

Quantitative evaluation of different aspects of motion trackers under various challenges

Fei Yin, Dimitrios Makris, Sergio A Velastin, James Orwell
Digital Imaging Research Centre (DIRC),
Faculty of Computing, Information Systems and Mathematics,
Kingston University, Surrey, KT1 2EE, UK

<{fei.yin,d.makris,sergio.velastin,j.orwell}@kingston.ac.uk>

Abstract

When a motion tracking system underperforms, researchers and practitioners face the challenge of modifying and improving it. This paper proposes a framework of a rich set of metrics for quantitative evaluation that is able to indicate which modules of such a complicated system needs adjustment. We also use six different video sequences that represent a variety of challenges to assess the system performance under different conditions. We illustrate the practical value of the proposed metrics by evaluating and comparing two motion tracking algorithms.

1 Introduction

In the last two decades, researchers and industry have shown a growing interest for motion tracking systems [1] [2] [3] [4]. Performance evaluation has played an important role on developing, assessing and comparing motion tracking algorithms. However, performance evaluation has different meanings and usages to different categories of people. End-users and public bodies are interested in assessing systems for validation and standardisation and therefore are interested in measuring high-level metrics of performance, as specified by end-users. For example, The i-LIDS evaluation programme, developed by the UK Home Office, focuses on measuring the accuracy of detection of high-level events such as “vehicle illegal parking”, “sterile zone intrusion”, “abandoned bag” and “door entering and exiting”[5].

On the other hand, the research community, as expressed by workshops (e.g. PETS), projects (e.g. ETISEO, CAVIAR) and the peer-reviewed publication process, aims to compare algorithms and systems in order to identify state-of-the-art techniques.

Individual researchers and practitioners, when they work to develop and improve their systems, they are very interested in identifying which modules of the tracking system fail. Our work aims to address this issue and proposes a framework that estimates the potential reasons of failure.

Ellis [6] investigated the main requirements for effective performance analysis for surveillance systems and proposed some methods for characterising video datasets. Nascimento and Marques (CAVIAR) [7] proposed a framework which compares the output of different motion detection algorithms against given ground truth and estimates objective metrics such as false alarms, detection failures, merges and splits. Lazarevic-McManus et al [8] evaluated performance of motion detection based on ROC-like curves and the F-measure. The latter allows straight-forward comparison using a single value that takes into account the application domain.

While the above work mainly deals with evaluation of motion detection, other researchers attempt to deal with the evaluation of both motion detection and object tracking. Needham and Boyle [9] proposed a set of metrics and statistics for comparing trajectories to account for detection lag, or constant spatial shift. However, they take only trajectories (sequences of points over time) as the input of evaluation and therefore their approach may not give sufficient information on the precision of the object size estimation and spatial extent over time. Bashir and Porikli [11] gave definitions of evaluation metrics based on the spatial overlap of ground truth and system bounding boxes that are not biased towards large objects. However they are counted in terms of frame samples. Such an approach is justified when the objective of performance evaluation is object detection. In object tracking, counting TP, FP and FN tracks is more natural choice that is consistent to the expectations of the end-users. Brown et al [10] suggests a framework for matching ground truth tracks and system tracks and computing performance metrics. However their definition, based on the comparison of the system track centroid and an enlarged ground truth bounding box which favours tracks of large objects. Although such an approach is useful for evaluating the system's performance, it does not provide a clue about the source of potential system failures. Nghiem et al (ETISEO) [14] proposed a large set a metrics that can address each object detection and tracking problem separately, and could be used for comparison between algorithms. However, they do not provide a rigorous mathematical definition for each of those metrics.

In this work, we propose a rich set of metrics to assess different modules of tracking systems (such as motion segmentation, motion tracking and data association) and identify specific failures of motion tracking. We illustrate the approach through a variety of sequences which represent a wide variety of challenges for tracking systems.

2 Evaluation Metrics

2.1 Preparation

We define tracking as the problem of estimating the spatial extent of the non-background objects for each frame of a video sequence. The result of tracking is a set of tracks for all non-background objects. Before the evaluation metrics are introduced, we define the concepts of spatial and temporal overlap between tracks, which are required to quantify the level of matching between Ground Truth (G) tracks and System (S) tracks, both in space and time. We adopt the idea of spatial overlap proposed in [7] which is the bounding box overlapping $A(G_i, S_j)$ between G_i and S_j tracks in a specific frame k .

$$A(G_{ik}, S_{jk}) = \text{Area}(G_{ik} \cap S_{jk}) / \text{Area}(G_{ik} \cup S_{jk}) \quad (1)$$

We also define the binary variable $O(G_i, S_j)$, based on a threshold T_{ov} which in our work is set to 20% according to experiments. As we can see in figure 1 (we run a frame based evaluation, measuring the overlapping between detections and ground truth, for different values of T_{ov} over the range [0,1] using PETS2001 dataset1, camera1), the number of correct detections does not change significantly when the threshold is set between 10% and 40%, therefore we choose a threshold somewhere in the middle.

$$O(G_{ik}, S_{jk}) = \begin{cases} 1 & \text{if } A(G_{ik}, S_{jk}) \geq T_{ov} \\ 0 & \text{if } A(G_{ik}, S_{jk}) < T_{ov} \end{cases} \quad (2)$$

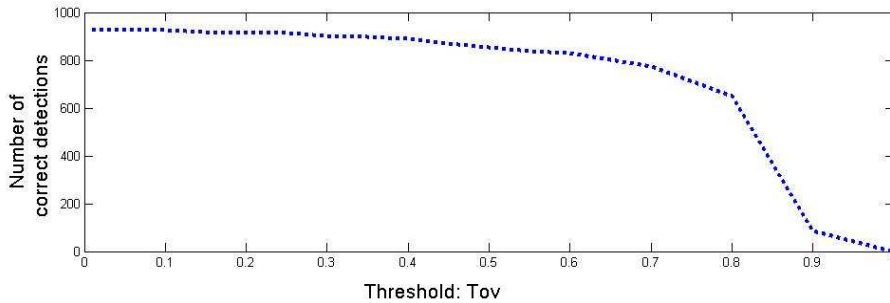


Figure 1 Number of correct detections for different values of T_{ov}

We define temporal overlap $TO(G_i, S_j)$ as a number that indicates overlap of frame span between system track j and GT track i :

$$TO(G_i, S_j) = \begin{cases} TO_E - TO_S & \text{if } TO_E > TO_S \\ 0 & \text{if } TO_E \leq TO_S \end{cases} \quad (3)$$

where TO_S is the maximum of the first frame indexes of the two tracks and TO_E is the minimum of the last frame indexes of the two tracks. We use a temporal-overlap criterion to associate systems tracks to GT tracks according to the following condition in order to find candidates for GT and system track associations:

$$L(G_i \cap S_j) / L(G_i) \geq TR_{ov} \quad (4)$$

where $L(\cdot)$ is the number of frames and TR_{ov} is an appropriate threshold (we tried out a

range of thresholds for TR_{ov} , and we find out that, the choice of different threshold does not significantly affect the evaluation results. Hence, we choose 15% in our experiment which will not miss any possible associations, for both trackers). If Eq.4 is true, then, we compute the metrics for that pair of tracks and start evaluating the performance of the system track. If more than one system track satisfies the condition of Eq.4, the GT track is still considered as one correct detected track, and the multiple associations will be reflected in the track fragmentation metric (see Sec.2.5.1). If there are multiple GT tracks associated with one system track, that will be reflected in the ID change metric (see Sec.2.5.2). Therefore, multiple track associations are addressed by the track fragmentation and ID change metrics and they do not affect the correct detection, detection failure and false alarm metrics. Note that we use the same values of T_{ov} and TR_{ov} for the whole evaluation procedure.

2.2 Performance overview

In this section, we introduce high level metrics such as Correct Detected Track (CDT), Track Detection Failure (TDF) and False Alarm Track (FAT) to obtain an overall view of performance of the tracking system. Brown et al [10] also proposed similar metrics to count the number of track true positives, track false negatives and track false positives. But their definition of track spatial overlap was whether the centroid of a system track is inside an enlarged bounding box of a GT track which favours bigger tracks.

2.2.1 Correct detected track (CDT) or True Positive (TP):

A GT track will be considered as been detected correctly, if it satisfies both of the following conditions:

Condition 1: The temporal overlap between GT track i and system track j is larger than a predefined track overlap threshold TR_{ov} which is set to 15%. (Eq.4)

Condition 2: The system track j has sufficient spatial overlap with GT track i . (Eq.5)

$$\sum_{k=1}^N A(G_{ik}, S_{jk}) / N \geq T_{ov} \quad (5)$$

where N is the number of temporal overlapping frames between G_i and S_j . Each GT track is compared to all system tracks according to the conditions above. Even if there is more than one system track meets the conditions for one GT track (which is probably due to fragmentation), we still consider the GT track to have been correctly detected.

2.2.2 False alarm track (FAT) or False Positive (FP):

Although it is easy for human operators to realise what is a false alarm track (event) even in complex situation, it is hard for an automated system to do so. Here, we give a practical definition of false alarm track. We will consider a system track as false alarm, if the system track meets any of the following conditions:

Condition 1: A system track j has temporal overlap smaller than TR_{ov} with any GT track i . (Eq.6)

$$L(G_i \cap S_j) / L(S_j) < TR_{ov} \quad (6)$$

Condition 2: A system track j does not have sufficient spatial overlap with any GT track i although it has enough temporal overlap with GT track. (Eq.7)

$$\sum_{k=1}^N A(G_{ik}, S_{jk}) / N < T_{ov} \quad (7)$$

FAT is an important metric for end users, because they usually require it as low as possible to ensure that operators will not overwhelmed by false alarms and therefore will not tend to ignore the system.

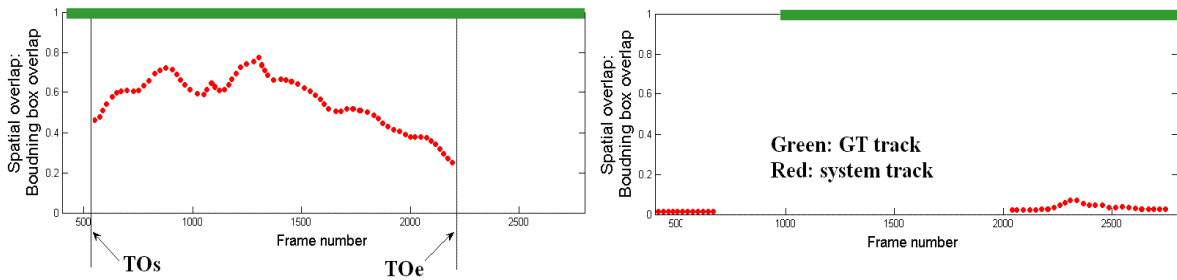


Figure 2 Example of correct detected track(left) and false alarm tracks(right)

2.2.3 Track detection failure (TDF) or False Negative (FN):

A GT track will not be considered detected (i.e. as a track detection failure), if it satisfies any of the following conditions.

Condition 1: A GT track i has temporal overlap smaller than TR_{ov} with any system track j . (Eq.4 is false)

Condition 2: Although a GT track i has enough temporal overlap with system track j , it has insufficient spatial overlap with any system track (Eq.7).

2.3 Motion segmentation evaluation

The following metric is introduced to evaluate the performance of the motion segmentation module, i.e. how accurate the localization of foreground objects is.

2.3.1 Closeness of Track (CT):

We adopted the idea of frame based bounding box area matching [7] and extended it to a track based metric. For a pair of associated GT track and system track, we define the closeness of track as a sequence of spatial overlaps for the period of temporal overlap:

$$a_t = \{A(G_{i1}, S_{j1}), A(G_{i2}, S_{j2}), \dots, A(G_{iN}, S_{jN})\} \quad (8)$$

From Eq.8, we can estimate the average closeness \bar{a}_t for that pair of GT and system tracks. Suppose there are M pairs of tracks in total. We define the closeness for this video sequence as a weighted average of the closeness of all M pairs of tracks:

$$\bar{A} = \frac{\sum_{t=1}^M L(a_t) \cdot \bar{a}_t}{\sum_{t=1}^M L(a_t)} \quad (9)$$

where \bar{a}_t is the average closeness for the t^{th} pair of tracks. The weighted standard deviation of track closeness for the whole video sequence is defined as:

$$\sigma_A = \frac{\sum_{t=1}^M L(a_t) \cdot \sigma_{a_t}}{\sum_{t=1}^M L(a_t)} \quad (10)$$

where σ_{a_t} is the standard deviation for the t^{th} pair of tracks.

$$\sigma_{a_t} = \sqrt{\frac{(A(G_{i1}, S_{j1}) - \bar{a}_t)^2 + (A(G_{i2}, S_{j2}) - \bar{a}_t)^2 + \dots + (A(G_{iN}, S_{jN}) - \bar{a}_t)^2}{(N-1)}} \quad (11)$$

2.4 Motion tracking evaluation

2.4.1 Latency of the system track (LT):

We introduce latency (time delay) of the system track as the time gap between the time that an object starts to be tracked by the system and the first appearance of the object. The optimal latency should be zero. A very large latency means the system may not be sensitive enough to trigger the tracking in time or indicates that the detection is not good enough to trigger the tracking.

Latency is estimated by the difference in frames between the first frame of system track, $F(S_i)$ and the first frame of GT track, $F(G_i)$.

$$l_i = F(S_j) - F(G_i) \quad (12)$$

If there are more than one system track associated with GT track i , we chose the shortest latency for GT track i . Suppose the total number of GT tracks is I , we can calculate the average latency for all the GT tracks in one video sequence as:

$$\bar{l} = \frac{\sum_{i=1}^I l_i}{I} \quad (13)$$

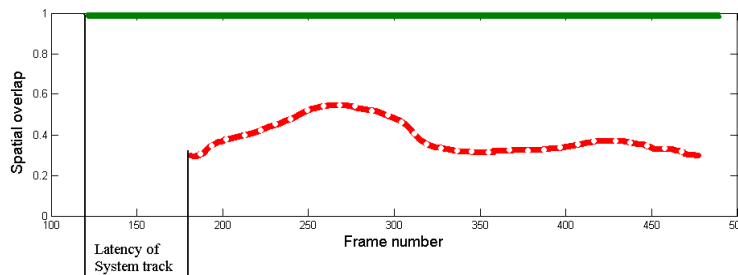


Figure 3 Example of latency

2.4.2 Track Distance Error (TDE):

This metric measures the positional error of system tracks which is proposed in [9]. In figure 4, (x,y) and (p,q) are the trajectory points (centroid of bounding boxes) for a ground truth track and a system track respectively.

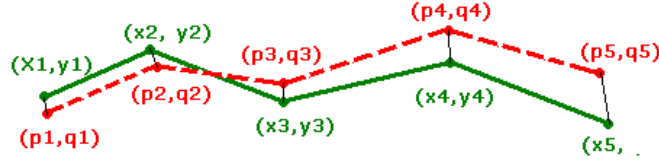


Figure 4 Example of a pair of trajectories: GT track(green), system track(red)
Track distance error \bar{D} for the whole video sequence is defined as the weighted average with the duration of overlapping of each pair of tracks as the weight coefficient.

$$\bar{D} = \left(\sum_{t=1}^M L(d_t) \cdot \bar{d}_t \right) / \sum_{t=1}^M L(d_t) \quad (14)$$

where \bar{d}_t is the average distance error for the t^{th} pair of tracks. The standard deviation of track matching errors for the whole sequence is defined as:

$$\sigma_D = \left(\sum_{t=1}^M L(d_t) \cdot \sigma_{dt} \right) / \sum_{t=1}^M L(d_t) \quad (15)$$

where σ_{dt} is the standard deviation of distance errors for the t^{th} pair of tracks.

2.5 Data association evaluation

Finally, three metrics (track fragmentation, ID change and track completeness) are used to detect data association errors. [14] also defines similar metrics (ID persistence and ID confusion). In this work, we put more constraints for the metric ID change to deal with object intersection or occlusion and hence obtain more reliable evaluation result.

2.5.1 Track Fragmentation (TF):

Fragmentation indicates the lack of continuity of system track for a single GT track. In an optimal condition, track fragmentation error should be zero which means the tracking system is able to produce continuous and stable tracking for the ground truth object. As mentioned before, we allow multiple associations between GT track and system track therefore fragmentation is measured from the track correspondence results.

$$TF = \sum_{i=1}^I TF_i \quad (16)$$

where TF_i is the number of system tracks that associated with GT track i and I is the total number of GT tracks (the condition for association is mentioned in Sec. 2.2.1).

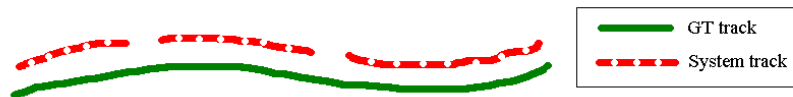


Figure 5 Example of track fragmentations

2.5.2 ID Change (IDC):

We introduce the metric IDC_j to count the number of ID changes for system track j . Note that such a metric provides more elementary information than an ID swap metric.

For each frame k , the bounding box $D_{j,k}$ of the system track S_j may be overlapped with $N_{D_{j,k}}$ GT areas out of total I_k , where $N_{D_{j,k}}$ is given by:

$$N_{D_{j,k}} = \sum_{i=1}^{I_k} O(G_{ik}, D_{jk}) \quad (17)$$

We take into account only the frames that $N_{D_{j,k}}=1$ (which means that the track S_j is associated (spatially overlapped) with only one GT Track for each of these frames). We use these frames to estimate the ID changes of S_j as the number of changes of associated GT tracks. We can estimate the total number of ID changes in a video sequences as:

$$IDC = \sum_{j=1}^J IDC_j \quad (18)$$

where J is the total number of system tracks.

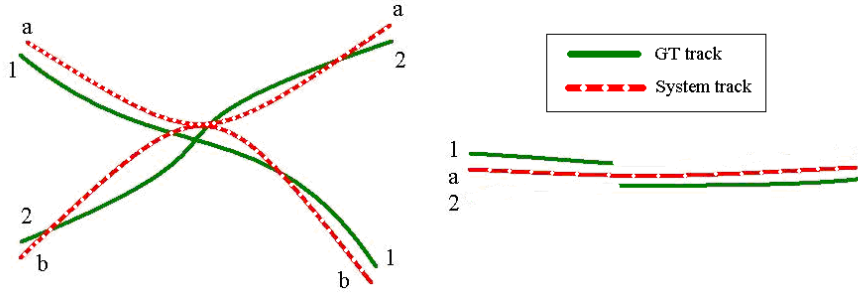


Figure 6 Example of ID changes (left: two IDC right: one IDC)

2.5.3 Track Completeness (TC):

The time span that the system track overlapped with GT track divided by the total time span of GT track. A fully complete track is where this value is 100%. (A vague definition of a similar metric was given in [14]).

$$c_{ij} = \sum_{k=1}^N O(G_{ik}, S_{jk}) / L(G_i) \quad (19)$$

If there is more than one system track associated with the GT track, then we choose the maximum completeness for each GT track. Also, we define the average track completeness of a whole sequence as:

$$\bar{C} = \sum_{i=1}^I \max(c_i) / I \quad (20)$$

where I is the total number of GT tracks and $\max(c_i)$ is the maximum completeness for GT track i and the standard deviation of track completeness for the whole sequence is defined as:

$$\sigma_c = \sqrt{\sum_{i=1}^I (c_i - \bar{C})^2 / (I - 1)} \quad (21)$$

3 Results

We demonstrate the practical value of the proposed metrics by evaluating two motion tracking systems (an experimental industrial tracker and the OpenCV1.0 blobtracker [12]). For openCV tracker, we used adaptive mixture of Gaussian models for background estimation, connected component analysis for data association and Kalman filtering for tracking blob position and size. We run the trackers on six video sequences (shown in Figure 7) that represent a variety of challenges, such as illumination changes, shadows, snow storm, quick moving objects, blurring of FOV, slow moving objects, mirror image of objects and multiple object intersections. We keep the parameters of each tracker constant throughout the whole test which reflects their performance in real world conditions. The results of performance evaluation are presented in Tables 1-6.



Figure 7 PETS2001 PetsD1TeC1.avi, i-LIDS SZTRA103b15.mov, i-LIDS PVTRA301b04.mov
 i-LIDS SZTRA104a02.mov, BARCO Parkingstab.avi, BARCO Snowdivx.avi

PetsD1TeC1.avi	Industrial tracker	OpenCV tracker
Number of GT Tracks	9	9
Number of System Tracks	12	17
Correct Detected Track	9	9
False Alarm Track	3	6
Track Detection Failure	0	0
Track Fragmentation	3	3
ID Change	5	7
Latency of Track	46	66
Average Track Closeness	0.47	0.44
Deviation of Track Closeness	0.24	0.14
Average Distance Error	15.75	5.79
Deviation of Distance Error	23.64	5.27
Average Track Completeness	0.67	0.58
Deviation of Track Completeness	0.24	0.89

Table 1 Results for PETS2001 Sequence

SZTRA103b15.mov	Industrial tracker	OpenCV tracker
Number of GT Tracks	1	1
Number of System Tracks	8	15
Correct Detected Track	1	1
False Alarm Track	3	12
Track Detection Failure	0	1
Track Fragmentation	0	0
ID Change	0	0
Latency of Track	50	9
Average Track Closeness	0.65	0.23
Deviation of Track Closeness	0.21	0.10
Average Distance Error	9.10	15.05
Deviation of Distance Error	12.48	3.04
Average Track Completeness	0.68	0.42
Deviation of Track Completeness	0.00	0.00

Table 2 Results for i-LIDS SZTRA103b15

SZTRA104a02.mov	Industrial tracker	OpenCV tracker
Number of GT Tracks	1	1
Number of System Tracks	4	9
Correct Detected Track	1	1
False Alarm Track	0	5
Track Detection Failure	0	0
Track Fragmentation	0	2
ID Change	0	0
Latency of Track	74	32
Average Track Closeness	0.79	0.34
Deviation of Track Closeness	0.21	0.17
Average Distance Error	7.02	16.69
Deviation of Distance Error	15.67	7.55
Average Track Completeness	0.73	0.44
Deviation of Track Completeness	0.00	0.00

Table 3 Results for i-LIDS SZTRA104a02

PVTRA301b04 .mov	Industrial tracker	OpenCV tracker
Number of GT Tracks	102	102
Number of System Tracks	225	362
Correct Detected Track	90	95
False Alarm Track	67	112
Track Detection Failure	12	7
Track Fragmentation	62	98
ID Change	95	101
Latency of Track	57	78
Average Track Closeness	0.30	0.22
Deviation of Track Closeness	0.21	0.16
Average Distance Error	49.70	24.65
Deviation of Distance Error	60.31	22.85
Average Track Completeness	0.34	0.26
Deviation of Track Completeness	0.57	0.65

Table 4 Results for i-LIDS PVTRA301b04

Parkingstab.avi	Industrial tracker	OpenCV tracker
Number of GT Tracks	4	4
Number of System Tracks	9	17
Correct Detected Track	4	4
False Alarm Track	1	11
Track Detection Failure	0	0
Track Fragmentation	0	0
ID Change	0	0
Latency of Track	72	35
Average Track Closeness	0.50	0.39
Deviation of Track Closeness	0.20	0.14
Average Distance Error	13.32	11.82
Deviation of Distance Error	11.55	8.16
Average Track Completeness	0.82	0.77
Deviation of Track Completeness	0.11	0.96

Table 5 Results for Parkingstab

Snowdivx.avi	Industrial tracker	OpenCV tracker
Number of GT Tracks	3	3
Number of System Tracks	28	29
Correct Detected Track	3	3
False Alarm Track	19	20
Track Detection Failure	0	0
Track Fragmentation	2	5
ID Change	0	0
Latency of Track	590	222
Average Track Closeness	0.14	0.42
Deviation of Track Closeness	0.23	0.12
Average Distance Error	28.50	16.69
Deviation of Distance Error	35.44	11.62
Average Track Completeness	0.33	0.35
Deviation of Track Completeness	0.47	0.71

Table 6 Results for Snowdivx

From the results provided by the tables above, we can note that the overall performance of the industrial tracker is better than the OpenCV tracker. Because, the industrial tracker has higher number of correct detected tracks, and lower number of false alarm tracks and track detection failures. We can also figure out the weakness of trackers against specific challenges. For instance, in video sequences SZTRA104a02 (Table 3) and Parkingstab (Table 5) both with significant illumination variations, very few false alarms were generated by the industrial tracker, but quite a few by the OpenCV tracker. Therefore, the industrial tracker seems more robust against illumination changes than the OpenCV tracker.

Regarding the motion segmentation module, the industrial tracker performs better than the OpenCV one, as reflected by the Track Closeness metric in most of the sequences. However, in snowy conditions (e.g. the Snowdivx sequence, see Table 6), the OpenCV motion segmentation module performs better.

The motion tracking module of the OpenCV tracker seems to overcome the above disadvantage and performs better, as it produces lower Average Distance Error for similar or slightly worse Track Completeness (see Tables 1, 4, 5 and 6). Also, it responds quicker than the industrial tracker because it has smaller track latency.

The data association module of the industrial tracker performs slightly better than the OpenCV tracker, since it has lower number of track fragmentations and ID changes and higher track completeness. For example, in the PETS2001 sequence (Table 1), the intersections of multiple objects, cause a few ID changes for both trackers which indicates that their data association modules need to be improved.

4 Conclusions

We presented a rich set of track based metrics to measure the performance of specific module of motion tracking algorithms. Metrics, such as Correct Detected Track (CDT), False Alarm Track (FAT) and Track Detection Failure (TDF) provide a general overview of the algorithm performance. Closeness of Track (CT) metric indicates the spatial extent of the objects and it is closely related to the motion segmentation module of the tracker. Metrics, such as Track distance Error (TDE) and Latency of Track (LT) indicate the accuracy of estimating the position and how quick the tracker responds respectively and they are related to the motion tracking module of the tracker. Track Fragmentation (TF) show whether the temporal or spatial coherence of tracks is established. ID Change (IDC) and Track Completeness (TC) are useful to test the data association module of multi-target trackers.

We tested two trackers using 6 video sequences that contain more than 30,000 frames which provide a variety of challenges, such as illumination changes, shadows, snow storm, quick moving objects, blurring of FOV, slow moving objects, mirror image of objects and multiple object intersections. The variety of metrics and datasets allows us to reason about the weaknesses of particular modules of the trackers against specific challenges, assuming orthogonality of modules and challenges. We also make comparison of performance of different trackers. This approach is a realistic way to understand the drawbacks of motion trackers, which is important for improving them.

5 Acknowledgements

The authors would like to acknowledge financial support from BARCO View, Belgium and the Engineering and Physical Sciences Research Council (EPSRC) REASON project under grant number EP/C533410.

References

- [1] I. Haritaoglu, D. Harwood, and L. S. Davis, "W⁴: Real-time surveillance of people and their activities," *IEEE Trans. Pattern Anal. Machine Intell.*, pp. 809 – 830, Aug. 2000
- [2] M. Isard and A. Blake, "Contour tracking by stochastic propagation of conditional density," in *Proc. European Conf. Computer Vision*, pp. 343 - 356 , 1996
- [3] M. Xu, T.J. Ellis, "Partial observation vs. blind tracking through occlusion", *British Machine Vision Conference, BMVA*, September, Cardiff, pp. 777 -786, 2002
- [4] Comaniciu D., Ramesh V. and Meer P., "Kernel-Based Object Tracking", *IEEE Trans. Pattern Anal. Machine Intell.*, pp. 564-575, 2003
- [5] Home Office Scientific Development Branch: Imagery library for intelligent detection systems (i-LIDS), <http://scienceandresearch.homeoffice.gov.uk/hosdb/cctv-imaging-technology/video-based-detection-systems/i-lids/>, [Last accessed: August 2007]
- [6] T. Ellis, "Performance Metrics and Methods for Tracking in Surveillance", *Third IEEE International Workshop on Performance Evaluation of Tracking and Surveillance*, June, Copenhagen, Denmark, 2002
- [7] J. Nascimento, J. Marques, "Performance evaluation of object detection algorithms for video surveillance", *IEEE Transactions on Multimedia*, pp. 761-774, 2005
- [8] N.Lazarevic-McManus, J.R.Renno, D. Makris, G.A.Jones, "An Object-based Comparative Methodology for Motion Detection based on the F-Measure", in '*Computer Vision and Image Understanding*', Special Issue on Intelligent Visual Surveillance, pp. 74-85, 2007
- [9] C.J. Needham, R.D. Boyle. "Performance Evaluation Metrics and Statistics for Positional Tracker Evaluation" *International Conference on Computer Vision Systems (ICVS'03)*, Graz, Austria, pp. 278 – 289, April 2003
- [10] L. M. Brown, A. W. Senior, Ying-li Tian, Jonathan Connell, Arun Hampapur, Chiao-Fe Shu, Hans Merkl, Max Lu, "Performance Evaluation of Surveillance Systems Under Varying Conditions", *IEEE Int'l Workshop on Performance Evaluation of Tracking and Surveillance*, Colorado, Jan 2005
- [11] F. Bashir, F. Porikli. "Performance evaluation of object detection and tracking systems", *IEEE International Workshop on Performance Evaluation of Tracking and Surveillance (PETS)*, June 2006
- [12] OpenCV Computer Vision Library
<http://www.intel.com/technology/computing/opencv/index.htm>, [Last accessed: August 2008]
- [13] Pets Metrics, <http://www.petsmetrics.net/> [Last accessed: August 2008]
- [14] A. T. Nghiem, F. Bremond, M. Thonnat, and V. Valentin. "Etiseo, performance evaluation for video surveillance systems", *Proceedings of AVSS 2007*, 2007.