the size of external rectangle. The weighted Bhattacharyya coefficient is used to calculate the likelihood of the target model and the candidate model. Experimental results showed that the proposed adaptive multi-part representation is more accurate than the single histogram, and achieves better results in predicting the correct orientation. Future work includes investigating an adaptive scale change.

Acknowledgment

Research supported by National Natural Science Foundation of China (No.61071217) and Jiangsu Science and Technology Program Foundation (No.BK20141216).

References

1. D.Coumaniciu, V.Ramesh. IEEE Trans. on Pattern Analysis and Machine Intelligence,**125**,(2003)
2. P.Perez, C.Hue, J.Vermaak, ECCV,**1**(2002)
3. K.Fukunage and L.D.Hostetler, IEEE Trans.on Information Theory,**1**(1979)
4. D.Coumaniciu and P.Meer, IEEE trans.on Pattern Analysis and Machine Intelligence,**5**(2002)
5. Kailath,T., IEEE Trans. Commun.Technol.,**15**(1967)
6. Emilio Maggio and Andrea Cavallaro, Proc IEEE Int Conference on Image Processing. Genoa,(2005)
7. <http://www.poynton.com/PDFs/coloureq.pdf>.
8. http://www.iipl.fudan.edu.cn/~zhangjp/Dataset/fd_pe_de_dataset_intro.htm.
9. <http://www.elec.qmul.ac.uk/staffinfo/andrea/MP.html>.