Real-time Multi-Object Tracking with Occlusion and Stationary Objects Handling for Conveying Systems

Adel Benamara, Serge Miguet, Mihaela Scuturici

Real-time Multi-Object Tracking with Occlusion and Stationary Objects Handling for Conveying Systems

Adel Benamara[1,2], Serge Miguet[1,2], and Mihaela Scuturici[1,2]

[1] Université de Lyon, CNRS
[2] Université Lyon 2, LIRIS, UMR5205, F-69676, France

Abstract. Multiple object tracking has a broad range of applications ranging from video surveillance to robotics. In this work, we extend the application field to automated conveying systems. Inspired by tracking methods applied to video surveillance, we follow an on-line tracking-by-detection approach based on background subtraction. The logistics applications turn out to be a challenging scenario for existing methods. This challenge is twofold: First, conveyed objects tend to have a similar appearance, which makes the occlusion handling difficult. Second, they are often stationary, which make them hard to detect with background subtraction techniques. This work aims to improve the occlusion handling by using the order of the conveyed objects. Besides, to handle stationary objects, we propose a feedback loop from tracking to detection. Finally, we provide an evaluation of the proposed method on a real-world video.

1 Introduction

Multiple object tracking is a fundamental problem in computer vision; It plays a significant role in many applications, such as video surveillance, robot navigation, human-machine interaction, self-driving cars. Many studies in the recent years focus on tracking pedestrian, cars, bicycles. In this work, we investigate a novel application of multiple object tracking, which is automated conveyor systems. Automated conveyor systems play an important role in many industries, including postal services, packaging, automobile, aeronautics. The purpose of the conveying system is to transport material from one location to another. The controller of the conveying system needs to know the position of the conveyed objects to route them to the right location. Therefore, a robust real-time tracking system is required. Several tracking technologies have been used in the logistics industry, such as photo-electrical sensor network and RFID.

In this work, we follow an on-line tracking-by-detection approach that uses background subtraction techniques for object detection. We address the following problems:

Complex occlusions management: Some methods [9] [5] [6] introduce the notion of group in order to track collectively occluded and occluding objects during occlusions. The model of each member of the group is used at the end

of occlusion to achieve correct re-association. In these approaches object re-identification relies on appearance model (color and shape). However, in the logistics scenario, conveyed objects tend to have similar appearance both in color and size (e.g. parcels in the postal application). As a result, these methods fail to recover objects identity. Furthermore, conveyed objects are close to each other, which results in complex group aggregation as depicted in Fig 1.a. We propose to overcome this issue, an association scheme based on the ordering relation defined by conveyor pathways. In other words, we attempt to exploit the fact that conveyed objects travel serially along the conveyor routes.

Stationary object detection: This is a known issue of background subtraction methods [3] [4]. The absorption of moving objects that become motionless is caused by the adaptation process of background subtraction techniques, where pixels of the stationary objects are progressively included to the background model as shown in Fig 1.b. In particular, in the logistics scenario, this issue has a dramatic effect on the tracking performance since conveyed objects may be stopped for certain periods of time for a routing purpose. In this paper, we propose a feedback loop from tracking to the detection module; the idea is to stop the adaptation process for regions that correspond to stationary objects. An idea was first presented in [8], we extend the original idea to several state-of-the-art background subtraction techniques [1] [12], we show significant improvements of tracking results in real video sequence.

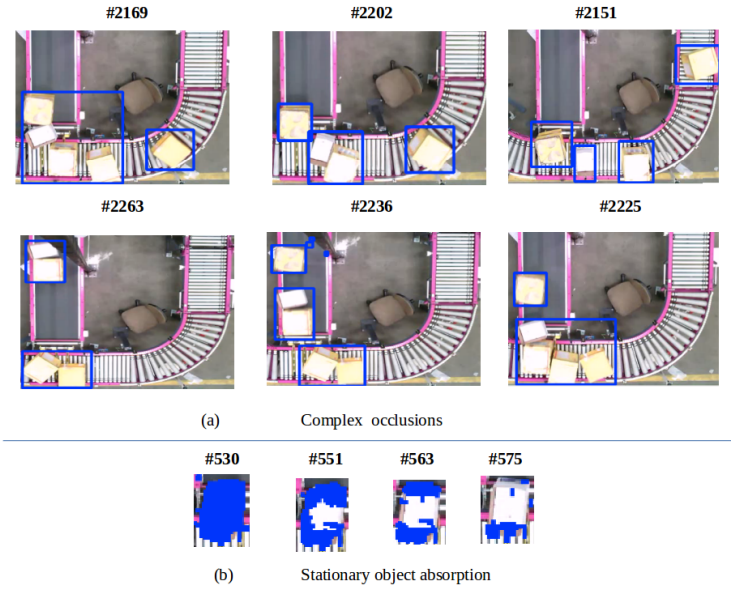


Fig. 1. Examples of the addressed issues.

2 Overview of the method

Fig.2 presents the overview of the pipeline. The proposed algorithm is a tracking-by-detection approach. Therefore, the pipeline is decomposed into two main modules: detection and tracking. The acquisition module grabs frames from the camera sensor and feeds sequentially the tracking pipeline with input frames. The first block of the detection module is background subtraction. This block extracts foreground mask by comparing the input frame to the background model. The morphology block removes noise and fills the holes present in the foreground mask. Erosion is first performed to remove noisy pixels, followed by dilation to fill the holes. Connected pixels are labeled using connected component analysis [11], the output of this block are called blobs (a blob is a set of 8-connected pixels). The blobs are filtered, blobs with small size are discarded. Features are extracted from the set of selected blobs, each blob is represented by its center coordinates, bounding box, and color histogram. The tracking module is decomposed into two blocks. The data association block detects associations between the tracked objects and detected blobs. Complex associations are first detected based on the overlapping criteria between objects and blobs. Blobs and objects who are not involved in a complex association are then matched using the Hungarian algorithm based on their centroid. Depending on the type of detected associations between blobs and objects, the tracking management block creates, suppress, associates, recovers objects after splits or merges objects into groups. The model of associated objects is also updated.

The paper is structured as follows: Section 3 presents the tracking algorithm that deals with occlusions. Section 4 explains the tracking to detection feedback mechanism used to stop static objects integration to the background. Section 5 presents extensive results on a real video sequence.

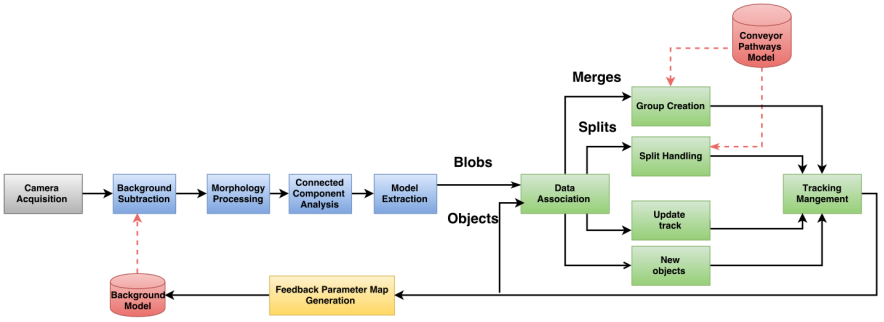


Fig. 2. Tracking-by-detection pipeline with tracking feedback to detection.

3 Tracking

3.1 Data association

The aim of tracking is to maintain objects identity by resolving at each step the association between tracked and detected objects. In the ideal case, where individual detections are provided for each object in the scene, a greedy one-to-one association scheme would be sufficient to link detections reliably. However, in real situations, individual detection cannot be achieved for partial or total occlusions. Especially when a background subtraction technique is used, where a single pixel may connect two distinct blobs into a merged blob. Therefore, we perform data association in two steps. The first step detects complex situations that may result from merging and splitting blobs or in the presence of occlusions between objects. The second step considers only simple cases where individual objects are visible and can be matched with at most one blob.

3.2 Complexe association

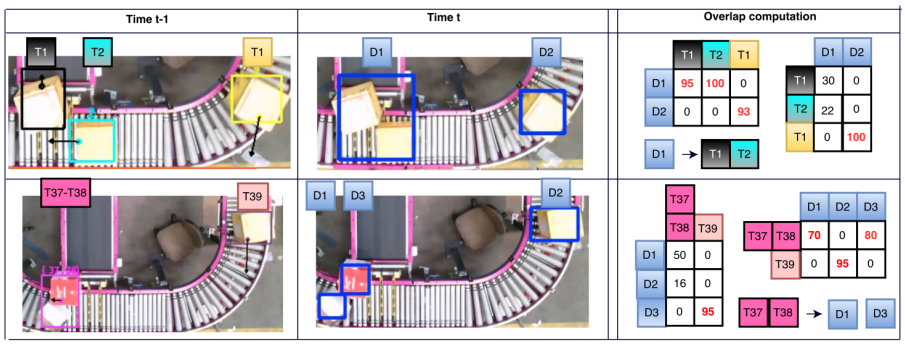


Fig. 3. Split and merge detection

Complex association is based on the following observations :

- A blob b_j with a large bounding box that overlaps simultaneously with smaller bounding box of several objects $M = \{o_1, o_2, ..., o_n\}$, with $n > 1$ corresponding to a **merged blob**.
- An object o_i with a large bounding box that overlaps simultaneously with smaller bounding box of several blobs $S = \{b_1, b_2, ..., b_m\}$, with $m > 1$ corresponding to a **splitting object**.

The above observations are valid when the detection step is fast enough to observe a slow evolution of the objects bounding box.

3.3 Simple association

For objects (denoted $\mathcal{O}$) and blobs (denoted $\mathcal{B}$) that are not involved in a complex assignment, detected blobs should be assigned to separate and visible objects based on spatial proximity. Hence, we define the cost ϕ_{ij} of assigning blob j to object i using the Euclidean distance as

$$\phi_{ij} = \begin{cases} \left\| b_j^c - o_i^c \right\| & \text{if } \left\| b_j^c - o_i^c \right\| < \tau_{max}, \\ \infty & \text{otherwise} \end{cases} \tag{1}$$

where b_j^c and o_i^c are respectively the centroid of blob j and object i, τ_{max} is a distance threshold used to define the maximum allowed displacement of objects between two successive frames. In order to obtain the optimal assignment, we use Hungarian algorithm [7] for computing the assignment matrix $\mathbf{A}^* = [a_{ij}]$,$a_{ij} \in \{0, 1\}$, which minimizes the total assignment cost:

$$\begin{aligned} \mathbf{A}^* = \underset{\mathbf{A}}{\operatorname{argmin}} & \quad \textstyle\sum_{i=1}^{|\mathcal{O}|} \sum_{j=1}^{|\mathcal{B}|} a_{ij}\phi_{ij} \ , \\ \text{s.t.} & \quad \sum_{i=1}^{|\mathcal{O}|} a_{ij} = 1 \ , \ \forall j \in \{1, \ldots, |\mathcal{B}|\} \ , \\ & \quad \sum_{j=1}^{|\mathcal{B}|} a_{ij} = 1 \ , \ \forall i \in \{1, \ldots, |\mathcal{O}|\} \ , \end{aligned} \tag{2}$$

Although the Hungarian algorithm is designed for square matrix ($|\mathcal{O}| = |\mathcal{B}|$), the algorithm can easily be extended for rectangular cost matrix by padding with impossible cost ∞.

3.4 Occlusion handling with ordering

We follow a merge-split approach to deal with occlusions. In this approach, as soon as the beginning of an occlusion is detected, individual objects are frozen, and a group containing these objects is created. The group is tracked as any other objects (i.e. with its model and attributes). When a split is detected, the problem is to identify the object that is splitting from the group. Several methods rely on appearance model to re-establish identities at each split. In the context of logistics, objects of interest have a similar appearance, making object re-identification hard. On the other hand, objects are conveyed in procession. Their trajectory follows the conveyor pathways. The ordering of the objects is then preserved during their routing. We propose to use this order to re-identify objects at the end of occlusion. The ordering relation is derived from the conveyor model. A simple polyline that fit the pathways of the conveyor is sufficient to order conveyed objects. This model can be either defined manually during a configuration step or by clustering objects trajectories in a training phase. Fig 4-b illustrates the conveyor model used during the experiments.

In our method when a merge occurs, we maintain the object ordering inside the group using the conveyor model. When the group split into several blobs, we

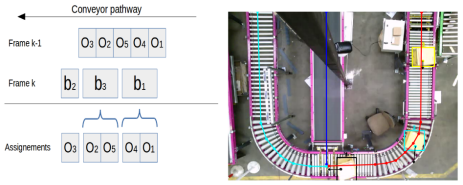
use Algorithm-1 to re-establish objects identity. The algorithm finds the group partition that maximizes the likelihood to the blobs involved in the split such as illustrated in Fig 4-a. Blobs are first sorted using the ordering relation. The algorithm processes the ordered objects and blobs sequentially. Let be i the index of the current object and j the index of the current blob. For each blob b_j, the algorithm tries to determine if the blob corresponds to a single object or several grouped objects using the area ratio as a clue. The algorithm constructs a new group $\mathcal{G}_j$ initialized with a single object o_i, if the area ratio match with the blob b_j, the algorithm moves to the next object and blob. Otherwise, the next objects are added to the candidate group until a match is found. Aside from the sort performed on the blobs, the complexity of the algorithm is $O(max(N, M))$, where N is the number of objects member of the group and M is the number of blobs detected in the split.

Procedure 1 Association based on order

Input: A group of ordered N tracked objects $\mathcal{G} = \{o_1, o_2, \ldots, o_N\}$ that split into M blobs $\mathcal{B} = \{b_1, b_2, \ldots, b_M\}$.

Output: Assignments pairs $< \mathcal{G}_|, b_j >$ where $\{\mathcal{G}_1, \ldots, \mathcal{G}_M\}$ is M partition of $\mathcal{G}$.

procedure RESOLVE–SPLIT
  $Sort(\mathcal{B})$ ▷ sort the blobs using the ordering relation.
  $j = 1$
  $i = 1$
  while $i \leq |\mathcal{G}|$ and $j \leq |\mathcal{B}|$ **do**
    $\mathcal{G}_j = \{o_i\}$
    while $Match(\mathcal{G}_j, b_j) \neq true$ and $i + 1 \leq |\mathcal{G}|$ **do**
      $\mathcal{G}_j = \mathcal{G}_j \cup \{o_{i+1}\}$
      $i = i + 1$
    end while
    $j = j + 1$
    $i = i + 1$
  end while
end procedure
function $Match(\mathcal{G}_j, b_j)$
  $r_{\mathcal{G}_j} = area(\mathcal{G}_j)/area(\mathcal{G})$ ▷ compute the contribution of $\mathcal{G}_j$ to group $\mathcal{G}$.
  $r_{b_j} = area(b_j)/area(\mathcal{B})$ ▷ compute the contribution of b_j to blobs set $\mathcal{B}$.
  if $r_{\mathcal{G}_j} \approx r_{b_j}$ **then**
    return true
  else
    return false
  end if
end function



(a) Assignment that preserve objects ordering.

(b) Object ordering using the conveyor pathways model.

Fig. 4. Occlusion handling with ordering

Methods	Update strategy	Parameters map generation
ViBe [1]	Conservative	$(\mathrm{R},\sharp,\phi) = \begin{cases} (R^{static}, \sharp^{static}, \phi^{static}), & \text{if } (x,y) \in M_{static} \\ (R^{bg}, \sharp^{bg}, \phi^{bg}), & \text{otherwise} \end{cases}$
GMM [12] [10]	Blind	$\alpha(x,y) = \begin{cases} 0, & \text{if } (x,y) \in M_{static} \\ \alpha^{bg}, & \text{otherwise} \end{cases}$

Table 1. Background subtraction methods and parameters map generation. The (x, y) subscript is left for ViBe in the first row, $M_{stationary}$ is a binary mask generated based on the union of the bounding boxes of objects detected as *Stationary*.

4 Feedback Framework

4.1 Feedback of the parameter map

In this section, we will describe the proposed feedback mechanism. We will first discuss background subtraction technique regarding the strategy they follow to update their background model. The update strategy can follow two schemes blind and conservative [1]. Conservative methods update their background model only with pixels classified as background. These methods can indefinitely detect stationary objects. However, pure conservative update scheme leads to everlasting foreground pixel in the case of misclassification. Conversely, blind methods update the background model with pixels whether they have been classified as background or foreground. These methods are not subject to the deadlock situation. However, blind methods are sensitive to slow moving objects.

We have modified the original methods in order to be able to selectively control the update parameters at pixel level rather than a global level. As shown in Table.1, the Gaussian mixture model (GMM) incorporates pixel samples with the learning rate α. We zero out α for stationary objects pixels. ViBe is a conservative method uses a spatial update scheme to incorporate background information. This mechanism can cause the absorption of stationary objects. Therefore, we disable the neighborhood update process for stationary objects pixels. We also adapt the parameter of the method for stationary objects pixels. $\sharp_{min}$ is the minimum number of samples of the background model close to the current pixel (the distance in the color space is under R), ϕ controls the frequency of the update process.

Stationary objects are detected using the speed estimation derived by the Kalman filter associated to each tracked object. An object is considered as stationary when its speed is less than a fixed threshold v_{sta} for c_{sta} successive frames.

5 Evaluation

In this section, we will discuss the evaluation protocol. We describe first the collected sequence and the methodology used to generate the ground truth dataset. We use a smart embedded camera to perform the video acquisition and the performance evaluation. The embedded platform includes a quad processor ARM Cortex-A9 running at 1GHz with 1 GB of memory. The video was taken with an Omnivision OV5640 at a frame rate of 25 frames per second at VGA resolution (640x480). The video is 4 minutes and 9 seconds long (approximately 6231 frames). The annotated sequence contains 5419 frames. We use a dedicated tool for this task. Objects are manually annotated only on some keyframes by defining object's bounding box, the annotation between key frame is obtained by linear interpolation. The camera is mounted above a conveyor with a bird view angle. The camera capture only a portion of the conveyor system. We use 5 parcels in the experiments: 4 yellow boxes with similar color and size and one white rectangular parcel. The objects are circulating on the conveyor system in a loop. 22 objects are annotated in the whole sequence, including a human operator that manipulates a blocked parcel.

We use the CLEARMOT [2] metrics, particularly : the *Multiple Object Tracking Accuracy*(MOTA), *Multiple Object Tracking Precision* (MOTP), *False Positive* (FP), Missed Object (Miss), *ID Switch* (IDs). To measure the improvement of each contribution, we run the tracking pipeline with different settings. We execute the pipeline with the proposed tracking-to-detection feedback enabled and disabled for Vibe and GMM. In conjunction with the two possibles occlusion handling methods : ours with order and appearance as a baseline. For occlusion handling with appearance, we use a 2D histogram and quantize the H,V components of HSV color space using 3 and 2 bits respectively.

Quantitative results of our experiments are listed in Table 2. As expected GMM is more sensitive to stationary objects, the feedback loop demonstrates a significant improvement to lower missed objects (i.e. stationary objects). ViBe, on the other hand, is less sensitive to stationary objects due to the conservative update scheme. However, the inhibition of the spatial update mechanism with the feedback loop has also a significant effect in lowering missed objects. We achieve significant improvements with the proposed occlusion handling with ordering relation, we lower both id switches and missed objects in comparison to the appearance method. Indeed, the lowering of id switches is explained by the presence of identical yellow boxes, which makes the appearance method fail to recover their identity reliably. On the other hand, the lowering of missed objects is explained by the ability of our method to handle complex occlusion situations, where a group splits into several smaller groups. In this situation, the appear-

Occlusion Handling	Methods	MOTA [%]	MOTP [px]	FP	MISS	IDS
Appearance	ViBe	80.4237	65.09	515	2223	34
	ViBe + Feedback	83.5805	60.98	566	1731	28
	GMM	61.9845	81.18	607	4737	39
	GMM + Feedback	84.7811	64.86	772	1357	26
Order	ViBe	86.815	67.13	742	1098	27
	ViBe + Feedback	**88.6017**	**56.14**	1577	**23**	**14**
	GMM	66.363	81.43	**111**	4043	37
	GMM + Feedback	82.3517	70.31	577	1897	25

Table 2. Tracking performance. MOTA (higher is better), MOTP (lower is better), FP false positive (lower is better), Miss (lower is better), IDS id switch (lower is better).

ance method can only handle blobs that correspond to individual objects, the blobs that correspond to more than one objects will not be matched and then generate misses.

6 Conclusion

We have proposed a multi-object tracker adapted for conveying systems. We have proposed an occlusion handling method that exploits the ordering of the object to reliably re-establish objects identity in complex occlusion situations. We have also addressed stationary objects detection with by introducing a feedback loop from tracking to detection. Our evaluations on real data demonstrate significant improvements of tracking results in logistics scenario compared to state-of-the-art approach for occlusion handling. In future works, we will extract the ordering relation by learning conveyed objects trajectories rather than relying on manual configuration.

References

1. Olivier Barnich and Marc Van Droogenbroeck. Vibe: A universal background subtraction algorithm for video sequences. *IEEE Transactions on Image processing*, 20(6):1709–1724, 2011.
2. Keni Bernardin and Rainer Stiefelhagen. Evaluating multiple object tracking performance: the clear mot metrics. *EURASIP Journal on Image and Video Processing*, 2008(1):1–10, 2008.
3. Thierry Bouwmans. Traditional and recent approaches in background modeling for foreground detection: An overview. *Computer Science Review*, 11:31–66, 2014.
4. Carlos Cuevas, Raquel Martínez, and Narciso García. Detection of stationary foreground objects: A survey. *Computer Vision and Image Understanding*, pages –, 2016.
5. Rosario Di Lascio, Pasquale Foggia, Gennaro Percannella, Alessia Saggese, and Mario Vento. A real time algorithm for people tracking using contextual reasoning. *Computer Vision and Image Understanding*, 117(8):892–908, 2013.

6. Aziz Dziri, Marc Duranton, and Roland Chapuis. Real-time multiple objects tracking on raspberry-pi-based smart embedded camera. *Journal of Electronic Imaging*, 25(4):041005–041005, 2016.
7. James Munkres. Algorithms for the assignment and transportation problems. *Journal of the society for industrial and applied mathematics*, 5(1):32–38, 1957.
8. Aristodemos Pnevmatikakis and Lazaros Polymenakos. Kalman tracking with target feedback on adaptive background learning. In *International Workshop on Machine Learning for Multimodal Interaction*, pages 114–122. Springer, 2006.
9. Matthieu Rogez, Lionel Robinault, and Laure Tougne. A 3d tracker for ground-moving objects. In *International Symposium on Visual Computing*, pages 695–705. Springer, 2014.
10. Chris Stauffer and W Eric L Grimson. Adaptive background mixture models for real-time tracking. In *Computer Vision and Pattern Recognition, 1999. IEEE Computer Society Conference on.*, volume 2. IEEE, 1999.
11. Satoshi Suzuki et al. Topological structural analysis of digitized binary images by border following. *Computer Vision, Graphics, and Image Processing*, 30(1):32–46, 1985.
12. Zoran Zivkovic. Improved adaptive gaussian mixture model for background subtraction. In *Pattern Recognition, 2004. ICPR 2004. Proceedings of the 17th International Conference on*, volume 2, pages 28–31. IEEE, 2004.