Object Tracking with Dynamic Template Update and Occlusion Detection

Longin Jan Latecki, Roland Miezianko
CIS Department, Temple University, Philadelphia, PA 19122, USA
{latecki, rmiezian}@temple.edu

Abstract

The objective of this paper is to track moving objects using dynamic template initializations and updates, and to identify tracking events in videos such as occlusions and merging of motion regions. The proposed tracking method is based on spatiotemporal texture motion regions, image alignment, and minimum cost estimation based template selection. The dynamic template update is based on the detection of events in videos. The proposed method has been experimentally evaluated on color and thermal infrared videos.

1. Introduction

Until today one of the main tracking methods is Lucas-Kanade tracker introduced in [1]. It computes an optimal alignment of an object template from a previous frame to the actual frame. Image alignment works well when the template from the previous frame is very similar to the object in the actual frame, which is rarely the case if the template is not updated, due to size, appearance, and light changes. Therefore, a robust method to update the template is the key step to successful tracking by template alignment. We introduce such a method in this section. We use the minimal cost matching [2] to associate the aligned template with a motion region in the actual frame. Then the associated motion region becomes our new template. We call our method selective hypothesis tracking, since by combining [1] and [2], with spatiotemporal motion detection [6], we can predict and verify certain tracking events like occlusion, split or merge of tracked motion regions. The minimal cost matching [2] alone does not work well when objects change direction and speed abruptly or in the presence of occlusion or merge. We show below that the proposed selective hypothesis tracking not only can track robustly but also recognizes such tracking events.

An overview of the existing approaches to motion detection and tracking can be found in the collection of papers edited by Remagnino et al. [7] and in the special section on video surveillance in IEEE PAMI edited by Collins et al. [8]. Wren et al. [9] were the first who used a statistical model of the background instead of a reference image. One of the most popular approaches for motion detection was introduced by Stauffer and Grimson (S&G) [10].

2. Selective hypothesis tracking

Let us consider a simple example of tracking one moving object in a video scene to illustrate our approach. A moving object appears in new frame t_0. Motion is detected and a bounding box is established for the new motion region. New template is created immediately as this is the first detected motion region and no other template yet exists.

New motion region is detected in frame t_1. Motion bounding box is established based on the spatiotemporal blocks and tracking begins as one template already exists. Image alignment and minimum cost computation establish the best template to motion region association. A motion vector is computed between the associated template and motion region. Template assumes the location and size of its associated motion region.

Two motion regions are detected in frame t_2 and bounding boxes are established for each region. Image alignment and minimum cost computation between motion regions and the template are performed where single best association is selected. The motion vector is updated the associated template and motion region. Again, the template assumes the location and size of its associated motion region, and new template is created for the motion region without any template association. Thus, using motion detection we detect a new moving object and initialize its template for tracking. The instantaneous template update and initialization is the key improvement that makes tracking using image alignment robust if the appearance of the tracked object changes.

2.1. L-K Image alignment overview

One of the most widely used image alignment techniques is the Lucas-Kanade (L-K) algorithm [1]. It has become not only the standard for image alignment but also a standard for optical flow measurement [5]. The basis to image alignment is the gradient descent computation, which is the de facto standard method. The image alignment is just one of three major components of the selective hypothesis tracking algorithm. A brief overview of the Lucas-Kanade algorithm follows.

Let a template T_R^i be a template extracted from one of the previous frames *F*, where *i* is a template index. The output of L-K is displacement vector (*u,v*) such that $T_R^i + (u,v)$ is the best alignment to the current frame *G*.

When the template is occluded or when regions merge into a single region, the L-K algorithm is destined to fail. Therefore, in such cases, we simply estimate the position using the motion vector T_s^i as follows:

$$T_s^i = S \cdot T_s^i + (1-S) \cdot (M_C^j - T_C^i) \quad (1)$$

where M_C^j is the motion region's centroid, and T_C^i is the tracked object's centroid, and *S* is a velocity decay value (in our experiments *S*=0.9) [2]. It is important to stress that we can robustly detect template occlusion and merge in our framework as shown in Section 2.4.

2.2. Selective decay of templates

Each unassociated template, a template without a mapping to any motion region, has its time-to-live factor decremented based on the presence of any new motion region within its vicinity. Each active template is first aligned to the current frame *G* only if its time to live value T_t^i is set to the maximum threshold value TTL_{MAX}, where $TTL_{MAX} = 3 \cdot fps$. This time to live threshold allows us to track the same object through occlusions. The frame per second (fps) value depends on the video stream. Our test included frames per second values between 2 and 30.

Initially, the time to live factor is set to TTL_{MAX}, and it is reset to TTL_{MAX} when template is associated with a motion region. The vicinity threshold between a template and any motion region is defined as

$$\left| M_C^j - T_C^i \right| < P_{MAX} \quad (2)$$

where $P_{MAX} = 10 \cdot BLOCKSIZE$ is the maximum distance threshold between a motion region and a template T_C^i in pixels (it is based on the motion block size used, i.e. 4). The value of P_{MAX} depends on the type of object we wish to track. The template's time-to-live is decremented by $\delta = 1$ when (2) is true and no alignment was established for the template to any motion regions. Otherwise it is decremented by a factor of $\delta = 3$.

$$T_t^i = T_t^i - \delta \quad (3)$$

Thus, there is a limit on the time given that the template may disappear from the field of view (or stop moving). If the template is not associated to any motion region, and there is no motion region within a vicinity of the template, the template's time to live value is decrement three times faster than the template that may disappear intermittently. This intermittent disappearance may be associated with thin object occlusions.

2.3. Template dynamic position update

Initial work on minimum cost estimates and template position update was done by Shah et al. [2]. Minimum cost estimation provides a single value allowing to compare multiple templates to a single motion region. Our approach is similar to using predicted position and size in the minimum cost estimation. We have included two additional factors and one additional weight in our minimum cost estimation. The first additional factor is the difference in direction between estimated direction of predicted motion vector T_s^i, and the direction from template centroid T_C^i to motion region centroid M_C^j. The direction cost has an associated weight factor w_d. The second additional factor in the minimum cost estimation is the cost of persistence. It is computed based on the template's time to live value T_t^i, where templates are rewarded with zero cost if their $T_t^i = TTL_{MAX}$, otherwise the cost increases linearly to one.

Each template's centroid and rectangle are updated using the motion vector (4), however, its value is bounded by the motion region rectangle. If any of the sides of the predicted template rectangle are outside the motion region rectangle to which it is associated, the predicted template rectangle is clipped to the location of motion bounding box.

$$T_C^i = T_C^i + T_s^i \text{ and } T_R^i = T_R^i + T_s^i \quad (4)$$

where template's centroid T_C^i and rectangle T_R^i are bounded by motion rectangle M_R^j.

The cost based on distance is estimated as the absolute difference between template and motion region centroids

$$\Delta p = \left| M_C^j - T_C^i \right| \quad (5)$$

the cost of the size difference is estimated as

$$\Delta s = \left| M_R^j - T_R^i \right| / (M_R^j + T_R^i) \quad (6)$$

the cost of the direction difference is estimated as

$$\Delta d = \left| \arctan(T_s^i) - \arctan(M_C^j - T_C^i) \right| \quad (7)$$

where the angle direction is within range of 0 to 2π; and the cost based on persistence is estimated as

$$\Delta t = (TTL_{MAX} - T_t^i) / TTL_{MAX} \quad (8)$$

where TTL_{MAX} is the maximum time to live threshold value.

The total cost C between each motion region and each template is then computed as

$$C = w_p \Delta p + w_d \Delta d + w_s \Delta s + \Delta t \quad (9)$$

where w_p is the weight factor for position and distance offset, w_d is weight factor for direction difference, and w_s is the weight factor for size difference, and

$$w = w_p + w_d + w_s = 1 \quad (10)$$

In our experiments we used $w_p = 0.4$, $w_d = 0.5$, and $w_s = 0.1$. The persistence factor Δt in the minimum cost computation is normalized by the time to live threshold TTL_{MAX}. It is zero for each template with its time to live value T_t^i equal to TTL_{MAX} and it increases to the maximum value of 1 when T_t^i approaches zero.

Each new frame containing motion regions is evaluated against known templates. New template is created from a motion region only if a motion region has no associated template based on image alignment, predicted position, or minimum cost computation.

Templates that are not within any motion region have their time to live decreased. Once this value reaches zero, the template will no longer be used during the association with motion regions. This may provide false results when an object disappears and reappears beyond the time to live threshold value. A higher level shape recognition method may be used to solve this problem.

2.4. Tracking events detection

Shah et al. [2] distinguish the following main tracking cases for a realistic scenario of stationary camera based motion detection and tracking:

Case 1 *Inter-object occlusion* comprises of two or more individual templates T converging on single motion region M in the camera field of view. Tracking is handled by the pre-occlusion predicted motion T_s^i of each individual template bound by the single motion region rectangle M_R^j. The L-K algorithm has most difficulty with this case and predicted position and motion region is necessary to maintain tracking. This scenario is shown in Fig. 1. Clearly, our approach provides a solution to this case only if the actual motion of the occluded object does not differ too much from the predicted motion. Our main contribution here is the fact that it is possible to detect this even in our framework. Our tracking algorithm detects the occlusion event when two templates are aligned to one motion region. At that time we switch from image alignment (L-K algorithm) to predicted position in order to track both templates within one motion region.

Case 2 *Occlusion of objects by a large stationary structure* occurs when template T^i is completely hidden and no motion regions are detected within the vicinity of the last know template position. Tracking is handled by the pre-occlusion motion velocity T_s^i and the dynamic decay of time to live parameter T_t^i. This scenario is illustrated in [11].

Case 3 *Occlusion of objects due to thin structures* consists of template being partially covered, or split by a thin foreground object, such a tree. Both image alignment vector T_v^i and predicted motion velocity T_s^i are used to verify objects association with a motion region rectangle M_R^j. Only one of the motion regions of the object will be associated with a template, other motion regions will become new templates.

Case 4 *Objects exiting from the scene* may be classified as a permanent occlusion. The position vector T_s^i is the indication as to the template's exit from the field of view (FOV). It is checked against the boundary of the FOV. Additionally, the absence of any motion blocks increases the decay of template's time to live parameter T_t^i, which in turn marks the template as no longer used in future tracking. Objects that re-enter the scene are considered new objects if their T_t^i is zero. In addition, due to the nature of the proposed technique, we distinguish two additional cases:

Case 5 *Single region splits into multiple regions*. One template T^i splits into two or more individual motion regions. This case is handled by the L-K vector T_v^i and minimum cost computation allowing for

continuous use of template label. Image alignment must be augmented with minimum cost estimation to correctly predict the label assignment. Predicted position computation alone does not reassign the tracking label correctly (e.g. when direction change during the split process). Once two motion regions are detected the single template is aligned to both motion regions. The best alignment is augmented by the minimum cost estimation to select the best motion region that matches the single template. The other unmatched motion region becomes a new template.

Case 6 *Regions have merged in a single region* is handled by the individual predicted velocity T_s^i of each template within the boundaries of the motion region M_R^j. Multiple templates merge into one motion region. Both image alignment and minimum cost prediction are necessary for the reassignment of know labels to new individual templates. This scenario is shown in Fig. 1. Selective hypothesis tracking algorithm handles Case 6 similarly to Case 1. Another possible solution would be to create a new template from the single motion region that two or more templates occupy.

3. Results

Two methods are used to evaluate selective hypothesis tracking results: the visual inspection of identified tracking objects and comparison of tracking centroids to independent ground truth data.

Figures illustrating the precision of the proposed initialization of new templates on *Campus1* video are shown in [11]. By identifying tracking events, we are able to detect appearance of new moving objects.

Some challenging tracking events involve several objects merging and splitting where two or more objects cross paths as observed by the camera and some objects become hidden during the merging. The tracking algorithm must predict the possible location of each individual object despite the fact that the motion detection only provides a single motion rectangle.

In the merging of regions scenario, the image registration technique will not work as only one of the merged objects is in the foreground. The known velocity of each object before the merge occurred is used to update the predicted position of each object. The predicted position is then bound by the observed motion rectangle, limiting the objects position to within the motion rectangle.

For an object that disappears and reappears much later than is allowed by the algorithm (template's time to live value) a new template is created instead of matching to templates already seen by the system. The splitting of single template into multiple objects is a difficult case of label assignment. The single label of the object before the split must now be assigned to one of the motion regions after the split.

Single region splitting into two objects is shown in Fig. 2. Group of people (object labeled 4) is approaching parked cars in frame 1043. In frame 1099, a person left one of the parked cars while object 4 passed next to it. In this situation there is a single motion blob corresponding to tracking object 4. While this motion region is expanding due to the fact that a single person is walking in the opposite direction to the object 4, there is still single object being tracked. The split occurs around frame 1141 where the single person is assigned new tracking label 6 while object 4 continues along its course maintaining it own label, as seen in frame 1214.

An example of regions that have merged into a single region is shown in Fig. 1. In frame 863, object 3 (van) and object 4 (group of people) approach each other. In frame 898 the merge occurs with object 3 is in the foreground. Image registration of object 4 is impossible as it is partially visible. The predicted position based on last know motion velocity along with bounding motion rectangle provide a sufficient location of individual objects (frame 923). In frame 963 both objects are no longer occluded and continue in their respective directions while maintaining the correct labels.

Example of occlusion of objects due to large structures is shown in [11]. *Infra3* is a thermal infrared video sequence showing single person walking behind two trees. In frame 233 object 1 approaches the first tree and becomes invisible to the camera in frame 252. It reappears in frame 263 as the tracking algorithm keeps the same object label and does not create a new template. Object 1 is tracked continuously until frame 431 when it disappears again. Later it reappears again as object 1 in frame 461 before permanently leaving the field of view.

3.1. Tracking ground truth data evaluation

Independent ground truth data was used to verify the selective hypothesis tracking algorithm. *Split1* video from [3] along with its ground truth data was used to evaluate the tracking results. Once each tracking object is identified, their centroids were compared to the ground truth data. *Split1* figure in [11] displays the projection of all ground truth data onto a single frame along with all tracking centroids. On average, the tracking centroid distance from ground truth data was 5.2 pixels with standard deviation of 2.6

pixels for *Split1* video, while using the motion block size of 4x4 pixels.

4. Conclusion

The proposed selective hypothesis tracking algorithm combines motion regions, image alignment, and minimum cost estimation to handle dynamic template updates and event recognition. The spatiotemporal motion regions provide the bounding regions for the search and alignment space of previously seen templates. Image alignment of templates to motion regions is selected when no occlusion is detected. The minimum cost computation is selected when full or partial occlusion is detected, and it is based on different factors than existing methods, such as direction and persistence factors. The key factor is the detection of events, such as occlusions and template divergence into single motion region, to allow switching from image alignment to predictive position estimation, and then switching back to image alignment and minimum cost estimation once a split event occurred.

5. Acknowledgments

R. Miezianko has been partially supported by Terravic Research Grant (RGX-200512).

6. References

[1] B. Lucas, and T. Kanade, "An iterative image registration technique with an application to stereo vision". In Proceedings of the International Joint *Conference on Artificial Intelligence*, pages 674–679, 1981.

[2] O. Javed, and M. Shah, "Tracking and Object Classification for Automated Surveillance", *The seventh European Conference on Computer Vision*, Copenhagen, May 2002.

[3] EC Funded CAVIAR project IST 2001 37540, found at URL: http://homepages.inf.ed.ac.uk/rbf/CAVIAR/.

[4] PETS video repository, video sequences Campus 1: ftp://pets.rdg.ac.uk/PETS2002//DATASET1/TESTING/***/.

[5] S. Baker, and I. Matthews, "Lucas-Kanade 20 Years On: A Unifying Framework" *Int. J. Computer Vision* 56, 3, 221-255, 2004.

[6] L. J. Latecki, R. Miezianko, and D. Pokrajac. "Motion Detection Based on Local Variation of Spatiotemporal Texture". *CVPR Workshop on OTCBVS*, Washington DC, July 2004.

[7] Remagnino, P., G. A. Jones, N. Paragios, and C. S. Regazzoni, eds., *Video-Based Surveillance Systems*, Kluwer Academic Publishers, 2002.

[8] R.T. Collins, A.J. Lipton, and T. Kanade, "Introduction to the Special Section on Video Surveillance", *IEEE PAMI* 22(8) (2000), pp. 745–746.

[9] C. Wren, A. Azarbayejani, T. Darrell, and A.P. Pentland, "Pfinder: Real-time Tracking of the Human Body", *IEEE PAMI* 19(7) (1997), pp. 780–785.

[10] C. Stauffer, W. E. L. Grimson, "Learning patterns of activity using real-time tracking", *IEEE PAMI* 22(8) (2000), pp. 747–757.

[11] R. Miezianko, Temple University Video & Vision Lab evaluation videos, ground truth data, and test figures: http://knight.cis.temple.edu/~video/Tracking/

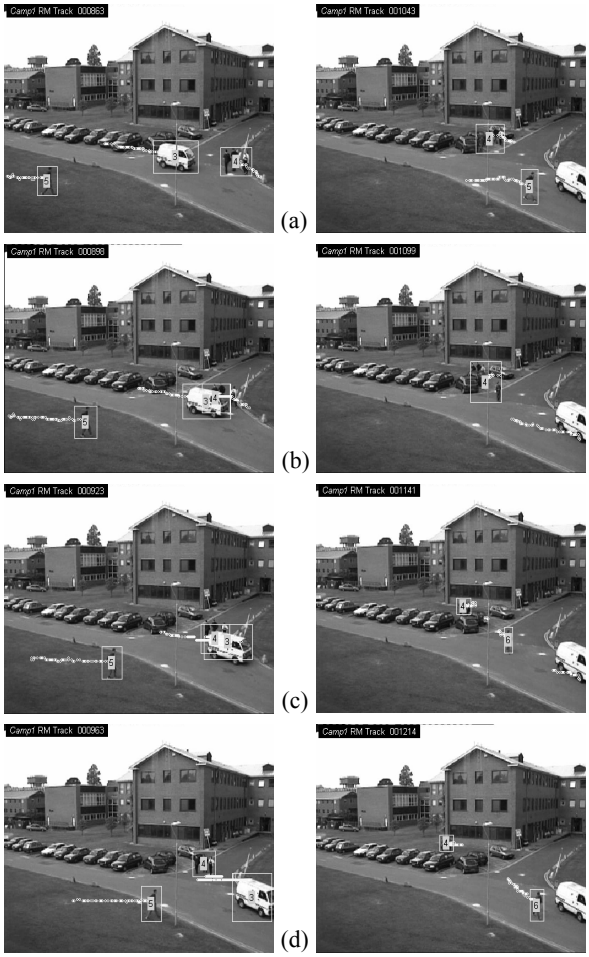

Fig. 1. Object occlusion.

Fig. 2. Object splitting.