Robust Object Tracking Using Motion Context in Crowded Scenes

FeiMing Xu, Tong Lu*, and Yirui Wu

National Key Lab for Novel Software Technology
Dept. of Computer Science and Technology, Nanjing University, 210046, China
{xfm18901105,wuyirui1989}@163.com, lutong@nju.edu.cn

Abstract. Tracking objects in a crowded scene with occlusions has been a challenge in computer vision and multimedia in the past years. This paper presents a novel framework to track any arbitrary object through modeling its coupled motion context. For a scene which is densely packed, an individual movement is restricted into a specific pattern to make it regular to be detected. Moreover, members in a crowded scene are clustered into groups and thereby modeled as crowded scene contexts of a tracking object. Accordingly, we present a novel framework to track motions for any object in a crowded scene even with occlusions by using the modeled contexts. Experiments on a number of real-life surveillance videos illustrate the effectiveness and robustness of our method especially in handling occlusions in crowded scenes.

Keywords: object tracking, crowded scenes, occlusion, scene context.

1 Introduction

Object tracking in crowded scenes plays a crucial role for a wide range of computer vision and multimedia applications such as daily surveillance, behavior modeling and abnormal detection. Although many object tracking algorithms have been investigated and a significant progress has been made in the past few years, reliable object tracking in a crowded scene still remains a challenge due to the complexity of noises, varying viewpoints, clutter backgrounds and even illumination changes. Most seriously, a tracking object is always easily to lose when occlusions exist, which are very common and sometimes unavoidable in real-life crowded scenes.

In this paper, we propose a novel algorithm to track any arbitrary object through modeling its coupled motion context in a crowded scene. Essentially, the aim of the proposed method is to continually track an object even when it becomes unseen, which is in general brought by occlusions in a crowded scene. To achieve this, we hypothesize that when a scene is densely packed, individual movements will be restricted into a coupled pattern and thereby makes themselves relatively regular to be described. Moreover, members in a crowded scene

* Corresponding author.

B. Huet et al. (Eds.): PCM 2013, LNCS 8294, pp. 550–560, 2013.
© Springer International Publishing Switzerland 2013

are clustered into groups, which essentially constituting scene contexts for a tracking object. Accordingly, we present a novel object tracking framework to predict motions for continually tracking in a crowded scene with occlusions.

The main contributions of this paper are as follows:

1. We exploit a hybrid representation by simultaneously modeling the motions of an expected object and its coupled environmental contexts. Comparing with the existing object-centric representations, our method is more efficient to characterize a crowded scene, and
2. A novel object tracking algorithm is accordingly presented based on the representation, where specific occlusions can be detected and further well handled for a continuous and relatively long-time tracking. In this way, objects can be robustly tracked especially in crowded scenes.

The rest of the paper is organized as follows. Section 2 introduces the related works. Section 3 discusses the details of our object tracking framework by modeling motion context in crowded scenes. Experimental results and discussions are illustrated in Section 4. Finally, Section 5 concludes the paper and gives our future work.

2 Related Work

Object tracking has always been one of the interests in computer vision and multimedia in the past decade [4][10]. Many efforts have been paid to solve the challenges and the existing methods can be roughly classified into three categories: appearance modeling, motion modeling, and scene context modeling.

The appearance modeling approach avoids the problems of tracking drifts and occlusions by improving appearance templates. For instance, adaptive appearance modeling techniques track objects by indirectly employing statical analysis [7][9][17]. However, their appearance models are susceptible to be contaminated by long-time occlusions due to their blind update strategies. In [12], a tracking object is divided into regular grid cells and occlusion states are accordingly determined for every cell using a classifier. However, the classifiers have to be trained manually.

Considering the complexity of a crowded scene, the motion modeling approach analyzes spatial or temporal motion patterns to assist object tracking. Ali *et al.* [1] track objects in extremely crowded scenes and propose floor fields to predict target motions. [16] models various motion modalities at different locations inside a scene by employing the Correlated Topic Model [13]. Similarly, [11] captures the spatial and temporal variations in crowded scenes by training a hidden Markov model to model motion pattern. Unfortunately, occlusions remain a difficulty during tracking objects.

The context modeling approach makes use of additional scene information to aid object tracking. [6] employs networking targets in the same scene to estimate the location of an tracking object especially when it is unseen. But the relationship between the target and its supporters still faces difficulties in handling complex situations. [18] mines auxiliary objects from scene background and

collaboratively uses them to track objects as a strong verification of expected locations. Unfortunately, it is uncertain to verify the relationship between an auxiliary object and the tracking object in the method.

3 The Proposed Tracking Framework

We propose a novel Bayesian tracking framework to realize the mentioned steps inspired by [8], where the tracking problem is formulated by maximizing the posterior distribution of state x_t at time t with the following available measurement:

$$p(x_t|z_{1:t}) \propto p(z_t|x_t) \int p(x_t|x_{t-1})p(x_{t-1}|z_{1:t-1})dx_{t-1} \tag{1}$$

z_t is the frame at time t and $p(z_t|x_t)$ is its likelihood. The state x_t for frame z_t is modeled as a vector describing the location, height and width of a tracking object.

Essentially, our framework consists of the following three modules.

3.1 Modeling Motion Transitions

We represent the motion transition from frame $t-1$ to frame t by $f(x_t|x_{t-1})$, which can be modeled by a Gaussian distribution of $N(\mu, \sigma)$, where μ is the mean of optical flows to describe object displacements and σ is its covariance matrix. Based on the observation that pedestrians in a crowded scene tend to move in a coupled manner, we hypothesize that an individual pedestrian can not only be tracked according to his/her own motion characterizations, but also the accompanied motion constraints from his/her neighbors to improve the robustness. It can be described as follows

$$p(M|z_t) = w_{indi} \cdot p(M_{indi}|z_t) + w_{group} \cdot p(M_{group}|z_t) \tag{2}$$

We model the latter item as an *object group* to characterize the coupled motions between a specific tracking object and its neighbors, facilitating predicting the trajectory when occlusions are detected. The neighbors here are simultaneously referred as *supporters* for the tracking object.

3.2 Appearance and Motion Updating during Tracking

To well measure the likelihood distribution $p(z_t|x_t)$, we associate an appearance template together with motion characteristics in our method, which are helpful to distinguish a specific object from its visually similar backgrounds or motion similar couplers.

Specifically, we first filter the extracted optical flows which are too large or too small and hash them into the polar coordinates to obtain a group of motion histograms. Since our observation likelihood is essentially decided by an appearance term and a motion term, it can be computed as follows:

$$p(z_t|x_t) = \alpha_{app,t} Simi(x_t, x_{obs}) + \alpha_{mo,t} Simi(x_t, x_{obs}) \tag{3}$$

where $Simi(x_t, x_{obs})$ is a Bhattachayya distance measure to evaluate the similarity of two vectors. The coefficients of the appearance term and the motion term of $Simi(x_t, x_{obs})$ are decided by their abilities to distinguish an object from its surroundings, respectively. We use the score of the particle template calculated in a previous frame to iteratively update the appearance and motion coefficients of $\alpha_{app,t}$ and $\alpha_{mo,t}$ by

$$\alpha_{app,t} = \alpha_{app,t-1} \cdot \frac{var_{app}^2}{var_{app}^2 + var_{mo}^2} \tag{4}$$

$$\alpha_{mo,t} = \alpha_{mo,t-1} \cdot \frac{var_{mo}^2}{var_{app}^2 + var_{mo}^2} \tag{5}$$

The detection of an occlusion or a drift during object tracking is as follows. We store the motion and appearance templates for a tracking object in the beginning k frames of $T = T_1, .., T_k$ as an initialization for tracking. Then for a new frame $t(t > k)$, we compare the tracking result with the stored templates in T as follows: if the matching score is greater than an empirical threshold, we update the stored templates with the latest tracking result from frame t, else an occlusion or a drift is accordingly detected and we thereby stop updating the appearance and motion templates to avoid unexpected results.

3.3 Tracking Objects through a Supporter Model

After modeling motion transitions and giving motion updating strategies, a specific object in a crowded scene with occlusions can be continually tracked as follows. We first track an object using the particle filter method [14] until there is an occlusion or a drift is detected. Then we use the supporters of the tracking object, which are essentially the motion contexts inside the same crowded scene, to assist predicting the trajectory and further verifying whether the object can be tracked continually. These two stages are alternately switched during the whole tracking process in a dynamic way.

Obviously, it can be seen that the key to track an object through its contexts is to detect and model the relationship between a tracking object and its supporters. To search for proper supporters as scene contexts for a specific tracking object, we consider the following two properties are important:

1. A supporter should move regularly to make itself consistent in predicting another tracking object.
2. A supporter should be easy to match, thereby an object together with its supporters will be tracked in a coupled and robust way by propagating the motion details from the supporters to the tracking object as a prediction basis.

A tracking object together with its supporters forms the mentioned object group, which is generally characterized by similar motion patterns and composed of spatially close scene objects. We denote such a group by a state vector which

describes its motion and spatial properties as (L, G, M, t), where L is the 2-D location of the group, G denotes the spatial range of the group which is represented as a variance Σ of the distance from its boundary to the center, m denotes its motion characterization, l is the supporter number, t is its existing time. Then the details to search for supporters of a tracking object are shown by the following steps.

Segment a Crowded Scene. We first segment a crowded scene based on the motions characterized by optical flows. Due to the fact that the extracted feature points from a scene image may be too sparse, we employ the Large Displacement Optical Flow (LDOF) [3] to obtain dense optical flows. We segment every crowded scene frame according to the extracted optical flow vectors. Specifically, instead of clustering the vectors using K-means or kNN algorithms which do not consider spatial constraints, we employ the graph-based method [5] to obtain scene segmentation results (see the example results in Fig. 1). We use $G = (V, E)$, which is an undirected graph, to describe the segmentation results. V is the set of segmented regions where every vertex $v \epsilon V$ denotes a set of optical flow vectors, and $(v_i, v_j) \epsilon E$ corresponds to a pair of neighboring vertices. Note that an edge $(v_i, v_j) \epsilon E$ has a weight of $w(i, j)$, which is a non-negative measure of the dissimilarity between the two neighbors of v_i and v_j. The weight function is measured by $w(i, j) = |Op(p_i) - Op(p_j)|$, where $Op(p_i)$ is the optical flow of pixel p_i.

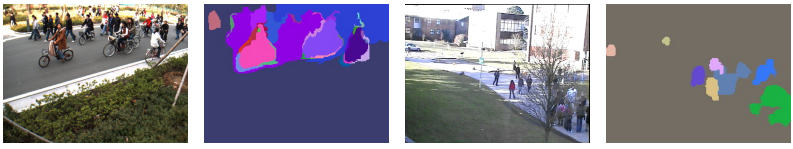


Fig. 1. Example results of graph-based segmentation for relatively crowded scenes

Search for Supporters from the Segmented Regions. After segmenting a crowded scene into several regions, we search for the supporters for a tracking object from the segmented regions which have regular motion descriptions. We employ the method in [14][15] to extract the attention regions (ARs) from the segmentation results, which we consider can be a good choice as motion supporters since the defined ARs are relatively easy to match and less error-prone. Moreover, we can narrow the searching space for more efficient computing since the potential max range of possible ARs in a new frame will be accordingly predicated.

Specifically, we consider a detected AR as a supporter instead of directly using a segmented scene region. Suppose the motion of a detected AR can be predicted by optical flows as a Gaussian distribution (μ, σ), we set the searching space for a new frame as $(-3\sigma, 3\sigma)$ and the local discrimination score is accordingly defined as $\rho_L(x) = min(D(x), D(y))$. Then in a new frame, we use the gradient

decent method to search for the most discriminative regions. An expected AR supporter is defined as a state vector of $Sm_i = \{T(i), S(i)\}$, where $T(i)$ denotes the templates describing the color histogram and the motion pattern of an AR, while $S(i)$ is its state vector characterizing the width, height, 2-D position and its duration time. Moreover, we categorize every supporter into two types according to whether it together with the tracking object exist in the same group, due to the fact that the objects inside the same group tend to have relatively closer motion correlations.

Track an Object Using Supporters in a Crowded Scene. In a crowded scene, we assume a tracking object and its detected supporters have regular relative movements, and therefore motions of a pedestrian are generally easy to be interfered by their surroundings. Specifically, we describe the relationship between a tracking object and its detected supporters by $y = f(x) + \epsilon$, where $f(x)$ is the mapping from the supporters to the object, ϵ denotes Gaussian noise with zero mean and variance σ^n, y and x are the states of the tracking object and its supporters. We model the states of x by the width, height, 2-D location of a supporter within frame t and its duration time t_s, respectively. Accordingly, the mapping from the supporters to the tracking object can be defined as $f(x) = w^T \cdot x$.

Then we need learn the function $f(x)$ which transforms the states from the supporters to the tracking object as $y = f(x) + \epsilon$. This is essentially a Gaussian linear regression problem. Assume for frame t, we obtain a matched supporter of x^* and calculate the function f^* by

$$p(f^*|x^*, X, y) = \int f(x^*|w)p(w|X, y)dw = N(\frac{1}{\sigma^2}(x^*)^\top A^{-1}Xy, (x^*)^\top A^{-1}x^*) \tag{6}$$

where $A = \Sigma_p^{-1} + \frac{1}{\sigma^2}XX^\top$, and w is the Gaussian prior over the parameter $w \sim (0, \Sigma_p)$. The X and y are the matching results of the supporters and the tracking object in $(t - t_s, ..., t - 1)$. Next, after predicting f through Gaussian linear regression, we combine the two types of supporters S to reduce the uncertainly of the position for the tracking object as follows:

$$P(y|S) = \lambda \sum_{S_i \in G^t_{indi}} w_i p(f_i^*) + \lambda_{context} \sum_{S_j \in G^t_{context}} w_j p(f_j^*) \tag{7}$$

Accordingly, the prediction result during object tracking is predicted by the coupled motions of the tracking object when occlusions or drifts are detected.

4 Experiments and Discussion

To demonstrate both the effectiveness and the robustness of our method, we test our algorithm on several challenging two real-world crowded scenes of a campus street and campus gate. The pedestrains within the scene walking together and

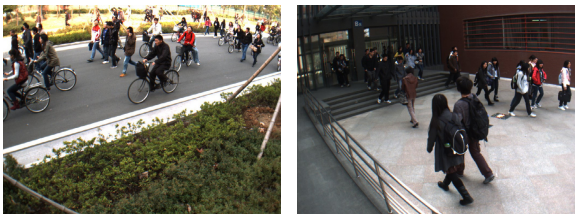


Fig. 2. We use two real-world crowded scenes containing a large amount of individuals to evaluate our method. The Campus Street(left) contains pedestrains and bicycles resulting in severe occlusions. The Campus Gate scene(right) has a simpler background and more regular motion, but it's easy to drift since the individuals walking closely.

Fig. 3. Comparing dense optical flows (left) with sparse optical flows (right) to characterize motions. We note that the detected feature points of sparse optical flows are sometimes even unable to see. Obviously, dense optical flows help analyze motion characteristics inside a crowded scene.

cover each other that result in severe occlusions(see Fig. 2). For every scene image, we zoom it to calculate LDOF features, which are found better than the sparse optical flows as a motion descriptor (see Fig. 3).

We first test our algorithm in the Campus Street scene, where there are many vehicles and pedestrians and thereby is challenging since a tracking object is easy to be partially or even completely occluded by other pedestrians and vehicles. We set the cuboid size of a supporter as 20×20, the initial coefficient of a motion template as 0.3 and the coefficient of appearance template as 0.7. Fig. 4 shows some detected supporters which help predict the estimated position of a tracking object (see the green bounding box in Fig. 4). A samller green bounding box in Fig. 4 indicates a single detected supporter, and the opposite end of the line connecting with the supporter is the calculated prediction indicating the possible position of the tracking object. Once the tracking object is occluded ,its position still can be predicted by its supporters(the red point means the supporter's prediction of the target's position when the occlusion or drift occurs). Accordingly, after the object appears again, we can keep tracking it again(see Fig. 4).

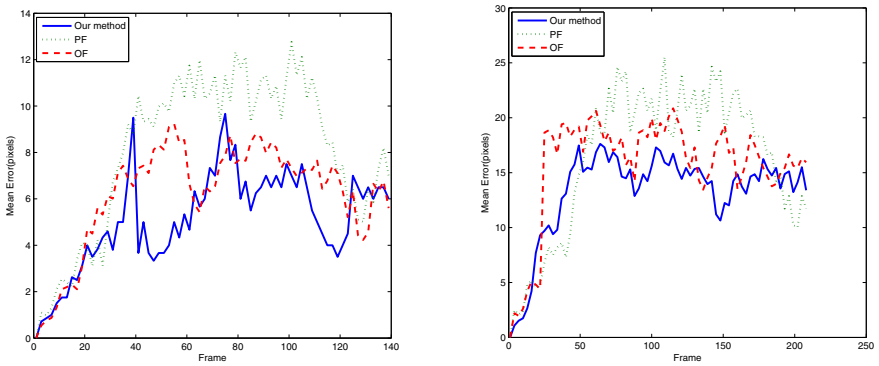
Fig. 5 shows more object tracking examples in a crowded scene. Every object is successfully tracked before occlusion occurs. Once an occlusion occurs, the tracking result will deviate from the ground truth. As illustrated, the OF

Fig. 4. Object tracking examples. The largest bounding box indicate the tracking target and the small green bounding box indicates a detected supporter, while the end of a connecting green line indicates the prediction of the tracking object from a supporter, and our algorithm can track a specific object even when it temporarily disappears.

algorithm [2] successfully handle weak occlusions (see Frame 2 in the first row of Fig. 4), but fail to deal with serious occlusions (see Frame 4 in the first row). The tracking object thereby shifts to the vehicle which is passing by the tracking object. Comparatively, our method successfully tracks the object (see the second row of Fig. 5 as a comparison). More object tracking examples can refer Fig. 5.

To quantitatively compare our algorithm with the existing object tracking methods of PF [15] and OF [2], we measure the results against the ground truth as follows. We first manually label 20 objects in the Campus Street scene and 18 objects in the Campus Gate scene, respectively. Then we define the error measure as the average distance $\|y_t - x_t\|$, where y_t is the ground truth and x_t is a tracked result. Fig. 6 shows the distribution of the errors, which are the means and the standard errors computed by trajectories analysis. The finial mean error of our method is 5.3403 in the Campus Street scene, which is lower than that of PF 8.0440 and OF 6.2129. In the Campus Gate, the mean error of our method is 13.244, which is lower than PF 16.2854 and OF 15.6778. We find the mean error is near to the quarter of object width, indicating that the tracking error is smaller than the size of the tracking object. It can be found that motion supporters help object tracking even under specific heavy occlusions and drifts. Error occurs when the pedestrains unexpectedly changed the motion and our method could not capture the multiple motion since the occluding individual changing the motion in different direction.

Fig. 5. Example object tracking results when occlusions occur. The green bounding box is the ground truth, the purple bounding boxes are the detected results of the OF method, while the red bounding boxes are the tracking results of our method. The red point indicates the centroid of the tracking object when occlusions or drifts are detected. Row 1 and Row 2, Row 3 and Row 4, Row 5 and Row 6 illustrate object tracking results using the OF method and our method for comparison.



(a) (b)

Fig. 6. The mean distance between the manually and automatically tracked trajectories of Street scene (a) and Gate scene (b)

5 Conclusion

This paper proposes a novel algorithm to track individual objects in a crowded scene with occlusions. We restrict an individual movement into a specific pattern to make it regular to be detected. Moreover, members in a crowded scene are clustered into groups and thereby modeled as crowded scene contexts of a tracking object. Accordingly, we present a novel framework to track motions for any object in a crowded scene even with occlusions by using the modeled contexts. The experimental results show that our method can guarantee a relatively long-term tracking even with specific heavy occlusions. We believe crowded motions can be further leveraged to help track irregular objects in our future work.

Acknowledgments. The work described in this paper was supported by the Natural Science Foundation of China under Grant No. 61272218 and 61021062, the 973 Program of China under Grant No. 2010CB327903, and the Program for New Century Excellent Talents under NCET-11-0232.

References

1. Ali, S., Shah, M.: Floor fields for tracking in high density crowd scenes. In: Forsyth, D., Torr, P., Zisserman, A. (eds.) ECCV 2008, Part II. LNCS, vol. 5303, pp. 1–14. Springer, Heidelberg (2008)
2. Baker, S., Matthews, I.: Lucas-kanade 20 years on: A unifying framework. International Journal of Computer Vision 56(3), 221–255 (2004)
3. Brox, T., Bregler, C., Malik, J.: Large displacement optical flow. In: IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2009, pp. 41–48. IEEE (2009)

4. Comaniciu, D., Meer, P.: Mean shift: A robust approach toward feature space analysis. IEEE Transactions on Pattern Analysis and Machine Intelligence 24(5), 603–619 (2002)
5. Felzenszwalb, P.F., Huttenlocher, D.P.: Efficient graph-based image segmentation. International Journal of Computer Vision 59(2), 167–181 (2004)
6. Grabner, H., Matas, J., Gool, L.V., Cattin, P.: Tracking the invisible: Learning where the object might be. In: 2010 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 1285–1292. IEEE (2010)
7. Han, B., Davis, L.: On-line density-based appearance modeling for object tracking. In: Tenth IEEE International Conference on Computer Vision, ICCV 2005, vol. 2, pp. 1492–1499. IEEE (2005)
8. Isard, M., Blake, A.: Condensation conditional density propagation for visual tracking. International Journal of Computer Vision 29(1), 5–28 (1998)
9. Jepson, A.D., Fleet, D.J., El-Maraghi, T.R.: Robust online appearance models for visual tracking. In: Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, CVPR 2001, vol. 1, pp. I–415. IEEE (2001)
10. Khan, Z., Balch, T., Dellaert, F.: An mcmc-based particle filter for tracking multiple interacting targets. In: Pajdla, T., Matas, J(G.) (eds.) ECCV 2004. LNCS, vol. 3024, pp. 279–290. Springer, Heidelberg (2004)
11. Kratz, L., Nishino, K.: Tracking with local spatio-temporal motion patterns in extremely crowded scenes. In: 2010 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 693–700. IEEE (2010)
12. Kwak, S., Nam, W., Han, B., Han, J.H.: Learning occlusion with likelihoods for visual tracking. In: 2011 IEEE International Conference on Computer Vision (ICCV), pp. 1551–1558. IEEE (2011)
13. Lafferty, J.D., Blei, M.D.: Correlated topic models. In: Proceedings of the 2005 Conference Advances in Neural Information Processing Systems, pp. 147–155. Citeseer (2006)
14. Nummiaro, K., Koller-Meier, E., Van Gool, L.: An adaptive color-based particle filter. Image and Vision Computing 21(1), 99–110 (2003)
15. Pérez, P., Hue, C., Vermaak, J., Gangnet, M.: Color-based probabilistic tracking. In: Heyden, A., Sparr, G., Nielsen, M., Johansen, P. (eds.) ECCV 2002, Part I. LNCS, vol. 2350, pp. 661–675. Springer, Heidelberg (2002)
16. Rodriguez, M., Ali, S., Kanade, T.: Tracking in unstructured crowded scenes. In: 2009 IEEE 12th International Conference on Computer Vision, pp. 1389–1396. IEEE (2009)
17. Ross, D., Lim, J., Yang, M.-H.: Adaptive probabilistic visual tracking with incremental subspace update. In: Pajdla, T., Matas, J(G.) (eds.) ECCV 2004. LNCS, vol. 3022, pp. 470–482. Springer, Heidelberg (2004)
18. Yang, M., Wu, Y., Hua, G.: Context-aware visual tracking. IEEE Transactions on Pattern Analysis and Machine Intelligence 31(7), 1195–1209 (2009)