Dynamic Performance Measures for Object Tracking Systems

Simon Denman, Clinton Fookes, Sridha Sridharan, Ruan Lakemond
Image and Video Laboratory, Queensland University of Technology
GPO Box 2434, Brisbane 4001, Australia
{s.denman, c.fookes, s.sridharan, r.lakemond}@qut.edu.au

Abstract

Performance evaluation of object tracking systems is typically performed after the data has been processed, by comparing tracking results to ground truth. Whilst this approach is fine when performing offline testing, it does not allow for real-time analysis of the systems performance, which may be of use for live systems to either automatically tune the system or report reliability. In this paper, we propose three metrics that can be used to dynamically asses the performance of an object tracking system. Outputs and results from various stages in the tracking system are used to obtain measures that indicate the performance of motion segmentation, object detection and object matching. The proposed dynamic metrics are shown to accurately indicate tracking errors when visually comparing metric results to tracking output, and are shown to display similar trends to the ETISEO metrics when comparing different tracking configurations.

1. Introduction

Evaluating the performance of tracking systems is presently done after tracking is completed by comparing to annotated ground truth. This allows a large number of metrics to be computed that accurately describe the systems performance. A large number of metrics have been proposed to evaluate object tracking systems offline [7, 9, 2], and tools such as Viper [1] exist to generate ground truth for comparison. However, for a live system it is desirable to be able to receive real-time feedback on the performance of a tracking system. Such feedback could be used to alert staff to when the system is unreliable, or to automatically reconfigure the system to provide better performance.

In this paper we propose three performance measures that are calculated while the tracking system is running. These measures, whilst not intended to be as accurate as metrics computed by comparing to ground truth, allow real-time feedback on system performance. Metrics are proposed for measuring motion detection, object detection and object matching performance.

The proposed metrics are tested by visually comparing the performance metrics to the tracking results to verify that the performance metrics accurately reflect the state of the system, and by comparing the performance metrics for systems with different configurations to the results of ground truth comparison using the ETISEO metrics [7]. The proposed metrics are shown to accurately indicate object tracking errors, and display similar trends to the ETISEO metrics when comparing different tracking configurations. The remainder of this paper is structured as follows, Section 2 outlines the tracking system that is used in this paper, Section 3 details the proposed dynamic performance metrics, Section 4 presents the testing results and Section 5 presents the conclusions.

2. Tracking System

The tracking system proposed in [4] is used in this work. The object tracking system uses a hybrid motion detector-optical flow technique [3] as a basis, and scans for appropriate regions of motion to detect people (see Figure 1). A scalable condensation filter [4] is used to track the people.

The condensation filter uses the input images, the results of the motion detection and progressively updated features for each tracked object to determine the most likely positions for any known tracked objects in the current frame. This information is used to guide the person detection routines which determine their actual locations in the image. The condensation filter is only used on its own for tracking when the object detection fails.

The system can detect and track two types of objects, people and vehicles. Person detection is performed by splitting the image into sub-regions which contain concentrated areas of motion, and then locating heads and fitting ellipses within each region

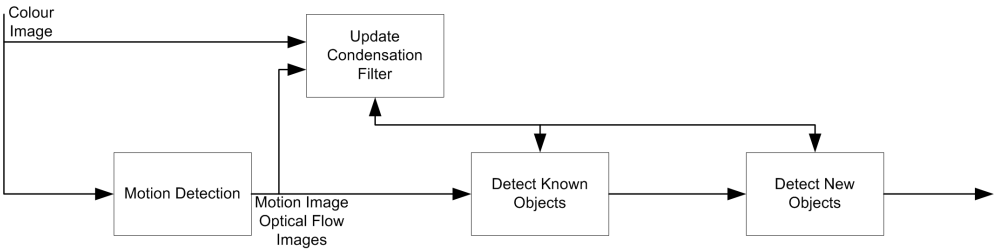


Figure 1. Tracking System Flowchart.

[5, 10]. Working within sub-regions overcomes problems caused by people occupying a common column of the image causing inaccurate vertical projections. Heads are detected by combining the vertical projection and pixel height of the top contour (to aid in overcoming problems caused by holes in the motion image), and finding local maxima; which are then filtered by analysing the surrounding region. Ellipses are fitted to the valid heads at an aspect dependent on the candidate head, and if there is a suitable occupancy (motion within the bounds of the ellipse) the candidate is accepted.

Vehicles are detected by locating large areas of motion, where there is a high concentration of motion pixels in the regions bounding box (i.e. most pixels are in motion), as most vehicles are roughly rectangular in shape. The detection process runs in two stages, the first simply groups large regions of motion together to form a list of initial vehicle candidates. The second analyses this initial list further, checking for overlapping objects to create a list of final vehicle candidates. This final list is then used by the system to update existing tracks and create new tracks.

3. Dyanmic Performance Metrics

During the processing of each frame, there are several outputs that can be used to gauge system performance. Such outputs include motion detection results, object detection results (the number of objects, their type, location) and object matching results (position error between tracked and detected objects, number of predicted objects). Based on these outputs, three dynamic performance metrics are proposed:

1. Motion Detection Performance
2. Object Detection Performance
3. Object Matching Performance

The formulation of these metrics are described in Sections 3.1, 3.2 and 3.3 for the motion detection, object detection and object matching performance metrics respectively. For each metric, a measure is computed for the current frame, which is incorporated into a overall metric,

$$P(t) = P(t-1) + \frac{p(t) - P(t-1)}{L}, \tag{1}$$

where $P(t)$ is the value of the overall performance metric at time t, $p(t)$ is the value of the metric calculated for the current frame, and L is the learning rate.

3.1. Motion Detection Performance

For each frame, the amount of motion detected should be proportional to the number of objects being tracked. The total observed motion in the frame at time t is

$$m_{obs}(t) = \sum_{i,j}^{X,Y} M(i,j,t), \tag{2}$$

where m_{obs} is the amount of motion observed in the motion image, M, which is $X \times Y$ pixels in size. M is a binary image, equal to 1 for pixels in motion, and 0 for all others.

The total amount of motion that is expected can be calculated using the list of tracked objects. For each class of object, c, the amount of motion expected to be associated with it is specified by a occupancy ratio and tolerance, $E_{occ}(c) \pm T_{occ}(c)$, such that the expected motion resulting from a single object, $O(n,c,t)$ (where n is the object's identifier and c object's class), is

$$e_{upper}(n,t) = Area(O(n,c,t)) \times (E_{occ}(c) + T_{occ}(c)), \tag{3}$$

$$e_{lower}(n,t) = Area(O(n,c,t)) \times (E_{occ}(c) - T_{occ}(c)), \tag{4}$$

where $e_{upper}(n,t)$ and $e_{lower}(n,t)$ are the upper and lower bounds for the amount of motion expected to arise from the presence of $O(n,c,t)$, and $Area(O(n,c,t))$ is the area of the object's bounding box. The total expected motion in the frame is

$$m_{upper}(t) = \sum_{n=1}^{N} e_{upper}(n,t), \tag{5}$$

$$m_{lower}(t) = \sum_{n=1}^{N} e_{lower}(n,t), \tag{6}$$

where $m_{upper}(t)$ and $m_{lower}(t)$ are the upper and lower bounds for the total expected motion, and N is the

total number of objects in the frame. The error in motion for the frame then becomes

$$m_{lower}(t) \leq m_{obs}(t) \leq m_{upper}(t); m_{err}(t) = 0, \quad (7)$$

$$m_{lower}(t) > m_{obs}(t); m_{err}(t) = \frac{m_{obs}(t) - m_{lower}(t)}{m_{obs}(t)} \quad (8)$$

$$m_{upper}(t) < m_{obs}(t); m_{err}(t) = \frac{m_{upper}(t) - m_{obs}(t)}{m_{obs}(t)}, \quad (9)$$

where $m_{err}(t)$ is the percentage error in the motion segmentation for the frame. The error at time t is incorporated into the global average, $M_{err}(t)$, the equation 1. When the motion segmentation is performing as expected, $M_{err}(t)$ will be 0. Negative values of $M_{err}(t)$ indicate that there is less motion than expected being detected, whilst positive values indicate that too much motion is being detected.

3.2. Object Detection Performance

Object detection is performed for each object class being tracked, resulting in the object list, $O(c,t)$, of size $n(c,t)$ for the object class c. Ideally, $O(c,t)$ should contain the number of objects of class c presently in the scene, $N(c,t)$. The number of objects detected, when compared to the number of objects present, is used to determine the performance of the object detection,

$$N(c,t) < \alpha;\ b(c,t) = 1, \quad (10)$$

$$b(c,t) = 1 - \frac{min(max(|N(c,t) - n(c,t)| - \alpha, 0), N(c,t))}{max(max(|N(c,t) - n(c,t)| - \alpha, 0), N(c,t))} \quad (11)$$

where $b(c,t)$ is the performance measure for the object detection at time t. A tolerance of α objects is allowed (within the proposed system α is set to 1) when comparing the number of detections. This tolerance ensures that when the system contains no objects, the appearance of an object does not result in the performance of the system dropping significantly.

The performance for a given frame, $b(c,t)$, is incorporated into a global performance metric, $B(c,t)$, using Equation 1. When the system is performing well, $B(c,t)$ should be equal to 1, and will move towards 0 as performance decreases. This metric can also be calculated in dependant of object class.

3.3. Object Matching Performance

Once objects have been detected, they are matched to the list tracked objects from the previous frame. Whilst the tracking system has a certain level of tolerance for object matching, ideally the detected objects should be very similar in size and position to the detected objects they are being matched to. Large differences in bounding box size, or the objects median position (centre of the bounding box) may indicate poorer performance from the tracking system. The median position is used rather than the centroid as it is more stable (the centroid is dependant on the motion detection, and can shift substantially due to changes in the motion detection result, despite no or very little change in the bounding box position).

When matching a tracked object, $T_{obj}(i,c,t)$ to a detected object, $D_{obj}(j,c,t)$, the position (median pixel) and size (bounding box area and aspect ratio) are compared to determine if there is a match between the objects (colour may also be used if there is a high level of uncertainty such as a poor match or multiple candidates). The average error between matched objects can be used to monitor the performance of the object detection and matching. The error at the median pixel can be expressed as

$$e_{median}(i,j,c,t) = \frac{|T_{obj}(i,c,t).x - D_{obj}(j,c,t).x|}{T_{obj}(i,c,t).w} + \frac{|T_{obj}(i,c,t).y - D_{obj}(j,c,t).y|}{T_{obj}(i,c,t).h}, \quad (12)$$

where $e_{median}(i,j,c,t)$ is the error in the median pixel position as a percentage of $T_{obj}(i,c,t)$'s size; and the error between bounding box area is

$$e_{area}(i,j,c,t) = \frac{|Area(T_{obj}(i,c,t)) - Area(D_{obj}(j,c,t))|}{Area(T_{obj}(i,c,t))}, \quad (13)$$

where $e_{area}(i,j,c,t)$ is the error in the bounding box area as a percentage of $T_{obj}(i,c,t)$'s size.

In each case, a ratio between the absolute error and the tracked object is used. This is used rather than the absolute error to ensure size invariance. In most situations surveillance cameras are mounted such that there is significant perspective distortion. If absolute errors are used, then the perspective will mean that the errors from objects near the camera will dominate when calculating the overall metric.

An average error, $e_{median}(c,t)$ and $e_{area}(c,t)$ for the median and area errors respectively, is calculated for the frame for all objects belonging to each object class,

$$e_{median}(c,t) = \frac{\sum_{i=1}^{C} e_{median}(i,c,t)}{C} \quad (14)$$

$$e_{area}(c,t) = \frac{\sum_{i=1}^{C} e_{area}(i,c,t)}{C} \quad (15)$$

where C is the total number of matches made for the class c. The results for the frame ($e_{median}(c,t)$ and $e_{area}(c,t)$) are incorporated into the overall metrics, $E_{median}(c,t)$ and $E_{median}(c,t)$, using Equation 1. Like the object detection metric (see Section 3.2), this metric can also be calculated for the all classes simultaneously. When the system is functioning well, the metric values should be close to 0, and will become increasingly large when the system performs poorly. These values however will be capped by the thresholds that are set for matching the objects, and as such the metrics will not indicate when object matches are failing.

4. Results

The proposed metrics are tested using a portion of the ETISEO database [6]. The dataset ETI-VS2-RD6 is used. This datasets shows a roadway with a mix of pedestrians and vehicles. The performance metrics are evaluated in two ways:

1. By comparing the performance metric scores to the visual tracking output to confirm that errors in tracking are measured by the performance metrics;

2. By comparing the performance metric scores of different tracking configurations to the ground truth comparison scores for the same configurations, configurations that perform poorly according to the performance metrics should also perform poorly when compared to the ground truth.

Three configurations of the tracking system are processed to evaluate the performance metrics (A, B, C, with A being the best performing and C the worst). Each configuration is tuned to different level of performance to test the metrics under different performance conditions. Configuration B is used when comparing the metrics to the visual output.

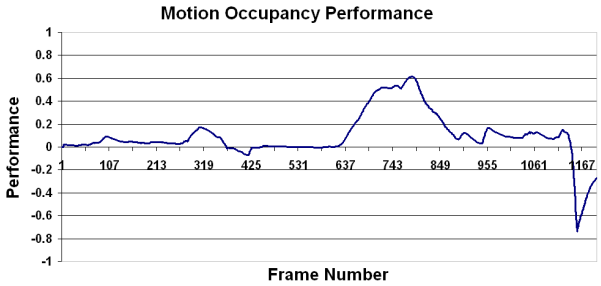
4.1. Comparisson between performance metrics and visual tracking results

Performance metric results for the RD6 dataset are shown in Figure 2. For the evaluation, the object detection metric (see Section 3.2) is calculated for all object classes due to the smaller number of objects in each frame, and the object matching metric (see Section 3.3) is calculated for each object class separately due to the different motion characteristics of the two classes (people and vehicles). The learning rate for all metrics is set to $L = 50$.

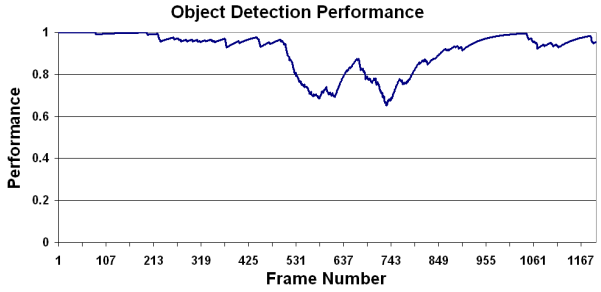
As Figure 2 shows, the tracking system performs at different levels throughout the sequence.

Figure 3 shows an example of the tracking output from frames 250 to 350, during which time the performance metrics show the system to be performing well. As can be seen in the sample output, the system is able to track the objects well. As a result, the performance metrics show relatively little error during this time.

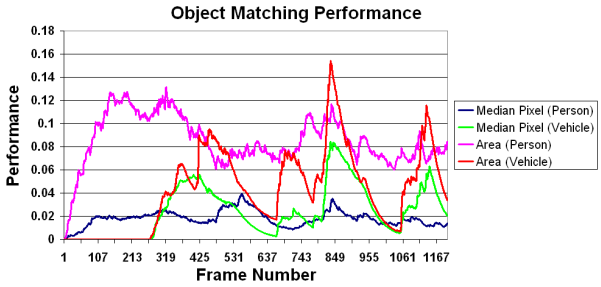
Figure 4 shows an example of the tracking output from frame 630 to 730. During this time, the object detection and motion occupancy performance decreases significantly. During this time, a vehicle stops to let a passenger out. The system fails to properly track the vehicle during this time, and struggles to initially detect the person as they leave the vehicle. The unreliable detection results in several missed detections, which accounts for the drop in the object detection metric. The unreliable detection and poor localisation of the stopped vehicle causes the expected motion to drop, resulting in the error in the motion occupancy



(a) Motion Occupancy Metric Performance



(b) Object Detection Metric Performance



(c) Object Matching Performance

Figure 2. Performance Metrics for RD6

(a) 250 (b) 270 (c) 290

(d) 310 (e) 330 (f) 350

Figure 3. Sample Tracking Output - Tracking performing correctly.

(a) 630 (b) 650 (c) 670

(d) 690 (e) 710 (f) 730

Figure 4. Sample Tracking Output - Errors tracking a stopped vehicle.

metric.

(a) 800 (b) 820 (c) 840

(d) 860 (e) 880 (f) 900

Figure 5. Sample Tracking Output - Failure to initially separate two vehicles that enter together.

Figure 5 shows an example of the tracking output from frames 800 to 900. During this time, the object matching metrics (for both people and vehicles) increase significantly. Figure 5 shows that this increase is due to errors detecting, and then tracking, the two vehicles that enter the scene together. Initially, the two vehicles are detected as a vehicle and a person. This error persists for several frames before the erroneous person track is deleted and a correct vehicle track is created. Once the error is corrected (by frame 880), the object matching metrics begin to improve.

(a) 1120 (b) 1140 (c) 1160

Figure 6. Sample Tracking Output - Failure to remove an object until several frames after it has exited.

Another significant drop in the motion occupancy metrics occurs around frame 1150. Figure 6 shows example output from this time. It can be seen that a vehicle exits the scene during this period, and the system continues to track the vehicle as it exits. The amount of motion associated with the vehicle decreases as it exits the scene, resulting in their being less motion detected than there is expected. As their are few other objects being tracked, the errors relating to the exiting vehicle dominate resulting in a negative spike in the motion occupancy metric.

4.2. Comparisson between performance metrics and ground truth evaluation results

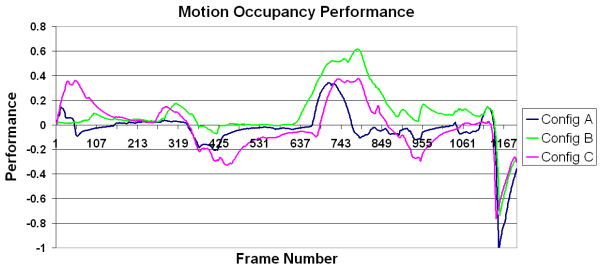
The tracking results of each configuration are evaluated using the ETISEO evaluation tool [6], to verify that performance trends indicated by the dynamic metrics are also visible in the tracking results. The ETISEO evaluation defined several metrics for gauging the performs of tracking systems, split into metrics for detection, localisation, tracking, classification and event recognition. Each group of metrics contains several sub-metrics to evaluate specific criteria and a global metric, which is the average of all metrics within the group. Our evaluation will simply use the overall metrics for detection, localisation and tracking to evaluate the system performance. All metrics yield a value in the range $[0, 1]$, with 1 being a perfect result, and 0 being complete failure. Detailed information on how the metrics are formulated can be found in [8].

For the performance metrics, the mean squared error is used to give an indication of overall performance. Table 1 shows the results of the ground truth evaluation and the MSE for the performance metrics. Figure 7 shows the motion occupancy and object detection metrics for the three different configurations.

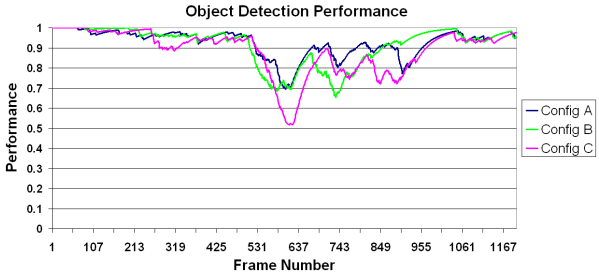
As Table 1 and Figure 7 shows, the performance measures are able to indicate increasingly poor performance in the same manner as the ground truth comparison. Configuration C clearly performs the worst in both sets of metrics. Configuration A outperforms B when considering the ETISEO overall tracking metric and the dynamic motion detection and object detection metrics, however B outperforms A when considering object detection. This can be attributed to the manner in which configuration B is de-tuned (compared to A) by increasing the motion detection sensitivity. This results in better object detection (improved ETISEO overall motion), but more false motion (increased motion detection dynamic metric) and poorer tracking due to more false tracks being spawned. The matching errors for the three configurations are similar for the three configurations, however as previously stated, this met-

Config	ETISEO Metrics			Dynamic Performance Metrics (MSE)					
	Detection	Localisation	Tracking	Motion Occupancy	Object Detection	Matching Error			
						Median Pixel (Person)	Median Pixel (Vehicle)	Area (Person)	Area (Vehicle)
A	0.63	0.92	0.53	0.031	0.010	0.001	0.001	0.010	0.004
B	0.66	0.93	0.50	0.047	0.019	0.000	0.001	0.007	0.003
C	0.53	0.92	0.45	0.042	0.025	0.001	0.001	0.013	0.002

Table 1. Comparisson between ETISEO Evaluation Metrics and Dynamic Performance Metrics



(a) Motion Occupancy Metric Performance



(b) Object Detection Metric Performance

Figure 7. Comparisson of performance metrics for different tracking configurations

ric only considers matches that are made, and so does not consider the increase in failed matches that is indicated by the ETISEO overall tracking metric.

5. Conclusions and Future Work

In this paper, we have proposed three dynamic performance metrics for an object tracking system, that provide real time information on the performance of tracking system. We have shown that these metrics accurately reflect the performance of the tracking system, both in terms of the tracking output and the performance when compared to the ground truth. Future work will focus on developing further performance metrics, and using the metrics to dynamically tune the tracking system to improve tracking performance.

References

[1] Viper-gt, the ground truth authoring tool, http://vipertoolkit.sourceforge.net/docs/gt/.

[2] L. M. Brown, A. W. Senior, Y.-l. Tian, J. Connell, A. Hampapur, C.-F. Shu, H. Merkl, and M. Lu. Performance evaluation of surveillance systems under varying conditions. In *IEEE Int'l Workshop on Performance Evaluation of Tracking and Surveillance*, Colorado, 2005.

[3] S. Denman, V. Chandran, and S. Sridharan. Adaptive optical flow for person tracking. In *Digital Image Computing: Techniques and Applications*, pages 8–8, Cairns, Australia, 2005.

[4] S. Denman, T. Lamb, C. Fookes, S. Sridharan, and V. Chandran. Multi-sensor tracking using a scalable condensation filter. In *International Conference on Signal Processing and Communication Systems (ICSPCS)*, volume 1, pages 429–438, Gold Coast, QLD, 2007.

[5] I. Haritaoglu, D. Harwood, and L. Davis. An appearance-based body model for multiple people tracking. In *15th International Conference on Pattern Recognition*, volume 4, pages 184–187, Barcelona, Spain, 2000.

[6] A. T. Nghiem, F. Bremond, and M. T. V. Valentin. Etiseo, performance evaluation for video surveillance systems. In *IEEE Conference on Advanced Video and Signal Based Surveillance (AVSS)*, pages 476–481, London, UK, 2007.

[7] Silogic. Etiseo - video understanding evaluation, 2007 2005.

[8] Silogic and Inria. Etiseo metrics definition (http://www-sop.inria.fr/orion/etiseo/download.htm). Technical report, 6th January 2006.

[9] F. Yin, D. Makris, and S. Velastin. Performance evaluation of object tracking algorithms. In *10th IEEE International Workshop on Performance Evaluation of Tracking and Surveillance (PETS2007)*, Rio de Janeiro, Brazil, 2007.

[10] T. Zhao and R. Nevatia. Tracking multiple humans in complex situations. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(9):1208–1221, 2004.