Visual Tracking: An Experimental Survey

Arnold W. M. Smeulders, *Senior Member, IEEE,* Dung M. Chu, *Student Member, IEEE,* Rita Cucchiara, *Senior Member, IEEE,* Simone Calderara, *Senior Member, IEEE,* Afshin Dehghan, *Student Member, IEEE,* and Mubarak Shah, *Fellow, IEEE*

Abstract—There is a large variety of trackers, which have been proposed in the literature during the last two decades with some mixed success. Object tracking in realistic scenarios is a difficult problem, therefore, it remains a most active area of research in computer vision. A good tracker should perform well in a large number of videos involving illumination changes, occlusion, clutter, camera motion, low contrast, specularities, and at least six more aspects. However, the performance of proposed trackers have been evaluated typically on less than ten videos, or on the special purpose datasets. In this paper, we aim to evaluate trackers systematically and experimentally on 315 video fragments covering above aspects. We selected a set of nineteen trackers to include a wide variety of algorithms often cited in literature, supplemented with trackers appearing in 2010 and 2011 for which the code was publicly available. We demonstrate that trackers can be evaluated objectively by survival curves, Kaplan Meier statistics, and Grubs testing. We find that in the evaluation practice the F-score is as effective as the object tracking accuracy (OTA) score. The analysis under a large variety of circumstances provides objective insight into the strengths and weaknesses of trackers.

Index Terms—Object tracking, tracking evaluation, tracking dataset, camera surveillance, video understanding, computer vision, image processing

1 Introduction

Visual tracking is a hard problem as many different and varying circumstances need to be reconciled in one algorithm. For example, a tracker may be good in handling variations in illumination, but has difficulty in coping with appearance changes of the object due to variations in object viewpoints. A tracker might predict motion to better anticipate its speed, but then may have difficulty in following bouncing objects. A tracker may make a detailed assumption of the appearance, but then may fail on an articulated object.

Given the wide variety of aspects in tracking circumstances, and the wide variety of tracking methods, it is surprising that the number of evaluation video sequences is generally limited. In the papers on tracking appearing in TPAMI or in CVPR 2011, the number of different videos is only five to ten. The length of the videos maybe long, one to fifteen minutes, but in five to ten different videos few of the above conditions will all be adequately tested.

- *A. W. M. Smeulders is with the Informatics Institute, University of Amsterdam, and also with the Centrum Wiskunde & Informatica, Amsterdam 1098, The Netherlands. E-mail: arnold.smeulders@cwi.nl.*
- *D. M. Chu is with the Informatics Institute, University of Amsterdam, Amsterdam 1098, The Netherlands. E-mail: chu@uva.nl.*
- *R. Cucchiara and S. Calderara are with the Faculty of Engineering of Modena, University of Modena and Reggio Emilia, Modena 41100, Italy. E-mail: {rita.cucchiara, simone.calderara}@unimore.it.*
- *A. Dehghan and M. Shah are with the School of Electric Engineering and Computer Science, University of Florida, Orlando, FL 32816-2365 USA. E-mail: afshin.dn@gmail.com; shah@eecs.ucf.edu.*

Manuscript received 28 Nov. 2012; revised 17 Oct. 2013; accepted 31 Oct. 2013. Date of publication 19 Nov. 2013; date of current version 13 June 2014. Recommended for acceptance by I. Reid.
For information on obtaining reprints of this article, please send e-mail to: reprints@ieee.org, and reference the Digital Object Identifier below.
Digital Object Identifier 10.1109/TPAMI.2013.230

The limited number of videos used for tracking is even more surprising given the importance of tracking for computer vision. In almost any video analysis task, tracking will play a role. Tracking has indeed progressed to impressive, even amazing, individual results like tracking of the motor bicycle in the dirt or the car chase [1]. But as long as most tracking papers still use a limited number of sequences to test the validity of their approach, it is difficult to conclude anything on the robustness of the method in a variety of circumstances. We feel the time is ripe for an experimental survey for a broad set of conditions.

The aim of the survey is to assess the state of the art in object tracking in a video, with an emphasis on the accuracy and the robustness of tracking algorithms. As there has been little conceptual consensus between most of the methods, we have sought to describe the state of the art at the other end: the data. We have composed a real-life dataset as diverse as we could consider and have recorded the performance of all selected trackers on them. We aim to group methods of tracking on the basis of their experimental performance. As a side effect, we aim to evaluate the expressivity and inter-dependence of tracking performance measures.

We have gathered 315 video fragments in the Amsterdam Library of Ordinary Videos dataset, named ALOV++, focusing on one situation per video to evaluate trackers' robustnesses. To cover the variety of (dynamic) situations, we have opted for short but many sequences with an average length of 9.2 seconds. We have supplemented the set with ten longer videos, between one to two minutes each. Eventful short sequences may be considered harder for trackers than long sequences, as the tracker has to adapt to the (hard) circumstances quickly. The current dataset appears to range from easy for all trackers to very hard for all.

The trackers in the survey cover a diverse set of methods. The condition is that access has been provided to the code. Half of the nineteen different trackers are from 1999 to 2006, diverse and well-cited. The other half has appeared in the major conferences in recent years. From a given bounding box in the first frame, the same one for all trackers, the evaluation records how well the tracker is capable of following the target by comparing the computed bounding box with the ground truth bounding box annotated in the complete dataset every fifth frame. As a secondary goal, evaluation metrics for single target tracking are reviewed and compared experimentally.

As a large variety of circumstances is included in many video sequences and a wide variety of trackers is included in the pool of trackers, we propose to perform objective evaluation of the performance. The evaluation of performance is done by objective testing to avoid subjective assessment usually used in current tracking papers with limited number of testing sequences. At the level of the overall performance, this is achieved by plotting survival curves with the associated Kaplan Meier statistics. To evaluate the score per individual video we use Grubbs test for outliers to compare the trackers relatively. Groups of videos are sorted by similar aspects in the recording circumstances and evaluated by correlation analysis.

In this survey we focus on the class of online trackers for which a bounding box in the initial frame is provided. We do not consider pre-trained trackers for which a target model is known before the start. These methods may use active shape or active appearance models [2] which tend to grow in complexity to capture the continuous deformations of the objects [3]. As a consequence, these methods need to pay attention to drifting [4]. Pre-trained tracking is a different problem for which the performance not only depends on the frames of videos but also on the training data, an issue we try to avoid here. We also do not consider offline tracking, which allows for global optimization of the path, scanning forwards and backwards through the frames of a video. While offline tracking is relevant in the medical and other domains, [5]–[7], we focus on the even larger application areas of online tracking. For forward-backward scanning, the methods of evaluation will remain largely the same as the ones proposed here.

The aim of the survey is to assess the accuracy and robustness of single object trackers. This is in contrast to tracking multiple objects where data association needs to be established among different objects, which assures that individual tracks are not being confused. Data association can be formulated as a frame-by-frame local optimization of bi-partite graph matching [8] or as a global optimization of a k-partite graph [5], by a minimum-cost flow [9] or by GMCP [7]. The multi-object association requires different aspects of evaluation than the ones needed here. Also, most of multiple object trackers assume object detection in all frames have already been performed.

Online tracking is a hard problem where all the information in the sequence is needed, especially in the initial frames. We have left out the class of specialized class of contour-based trackers [10]. Even though contour-based trackers provide more detailed shape and deformation of object as it is being tracked, there are severe difficulties in the initialization of the contour [11], in occlusions, and in the robustness against abrupt object motion. Contour morphing is applied in [12], [13], but in general abrupt motion still is an open problem [2]. To acquire and maintain a contour in the general case is a formidable problem by itself, making contour-based tracking currently suited for dedicated applications as face or lip tracking [14].

The main contribution of this work is the systematic analysis and the experimental evaluation of online trackers. We demonstrate that trackers can be evaluated objectively by statistical tests, provided that sufficiently many trackers and sufficiently many data are being used. An important finding of our evaluation is that the top performing trackers do not share a common underlying method. They perform well regardless whether they are modern or mature, whether they are based on target matching or discriminate foreground from background, or what the nature of their updating mechanisms are. As the best trackers are still remote from the ideal combination, we provide an objective analysis broken down over a variety of practical circumstances, the sum of which shows the strengths and weaknesses of the trackers.

2 Related Work

Tracking is one of the most challenging computer vision problems, concerning the task of generating an inference about the motion of an object given a sequence of images. In this paper we confine ourselves to a simpler definition, which is easier to evaluate objectively: *tracking is the analysis of video sequences for the purpose of establishing the location of the target over a sequence of frames (time)* starting from the bounding box given in the first frame.

2.1 Tracking Survey Papers

Many trackers have been proposed in literature, usually in conjunction with their intended application areas. A straightforward application of target tracking is surveillance and security control, initially provided with radar and position sensor systems [15] and then with video surveillance systems. These systems are built on some typical models, namely object segmentation (often by background difference), appearance and motion model definition, prediction and probabilistic inference. For instance, [16] provides an experimental evaluation of some tracking algorithms on the AVSS, conference on advanced video and signal based surveillance, dataset for surveillance of multiple people. The focus of such reviews as is narrower still, as in [17] which discusses tracking specific targets only, such as sport players. The survey of [18] is on tracking lanes for driver assistance. Other surveys address robot applications where tracking based on a Kalman filter is well suited [19]. Yet others are focusing on a single type of target, such as humans [20], [21]. Other tracking methods are designed for moving sensors as used in navigation [22]. Recently, a survey is presented for a wired-sensor network, focusing on the capability of methods to give a simple estimation for the position of the object [23].

Few reviews exist for surveying the performance of application independent trackers. The work of 2006 of Yilmaz *et al*. [10] still provides a good frame of reference

for reviewing the literature, describing methodologies on tracking, features and data association for general purposes.

The above mentioned survey and comparison papers are limited in the number of trackers, limited in the scope of the trackers and/or limited in the circumstances under which the trackers are put to the test. To the best of our knowledge, no such survey of the experimental and systematic evaluation of tracking algorithms is available in literature at the moment.

2.2 Data for Tracker Evaluation

Most papers in their experimental evaluation use a limited number of videos. For example, only six videos are being used in [24]. And, the often used BoBoT dataset, tested in [25], consists of ten different video sequences. A much larger set is found in the famous CAVIAR dataset [26], which was initially created to evaluate people tracking and detection algorithms with few but long and difficult videos. The dataset includes people walking, meeting, shopping, fighting and passing out and leaving a package in a public place. It is, however, limited to one application only. The i-LIDS Multiple-Camera Tracking Scenario [27] was captured indoor at a busy airport hall. It contains 119 people with a total of 476 shots captured by multiple non-overlapping cameras with an average of four images per person with large illumination changes and occlusions. The dataset is limited to one application and therefore unsuited for this paper. The recent 3DPeS dataset for people re-identification contains videos with more than 200 people walking as recorded from eight different cameras in very long video sequences [28], while the commonly used PETS-series [29] contains many videos divided by problem statement and surveillance application.

For general purpose tracking sometimes a large video benchmark is used such as the TRECVid video dataset. The selection in [30] is restricted to 28 different videos.

Most of the papers use the benchmark dataset to compare a new method with the literature. General experimental evaluations of trackers have been addressed only recently. The work of [31] proposes an interesting tool for evaluating and comparing people trackers.

In this paper we build a broad ALOV++ dataset with more than 300 video sequences aimed to cover as diverse circumstances as possible. For preliminary discussions on the dataset see [32].

2.3 Tracking Evaluation Measures

Many measures for evaluating the tracking performance have been proposed, typically with the comparison against ground truth, considering the target presence and position. This requires a considerable amount of annotation, with the consequence that the amount of videos with ground truth is often limited up to this point.

Erdem *et al*. in 2004 [33] proposed performance measures without ground truth by evaluating shape and color differences of the results. This is effective if and only if the tracking results in a reliable segmentation. This is often not the case (and one could wonder whether it is always necessary). Other proposals of evaluation without ground truth are based on comparing the initial and the final positions of object [33]–[36]. This measure evaluates only one aspect of tracking. A hybrid method is proposed in [37]. However, the paper performs the evaluation with only three video sequences, one of which is synthetic. In [35], a complete methodology of predicting the tracker accuracy on-line is evaluated on a single tracker with a small, heterogeneous dataset of existing popular video. A similar method is analyzed in [34] with eleven trackers but again on very few videos and specialized in multiple people tracking. Providing a ground truth set may be beneficial to the community.

The Performance Evaluation of Tracking and Surveillance, PETS, workshop series was one of the first to evaluate trackers with ground truth, proposing performance measures for comparing tracking algorithms. Other performance measures for tracking are proposed by [38] and [24], as well as in [36]. In the more recent PETS series, [29], VACE [39], and CLEAR [40] metrics were developed for evaluating the performance of multiple target detection and tracking, while in case of single object tracking evaluation there is no consensus and many variations of the same measures are being proposed. Here we provide a survey of the most common measures used in single target tracking.

The three basic types of errors in tracking are:

- Deviation: the track's location deviated from the ground truth.
- False positive: tracker identifies a target which is not a target.
- False negative: tracker misses to identify and locate the target.

A reasonable choice for overlap of target and object is the PASCAL criterion [41]:

$$\frac{|T^i \cap GT^i|}{|T^i \cup GT^i|} \geq 0.5, \tag{1}$$

where T^i denotes the tracked bounding box in frame i, and GT^i denotes the ground truth bounding box in frame i. When Eq. 1 is met, the track is considered to match with the ground truth [42]. In many works this PASCAL overlap measure is adopted without threshold, similar to the similarity metric without threshold called Dice in [36]. We prefer to use it with threshold as it makes it easier to evaluate large sets of sequences.

For n_{tp}, n_{fp}, n_{fn} denoting the number of true positives, false positives and false negatives in a video, $precision = n_{tp}/(n_{tp} + n_{fp})$, and $recall = n_{tp}/(n_{tp} + n_{fn})$. The F-score combines the two, see for example [43]

$$F = 2 \cdot \frac{precision \cdot recall}{precision + recall}. \tag{2}$$

The F-score is similar to the Correct Track Ratio in [44]. A variant of the F-score, the area-based $F1$-score [13], [31], is defined as:

$$F1 = \frac{1}{N_{frames}} \sum_i 2 \cdot \frac{p^i \cdot r^i}{p^i + r^i}, \tag{3}$$

where p^i and r^i are defined by $p^i = |T^i \cap GT^i|/|T^i|$, $r^i = |T^i \cap GT^i|/|GT^i|$. It provides insight in the average coverage of the tracked bounding box and the ground truth bounding box.

TABLE 1
Overview Characteristics of the Evaluation Metrics

Name	Equation	Aim	Measure
F-score [43]	$2 \cdot \frac{precision \cdot recall}{precision + recall}$	Accuracy	Thresholded precision and recall
$F1$-score [31]	$\frac{1}{N_{frames}} \sum_i 2 \cdot \frac{p^i \cdot r^i}{p^i + r^i}$	Accuracy	Precision and recall
OTA [40]	$1 - \frac{\sum_i (n^i_{fn} + n^i_{fp})}{\sum_i g^i}$	Accuracy	False positive and false negative
OTP [39]	$\frac{1}{\lvert M_s \rvert} \sum_{i \in M_s} \frac{\lvert T^i \cap GT^i \rvert}{\lvert T^i \cup GT^i \rvert}$	Accuracy	Average overlap over matched frames
ATA [31]	$\frac{1}{N_{frames}} \sum_i \frac{\lvert T^i \cap GT^i \rvert}{\lvert T^i \cup GT^i \rvert}$	Accuracy	Average overlap
$Deviation$ [47]	$1 - \frac{\sum_{i \in M_s} d(T^i, GT^i)}{\lvert M_s \rvert}$	Location	Centroid normalized distance
PBM [31]	$\frac{1}{N_{frames}} \sum_i \left[1 - \frac{Distance(i)}{T_h(i)} \right]$	Location	Centroid L1-distance

Developed in the CLEAR consortium [39], [40], *MOTA* evaluates the tracking performance for multiple objects with respect to false positives and false negatives and ID switching. Adapting *MOTA* to single object tracking, we consider the object tracking accuracy metric:

$$OTA = 1 - \frac{\sum_i (n^i_{fn} + n^i_{fp})}{\sum_i g^i}, \tag{4}$$

where g^i denotes the number of ground truth bounding boxes in frame i: g_i is either 0 or 1. *OTA* indicates how much tracking bounding boxes overlap with the ground truth bounding boxes. In the same consortium, *OTP* was defined as object tracking precision, which is similar to the mean Dice measure in [34]:

$$OTP = \frac{1}{|M_s|} \sum_{i \in M_s} \frac{|T^i \cap GT^i|}{|T^i \cup GT^i|}, \tag{5}$$

where M_s denotes the set of frames in a video where the tracked bounding box matches with the ground truth bounding box. The VACE metrics [31], [39] define the average tracking accuracy, *ATA*, slightly different from *OTP*:

$$ATA = \frac{1}{N_{frames}} \sum_i \frac{|T^i \cap GT^i|}{|T^i \cup GT^i|}. \tag{6}$$

The authors in [45] and [46] use Deviation, the error of the center location expressed in pixels as a tracking accuracy measure:

$$Deviation = 1 - \frac{\sum_{i \in M_s} d(T^i, GT^i)}{|M_s|}, \tag{7}$$

where $d(T^i, GT^i)$ is the normalized distance between the centroids of bounding boxes T^i and GT^i.

The paper [31] provides a large collection of evaluation parameters such as the position-based measure (PBM). This measure estimates the positional accuracy of the object by the mean distance between the centers of the ground truth bounding box and tracked bounding box. Let $T_h(i) = (width(T^i) + Height(T^i) + width(GT^i) + Height(GT^i))/2$. If GT^i and T^i overlap at least partially, $Distance(i)$ is defined as the $L1$-norm distance between $C(GT^i)$ and $C(T^i)$, where $C(X)$ denotes the center of X. If the bounding boxes do not overlap, $Distance(i)$ is defined as $T_h(i)$. The position-based measure is:

$$PBM = \frac{1}{N_{frames}} \sum_i \left[1 - \frac{Distance(i)}{T_h(i)} \right]. \tag{8}$$

The literature is divided into two types of measures for precision and recall: one based on the localization of objects as a whole such as the *F*-score or *OTA*, and one based on the pixels. The latter approach is useful only when the precise object segmentation is available at the pixel level. For instance, in shadow detection [47] pixel level evaluation is advised [48]. We will focus on object level evaluation since no segmentation is available. Candidates for measuring the capacity to track are the *F*-score, *F*1-score, *OTA*, *ATA* and *PBM*, in the sense of holding on the target. One way to consider both accuracy and precision at the same time is the curve which shows the *F*-score values for different overlap thresholds. As the overlap threshold decreases, the F-score values increases, indicative for the fact that trackers are capable of tracking a target for longer time but with lower precision. *Deviation* and *OTP* measure the capability of a tracker to determine the correct position of the target. We will evaluate the effectiveness of these measures over the trackers and all data. Table 1 summarizes the evaluation metrics used in our evaluation.

3 Trackers Selected for the Experimental Survey

In this survey we aimed to include trackers from as diverse origin as possible to cover the current paradigms. We have selected the nineteen trackers with an established reputation as demonstrated by the number of times they have been cited, supplemented with trackers which have appeared in the major conferences in recent years. The choice of trackers was restricted to the ones for which access to the code has been provided. Fig. 1 provides a reference model for the trackers evaluated in this paper.

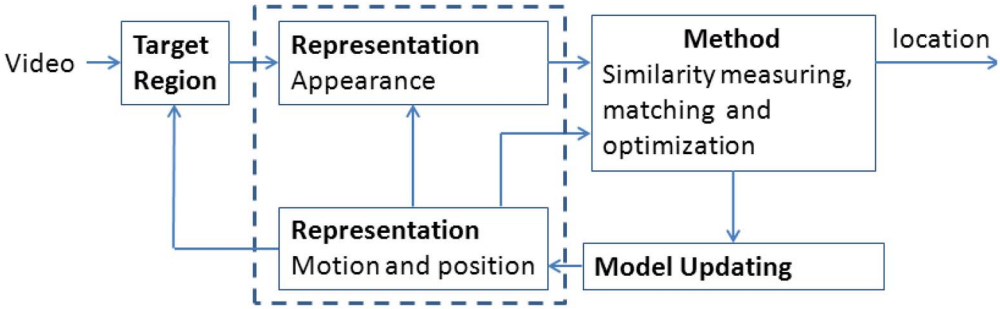

Fig. 1. Reference model of trackers with the five most important components.

3.1 Tracking Using Matching

The trackers in the first group perform a matching of the representation of the target model built form the previous frame(s).

[NCC] Normalized Cross-Correlation: A most basic concept of tracking is direct target *matching* by normalized cross-correlation which uses the intensity values in the initial target bounding box as template. We use the version of [49] where a fast algorithm is presented. At each frame, the tracker samples candidate windows uniformly around the previous target position. Each candidate window is compared against the target template using normalized cross correlation. The candidate with the highest score is selected as the new target location. In this version of NCC, no updating of the target template takes place.

[KLT] Lucas-Kanade Tracker: Ahead of his time by a wide margin, the tracker finds the *affine-transformed match* between the target bounding box and candidate windows around the previous location. The affine transformation is computed by incremental image alignment based on spatiotemporal derivatives and warping capable of dealing with scale, rotation and translation. The location of the target is determined by mapping the target position in the previous frame to the location in the current frame using computed affine transformation. We use the computationally efficient version in [50].

[KAT] Kalman Appearance Tracker: The paper in [51] addresses appearance change by *appearance-predicted matching* for handling target change under occlusion. The target region is represented by 20x20 template-intensities each of which is associated with a separate Kalman filter (with the parameters pooled over all predictors). In an additive Gaussian noise model, the filter predicts the development of each template-intensity over time. The target motion is modeled by a 2-D translation at a single scale and searched in an area around the previous target position. In the subsequent frame, candidate windows around the predicted target position are reduced to a 20×20 template and compared with the predicted template, while penalizing large differences to reduce the effect of outliers. The candidate with the lowest overall difference is selected. In KAT, the template is updated with the newly (predicted) target.

[FRT] Fragments-based Robust Tracking: The tracker in [52] pursues *matching the ensemble of patches* the target bounding box is broken into. In this way partial occlusions and pose changes can be handled patch-by-patch. A fixed array of 10 by 2 concatenated patches, here called fragments, keep track of the changes. When a new frame arrives, the tracker selects candidate windows uniformly around the previous position including scale changes by shrinking and enlarging the target by 10%. Each candidate window is fragmented into the same twenty patches, each represented by an intensity histogram and compared to the corresponding patch in the target region by the Earth Movers Distance. To be robust against outliers from the occluded patches, the 25% smallest score over all patches is selected as the final score to represent the candidate. In FRT, the candidate window with smallest score wins the new target position. The target is not updated.

[MST] Mean Shift Tracking: The famous tracker [53] performs *matching with histograms* rather than using any spatial information about the pixels, making it suited for radical shape changes of the target. It represents the target by an RGB-color histogram. For each new frame, the tracker compares candidate windows with the target region on the basis of the Bhattacharyya metric between their histograms. In order to find the best target location in the new frame, mean shift is used to find the mode of a function, the one which maximizes the Bhattacharyya distance. MST uses the target histogram formed in the first frame without any update during tracking.

[LOT] Locally Orderless Tracking: The tracker [42] offers adaptation in object appearance by *matching with flexible rigidity*. Given the initial bounding box, the target is segmented into super pixels. Each super pixel is represented by the center of mass and average HSV-values. The target's state is sampled using a Particle Filter with a Gaussian weight around the previous position. Each particle corresponds to a candidate window for which the super pixels are formed. The likelihood of each window is derived from a parameterized Earth Mover's Distance (see also FRT above) between the super pixels of the candidate and the target windows where the parameters determine the flexibility of the target. The new target state is the likelihood-weighted sum over all windows. Updating is done via the parameters of the noise model and the parameters of the EMD.

3.2 Tracking Using Matching with Extended Appearance Model

An essential step forward, for long term tracking especially, was the idea of maintaining an extended model of the target's appearance or behavior over the previous frames. This comes at the expense of having to search for the best match both in the image and in the extended model of appearance variations.

[IVT] Incremental Visual Tracking: The tracker in [54] recognizes that in tracking it is important to keep an *extended model of appearances* capturing the full range of appearances of the target in the past. Eigen Images of the target are computed by incremental PCA over the target's intensity-value template. They are stored in a leaking memory to slowly forget old observations. Candidate windows are sampled by Particle Filtering [55] from the motion model, which is a Gaussian distribution around the previous position. The confidence of each sample is the distance of the intensity feature set from candidate window to the target's Eigen image subspace. The candidate window with the minimum score is selected.

[TAG] Tracking on the Affine Group: The paper [56] also uses an *extended model of appearances*. It extends the traditional {translation, scale, rotation} motion types to a more general 2-dimensional affine matrix group. The tracker departs from the extended model of IVT adopting its appearance model including the incremental PCA of the target intensity values. The tracker samples all possible transformations of the target from the affine group using a Gaussian model.

[TST] Tracking by Sampling Trackers: The paper [45] observes that the real-world varies significantly over time, requiring the tracker to adapt to the current situation. Therefore, the method relies on tracking by sampling many trackers. In this way it maintains an *extended model* of trackers. It can be conceived as the extended equivalence of IVT. Each tracker is made from four components: an appearance model, a motion model, a state representation and an observation model. Each component is further divided into sub-components. The state of the target stores the center, scale and spatial information, the latter further subdivided by vertical projection of edges, similar to the FRT-tracker. Multiple locations and scales are considered. Sparse incremental PCA with leaking of HSI- and edge-features captures the state's appearance past over the last five frames, similar to IVT. Only the states with the highest Eigen values are computed. The motion model is composed of multiple Gaussian distributions. The observation model consists of Gaussian filter responses of the intensity features. Basic trackers are formed from combinations of the four components. In a new frame, the basic tracker with the best target state is selected from the space of trackers.

3.3 Tracking Using Matching with Constraints

Following major successes for sparse representations in the object detection and classification literature, a recent development in tracking reduces the target representation to a sparse representation, and performs sparse optimisation.

[TMC] Tracking by Monte Carlo sampling: The method [43] aims to track targets for which the object shape changes drastically over time by *sparse optimization* over patch pairs. Given the target location in the first frame, the target is modeled by sampling a fixed number of target patches that are described by edge features and color histograms. Each patch is then associated with a corresponding background patch sampled outside the object boundaries. Patches are inserted as nodes in a star-shaped graph where the edges represent the relative distance to the center of the target. The best locations of the patches in the new frame are found by warping each target patch to an old target patch. Apart from the appearance probability, the geometric likelihood is based on the difference in location with the old one. The new target location is found by maximum a posteriori estimation. TMC has an elaborate update scheme by adding patches, removing them, shifting them to other locations, or slowly substituting their appearance with the current appearance.

[ACT] Adaptive Coupled-layer Tracking: The recent tracker [57] aims for rapid and significant appearance changes by *sparse optimization in two layers*. The tracker constraint changes in the local layers by maintaining a global layer. In each local layer, at the start, patches will receive uniform weight and be grouped in a regular grid within the target bounding box. Each layer is a gray level histogram and location. For a new frame, the locations of the patches are predicted by a constant-velocity Kalman-filter and tuned to its position in the new frame by an affine transformation. Patches which drift away from the target are removed. The global layer contains a representation of appearance, shape and motion. Color HSV-histograms of target and background assess the appearance likelihood per pixel. Motion is defined by computing the optical flow of a set of salient points by KLT. The difference between the velocity of the points and the velocity of the tracker assesses the likelihood of the motion per pixel. Finally, the degree of being inside or outside the convex hull spanned around the patches gives the likelihood of a pixel. The local layer uses these three likelihoods to modify the weight of each patch and to decide whether to remove the patch or not. Finally, the three likelihoods are combined into an overall probability for each pixel to belong to the target. The local layer in ACT is updated by adding and removing patches. The global layer is slowly updated by the properties of the stable patches of the local layer.

[L1T] L1-minimization Tracker: The tracker [58], employs *sparse optimization by L1* from the past appearance. It starts using the intensity values in target windows sampled near the target as the bases for a sparse representation. Individual, non-target intensity values are used as alternative bases. Candidate windows in the new frame are sampled from a Gaussian distribution centered at the previous target position by Particle Filtering. They are expressed as a linear combination of these sparse bases by L1-minimization such that many of the coefficients are zero. The tracker expands the number of candidates by also considering affine warps of the current candidates. The search is applied over all candidate windows, selecting the new target by the minimum L1-error. The method concludes with an elaborate target window update scheme.

[L1O] L1 Tracker with Occlusion detection: Advancing the *sparse optimization by L1*, the paper [59] uses L2 least squares optimization to improve the speed. It also considers occlusion explicitly. The candidate windows are sorted on the basis of the reconstruction error in the least squares. The ones above a threshold are selected for L1-minimization. To detect occluded pixels, the tracker considers the coefficients of the alternative bases over a certain threshold to find pixels under occlusion. When more than 30% of the pixels are occluded, L1O declares occlusion, which disables the model updating.

3.4 Tracking Using Discriminative Classification

A different view of tracking is to build the model on the distinction of the target foreground against the background. Tracking-by-detection, as it is called, builds a classifier to distinguish target pixels from the background pixels, and updates the classifier by new samples coming in.

[FBT] Foreground-Background Tracker: A simple approach to the incremental *discriminative classifier* is [60] where a linear discriminant classifier is trained on Gabor texture feature vectors from the target region against feature vectors derived from the local background, surrounding the target. The target is searched in a window centered at the previous location. The highest classification score determines the new position of the target in FBT. Updating is done by a leaking memory on the training data derived from old and new points in the target and in the surrounding. We use the version in [61] with color SURF features rather than intensities as in the reference.

[HBT] Hough-Based Tracking: The recent tracker [62] aims at tracking non-rigid targets in a *discriminative classifier with segmentation* of the target. A rectangular bounding box will introduce many errors in the target/background labels into the supervised classifier, especially for non-rigid and articulated targets. Therefore, the authors aim to locate the support of the target through back projection from a Hough Forest. A Hough Forest is an extension of a Random Forest [63] by a vector d, measuring for positive samples the displacement of each patch to the target center, similar to the R-table [64] in Generalized Hough transforms. The target given in the first frame is used to train the target and the background. The Lab-color space, first and second derivatives of x- and y-direction, and a histogram of gradients are used as features for learning a rich appearance model. The Hough Forest provides a probability map of the target. The pixel with the maximum score is selected as the center of the target. The sparse pixels voting for that location are used to segment the target using the grabcut algorithm [65] and hence generate positive examples to start HBT anew in the next frame.

[SPT] Super Pixel tracking: The recent method [66] embeds the *discriminative classifier in super pixel clustering*. The purpose is to handle changes in scale, motion, and shape with occlusion. The HSI-histograms of the super pixels are extracted in the first 4 frames from the extended target region. Using mean-shift clustering, super pixels are grouped on the basis of the super pixel histograms with a cluster confidence derived from the overlap of the cluster with the target bounding box. In the new frame, candidate windows are sampled weighed according to a Gaussian distribution around the previous location. The search space is enlarged by considering different scales of the candidate windows. The super pixel confidence is derived from the cluster confidence it belongs to and from the distance the super pixel has to the center of the cluster. The candidate window with the highest confidence summed over all super pixels in the window is selected as target. SPT updates the target model every 15th frame.

[MIT] Multiple Instance learning Tracking: The paper [46] recognizes the difficulty in taking the current tracker region as the source for positive samples and the surrounding as the source for negative samples as the target may not completely fill the bounding box or cover some of the background. Therefore, it learns a *discriminative classifier* [67] from positive and negative bags of samples. In the MIL-classifier, the target bounding box supplemented with other rectangular windows at close range is grouped into the positive bag. Multiple negative bags are filled with rectangular windows at a further distance. Haar features are used as features. Candidate windows are sampled uniformly in a circular area around the previous location. The highest classification score determines the new position in the MIT. In the update of the classifiers, the old classifier parameters are updated with the input from the new data points.

[TLD] Tracking, Learning and Detection: The paper in [1] aims at using labeled and unlabeled examples for *discriminative classifier* learning. The method is applied to tracking by combining the results of a detector and an optical flow tracker. Given the target bounding box in the first frame, the detector learns an appearance model from many 2bit binary patterns [68] differentiated from patterns taken from a distant background. The authors use the fast Random Ferns [69] to learn the detector. When a new frame arrives, locations with some 50 top detector scores are selected. The optical flow tracker applies a KLT to the target region and proposes a target window in the current frame. The normalized cross correlation is computed for the candidate windows. The system selects the candidate window which has the highest similarity to the object model as the new object. Once the target is localized, positive samples are selected in and around the target and negative samples are selected at further a distance to update the detector target model. If neither of the two trackers outputs a window, TLD declares loss of target. In this way TLD can effectively handle short-term occlusion.

3.5 Tracking Using Discriminative Classification with Constraints

In the tradition of discriminative tracking, the recent paper [70] observes the difficulty in precisely sampling training samples, either labeled or unlabeled, as addressed also by HBT, MIT and TLD by confining to the target area. The paper accurately observes that the classifier pursues the correct classification of pixels which is different from finding the best location of the target.

[STR] STRuck: Structured output tracking with kernels: The *structured supervised classifier* [70] circumvents the acquisition of positively and negatively labeled data altogether, as it integrates the labeling procedure into the learner in a common framework. Given the target bounding box, it considers different windows by translation in the frame. Using these windows as input, the structured SVM (S-SVM) accepts training data of the form {appearance of the windows, translation}. The window's appearance is described by Haar features, like MIT, arranged on a 4x4 grid and 2 scales, raw pixel intensities on a 16x16 rescaled image of the target and intensity histograms. In a new frame, candidate windows are sampled uniformly around the previous position. The classifier computes the corresponding discriminant highest score selected as the new target location. Subsequently, the new data for updating the S-SVM

are derived from the new target location. While updating the S-SVM learner enforces the constraint that the current location still stays at the maximum. Locations which violate the constraint will become support vectors. In practice, as maximization during updating is expensive, the learner uses a coarse sampling strategy.

4 Analysis

Although the various tracking paradigms may be motivated and derived from different aspects of the problem, they can be decomposed into the five components depicted in Fig. 1. We discuss them one by one.

4.1 Target Region

In the general purpose trackers the target is often represented by a *target bounding box*, as in NCC [49] and many of the other trackers, or by an ellipse as in MST [53]. The common argument against a bounding box is that background pixels may be confused with the target pixels. However, advances of object classification [41], [71] have led to the deliberate inclusion of some background in the description of an object as the transition foreground-background provides much information. A further advantage is the low number of parameters associated with a bounding box as the representation of the target area.

Some trackers, not included in our analysis, instead work on *contour of the target* [72]–[74] allowing for maximum freedom to change the object's appearance. They are found in dedicated application areas like pedestrian tracking and bio-medical applications where a precise shape detector may be effective but less so for general applications.

Other trackers use a rough segmentation or *blob*-representation of the target, like HBT [62], [75], enhancing the discrimination from similar objects in the scene [76], [77] especially when the target occupies a small fraction of the surrounding box. This may happen when shape is the final goal, as in gesture analysis. Segmentation while tracking places high demands on the robustness of the segmentation in order not to introduce false information about the target.

Patch-based representations take into account the most informative tiles of the target which are allowed to undergo independent appearance changes. FRT [52] keeps an array of fixed patches, in contrast to ACT [57] which keeps a loose set. In LOT [42] and SPT [66] super pixels are used. In TMC [43] salient pixels are connected in a graph. When using these variations of patches, a precise distinction between target and background is no longer necessary as the information of patches may be turned on or off later in the analysis. The chances of succeeding in tracking dynamically shaped or occluded objects will be bigger with patches, at the expense of losing rigid or small targets for which patches have no advantage over a pixel-wise treatment of the target. When the object has an even appearance, salient points are likely to fail to lock to their positions. The method in [46] provides a potential robust solution by combining patches with many bounding boxes.

Representation by the independently moving *parts* of the target for general tracking is largely uncharted territory. For specific targets, like pedestrians, a wire frame or part description is the natural representation. To detect the structures of the parts from the first few frames is not trivially achievable, but see [78], [79].

The tracking algorithm may derive robustness from sampling more than one target-size box. The strength of keeping *multiple bounding boxes* as the target representation in [58], TLD [1] and L1O [59] is that alternative positions of the target as a whole are being considered, rather than optimizing one central position as the other trackers do.

For the moment, representation of the target by a bounding box or multiple bounding boxes is still an attractive option because of the low model complexity.

4.2 Representation of Appearance

The appearance of the target is represented by visual cues. There are three essentially different representations: a 2D-array like the image data, a 1D-histogram of ordered properties, or a feature vector. The appearance representation implies a constancy of a certain property to transfer one frame to the next. Without any such constancy assumption tracking cannot work.

An *array* of brightness values is still the most often used paradigm like in NCC [49], STR [70] and many others. Constant brightness is realistic for remote objects, but in many real cases, the assumption is violated quickly. In search for alternatives, a variety of color representations has been used: HSI color in LOT [42], HSI gradients in TST [45]. However for general tracking, a large, diverse feature representation may be necessary to capture the distinctive properties of the target.

In *histogram* representations the use of color is common, for example MST [53], TMC [45], HBT [62] and SPT [66], while FRT [52] and ACT [57] even use plain intensity histograms. This can only be successful because the patches they derive the histogram from are small. A histogram removes any spatial order giving maximum room for the flexibility of the target while moving. The spatial information has to be captured elsewhere in the tracking algorithm.

In *feature vectors* at most local order is represented. When the object's shape is important, it has to be encoded by geometric constraints later on. The Haar gradients used in MIT [46] are simple. 2D binary patterns are used in TLD [1], SURF-features in FBT [61], and Lab-color features among others in HBT [62]. The improved identification of the target by using more informative features outweighs the loss of signal quality. Color invariants [80] reduce the influence of the fluctuations in the illumination, which play an important role in tracking [61], for which fast implementations [81], [82] exist. However their use comes at a price. The target can drift more easily since the invariants tend to reduce the shading and hence the gradient magnitude on the target.

It may be useful to keep track of other appearance information in the scene. For static scenes, *background intensity representation* is an old solution [83], [84] only suited for standardized circumstances, improved with background intensity prediction with simple statistics [85]–[87]. The main attraction of the background subtraction is that the tracking algorithm poses neither a constraint on the target's appearance nor on its behavior over time. In an

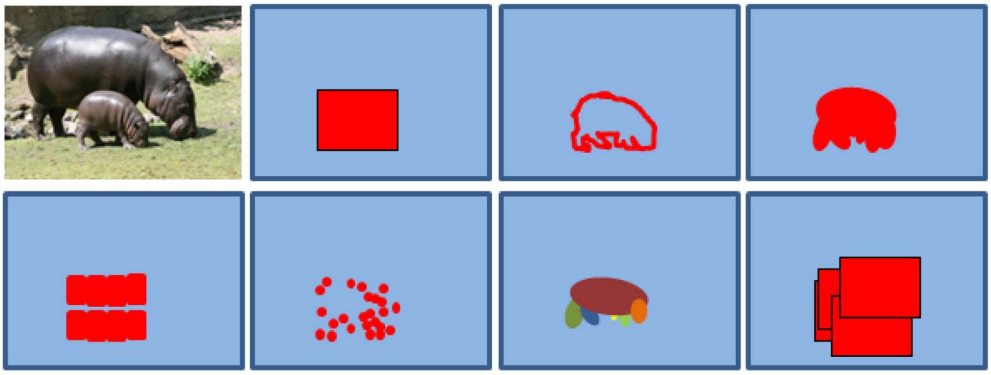

Fig. 2. Target region representations used in tracking. From left to right, top to bottom: target bounding box, target contour, target blob, patch-based, sparse set of salient features, parts, and multiple bounding boxes.

environment where the background is static, such as in surveillance with fixed cameras, it is effective and very fast. A static background is, however, rarely the case in general applications and hence not considered here. It may also be useful to keep track of other objects in the scene for occlusion detection [59], [76], [88]. The tracker in [89] includes *information on confusion*. Other factors to consider are the general illumination conditions and information on the wetness causing specularities to adjust the photometric parameters. There are few trackers which keep *information about the scene* [90]. Finally, it will be useful to keep track of the estimated camera panning, zooming and tilting, see [91].

We have discussed three different groups of appearance representations: array, histogram and feature vector. The large number of alternative representation within each group makes it harder to compare the effectiveness of other parts of the tracking algorithms as well.

4.3 Representation of Motion

For a target moving about, the least committing assumption by FBT [60], STR [70] and many others is that the target is close to its previous location, and the target is found by *uniform search* around the previous position. Explicit position search leaves the object entirely free in its motion pattern, hence is likely to be robust in general tracking, but it may lose the target when motion is fast.

An alternative is the use of a *probabilistic Gaussian motion* model usually centered around the previous position as used in IVT [54] and L1T [58] and many others. Sampling positions is computationally necessary when the search space is large. Since the Gaussian model assigns more weight to locations which are closer to previous position of target, it poses more bias on the motion of the target than uniform search. Hence it will fail easier when the camera is shaking.

To reduce the search space further *motion prediction* is done by a linear motion model, which can be described using Kalman filter [92]–[94]. Prediction of motion as applied in ACT [57] may be helpful to counter full speed targets but is not easily applicable in general tracking. Prediction of motion under the guidance of optical flow [95]–[97] to stabilize the search is an attractive alternative.

Implicit motion prediction as in MST [53], and KLT [50] does not use any constraints on the motion model. Instead it seeks the maximum by using optimization methods. However it requires that the movements of the target are small relative to the appearance changes in the scenes, which is rarely the case in general tracking.

In *tracking and detection* the strength of combining detection with tracking as applied in TLD [1] is interesting. Potentially many proposed candidates from the object detector are matched with the one proposal from the optical flow tracker, making the method robust to wild motions of the target and the camera. Recovery from drift may also prove a useful addition to make tracking robust for general purposes as in SPT [66].

More rich motion models describe rotation [70], scale [50], shear and full 2D-affine or 2D-projective deformation of the target as in TAG [56]. Prediction of position, scale, or motion may be useful in handling extended occlusions should they occur when the object is likely to continue to change and no update of the model is possible.

There are a few recurring patterns in motion models. The old uniform search has not yet lost its attractiveness

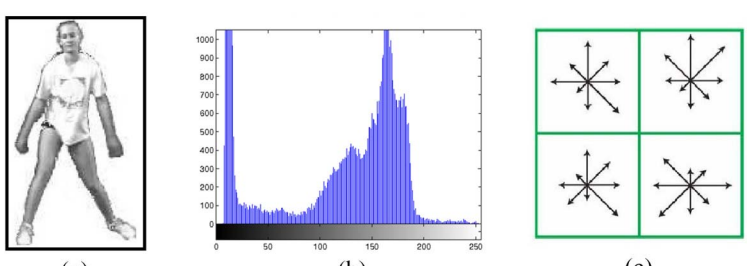

Fig. 3. Appearance representation. (a) 2D-Array ([10]). (b) Histogram. (c) Feature vector.

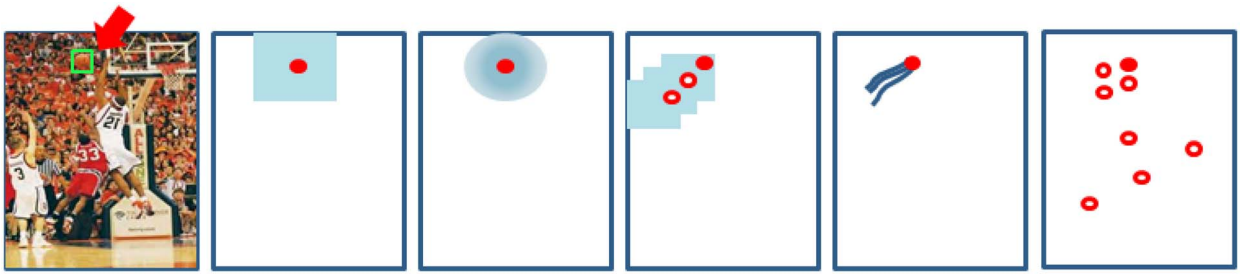

Fig. 4. Motion models used in tracking. From left to right: uniform search, Gaussian motion model, motion prediction, implicit motion model, and tracking and detection.

due to its simplicity. Motion under the guidance of other information like optical flow and recovery from loss is attractive and gaining ground.

4.4 Method

The core of the tracking algorithm is the computational paradigm to find the best location and/or the best state of the target in the new frame. Fig. 5 illustrates different methods used in the trackers evaluated in this survey.

Taking on tracking as an optimization for the best *match* is often found where the method limits the search to direct gradient ascent in KLT [50], or gradient ascent in probabilities in MST [53]. Gradient ascent poses heavy constraints on the appearance of the target and the background in order to avoid local maxima. Therefore, when more computational capacity is available, often a particle filter is used to scan the whole pdf as in IVT [54], TST [43] or TAG [45]. Searching without a priori bias for the best match over the image search space is robust to many possible object motion patterns, provided the search space is not too big. Searching for the best match rests on strong assumptions about the target appearance offering no protection against the intensity and albedo changes frequently seen in general applications. Also, when the initial bounding box is sloppy, the methods are likely to perform less. Searching for the most probable match is likely to work well in tracking when occlusions and confusions occur, and in low contrast images. In constrast, when the object is detectable, probabilistic matching has no advantage over direct matching.

Subspace matching enables the tracker to keep an extended model of the target as it has been observed over the past in IVT [54], or of the state of the trackers as TAG [43] or TST [45] does. The type of extended model dictates the space in which an optimum is sought, which in turn governs the choice of optimization algorithm and the internal representation, see the 3D-model [98] or the medium memory model in [99]. Extended model provides a long-term memory of the object's views, advantageous for long term tracking and motion during occlusion. At the same time they will pose more demands on the inference.

The method of *Constrainted optimization* provides a good foundation for the systematic adaptation to the many varying circumstances if the information in the constraints is reliable. The rules are derived from context as in TAG [56], or from the ensemble of patches as in L1T [58] and L1O [59], or the pose of patches ACT [57]. The new constraining information has to come from the first few frames. Therefore, constrainted optimization may be well suited for pre-trained tracking.

The method of tracking-by-detection is based on the observation that the distinction of the target from the background suffices in tracking on a single object. To that end FBT [60] maintains a simple *discriminative supervised classifier* by incremental Linear Discriminant Analysis, where HBT adds segmentation to randomized forests as the classifier [62]. In tracking the classification is always semi-supervised because beyond the first frame there are only unlabeled samples available from the sequence. Discriminative classification with clustering is used in SPT [66], and MIT [46]. TLD uses a pool of randomized classifiers [1]. Other alternatives are transductive learning as applied in [100]. Discriminative tracking opens the door to robustness in general tracking as very large feature sets can be employed which have been so successful in general object classification. The choice of the classifier is important as in tracking very few samples are available

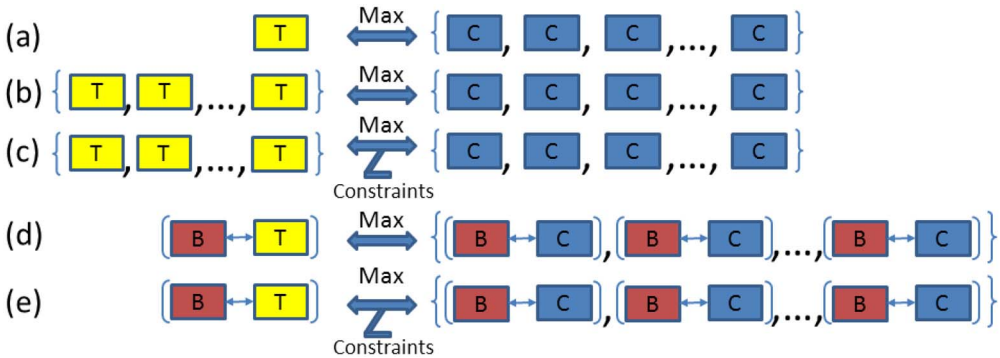

Fig. 5. Tracking methods employed in the trackers. (a) Matching. (b) Matching with extended appearance. (c) Matching with constraints. (d) Discriminative classification. (e) Discriminative classification with constraints. T-boxes represent target instances. B-boxes represent local background instances. And C-boxes represent candidate targets.

at the start by definition. Therefore, a simplistic linear discriminant analysis may be more effective overall than Multiple Instance Learning. A fundamental problem is that wrongly labeled elements of the background may confuse the classifier and lead to drifting. Therefore, representing the target tightly [1] may pay off. Discriminative trackers are robust to changes to the scene as a whole, yet live on the assumption that the target's features are unique in the scene which may fail for groups of objects.

A recent contribution is STR [70] which provides a discriminative classifier which directly maps its output on a continuous scale: the displacement of the target. In this, the *structured classifier* is fundamentally different as it outputs the displacement directly bypassing a label per pixel: target or not.

Many of the newer methods have convoluted models for maximization with many parameters. A more fundamental problem of many parameters is, however, that they need to be estimated. For tracking, the estimation has to be come from the first few frames, before the first change kicks in. For general tracking, the paradox is: while more convoluted models are needed to handle all different circumstances, they are unstable as the number of frames to estimate them robustly is limited.

4.5 Model Updating

Most trackers update the target's model in order to deal with the variation in the appearance. Of the trackers we have considered in our experimental survey, NCC [49], FRT [52] and MST [53] perform *no update* of the target appearance. KLT [101] also does not update the appearance model but starts the search for the best transform from the previously best choice, instead. Not updating the appearance of the target may seem too trivial, yet the advantage of a fixed model is that no false information is inserted in the model either. Especially for short sequences, the performance may be still acceptable.

A common way of updating is to substitute the old template with the *last seen template*, improved with partial updating [102], [103], with leaking [60], or with occasional target rebooting from scratch [104].

Predicting the target's appearance in KAT [51] is conceptually similar to using the last seen template. This comes in handy for long-term occlusions. When the target appearance changes gradually the prediction is more likely to succeed. However under abrupt changes, it becomes difficult for the prediction to adapt sufficiently quick, introducing errors into the updated model.

In tracking on the basis of patches, *patch updates* are needed to insert, to change, to move, or to delete patches in TMC [43], and ACT [57]. In general, these are complex decisions based on intrinsically limited amounts of information. As a consequence the chances of doing it wrong are considerable.

Updating an extended model by incremental PCA has the advantage of holding information from the past in the most condensed form as in IVT [54], TAG [56] and TST [45]. Obviously, the larger the model of the past, the higher the demands on inserting the new information in a reliable way.

Although finding the target has received much attention in target tracking, updating the target and background has received less attention. However attention to target updating is important, since it can easily lead to insertion of false information. A key for general purpose tracking is in restraint mechanisms of updating the model, especially when the tracking is long.

5 Experiments

5.1 Implementation of Trackers

Attempts have been made to extend the selected set even further with trackers for which only binaries were publicly available, all but one (TST) of these attempts have been successful since we were not able to successful run the binaries for our evaluation. In some cases we requested the use of code for evaluation but the authors declined or were unable to participate. Therefore we have limited our choice of trackers for which the code was publicly available, still covering a broad variety of methods, representation and models.

All trackers have been evaluated twice by running experiments at two university labs to be sure the code is stable and results are platform-independent. Making the trackers run without error on our variety of machines required a software installation effort of between 1 day to 1 month per tracker. To ensure that each tracker performs as intended by the authors, we have run the tracker with the videos as mentioned in the respective paper and verified that the performance is identical to the reported performance. Of two trackers, TMC and ACT, we have results until they crashed, which occured on two and fourteen sequences respectively.

All trackers but TLD use a maximum displacement between two consecutive frames to restrict the search region in the motion model. Some use as low as seven pixels [54], while others go as high as 35 pixels [46]. For fair comparison, we have fixed the maximum displacement for all trackers at twenty pixels in both x- and y-direction. For the trackers using a uniform search model, we fix the search region to a 40×40 pixel square centered at the previous target position. For the trackers using a probabilistic motion model, we fix the search region to a Gaussian distribution of standard deviation 11.55 with the same mass of probability as a uniform square of 40×40 pixels. We leave ACT, MST, KLT and TLD as they are, as they do not enforce a maximum displacement.

5.2 Data

We have collected the Amsterdam Library of Ordinary Videos for tracking, ALOV++, aimed to cover as diverse circumstances as possible: illuminations, transparency, specularity, confusion with similar objects, clutter, occlusion, zoom, severe shape changes, different motion patterns, low contrast, and so on (see [32] for a discussion on the aspects). In composing the ALOV++ dataset, preference was given to many assorted short videos over a few longer ones. In each of these aspects, we collect video sequences ranging from easy to difficult with the emphasis on difficult video. ALOV++ is also composed to be upward compatible with other benchmarks for tracking by including 11 standard tracking video sequences from the [105], [106] datasets for the aspects which cover smoothness and occlusion. Additionally, we have selected 11 standard video sequences

TABLE 2
Overview Characteristics of the Trackers Used in This Paper

Tracker	Target region	Appearance model	Motion model	Method	Update
NCC [50]	Bounding box	Array, intensities	Uniform	Match array	No update
KLT [51]	Bounding box	Array, intensities	Uniform	Match array by ascent	No update, previous transform
KAT [52]	Bounding box	Array, intensities	Uniform	Match predicted array	Prediction of appearance, robust
FRT [53]	Patches	Histogram, intensities	Uniform	Match values patch	No update
MST [54]	Bounding box	Histogram, RGB	Implicit from data	Match values	No update
LOT [42]	Patch super pixels	Array, HSIvalues	Gaussian	Match loose geometry	Update appearance, noise and metric
IVT [55]	Bounding box	Array, intensities	Gaussian	Subspace match array	Incremental extended model by iPCA
TAG [57]	Bounding box	Array, intensities	Gaussian	Subspace match array	Incremental extended model by iPCA
TST [46]	Bounding box	Array, HSIgradients	Multiple Gaussians	Subspace match state	Incremental extended model by iPCA
TMC [43]	Patch salient pixels in graph	Histogram, color	Implicit from data	Constraint optimization, two layers	Patch add, delete, modify
ACT [58]	Patches & image	Histogram, intensities	Linear Prediction	Constraint optimization, two layers	Patch add, delete + update global by stable patches
L1T [59]	Multiple boxes	Array, intensities	Gaussian	Constraint optimization in L1	Update bounding boxes
L1O [60]	Multiple boxes	Array, intensities	Gaussian	Constraint optimization L1, robust	Update bounding boxes
FBT [61]	Bounding box	Vector, SURF features	Uniform	Discriminative classifier	Incremental classifier from target
HBT [63]	Blob	Vector, Lab-color gradients and others	Gaussian	Discriminative classifier with segmentation	Incremental classifier from segmented target
SPT [67]	Patch super pixels	Histogram, HSI color	Gaussian	Discriminative in super pixel clustering	Renewed classifier every 15th frame
MIT [47]	Multiple boxes	Vector, Haar gradients	Uniform	Discriminative MIL classifier	Incremental classifier from target
TLD [1]	Multiple boxes	Vector, LBP + Array, intensities	N from detector + 1 from optical flow	Discriminative classifier + optical flow tracker	Incremental classifier from target
STR [71]	Bounding box	Array, intensities and histogram	Uniform	Structured output classifier	Incremental structured classifier from past

frequently used in recent tracking papers, on the aspects of light, albedo, transparency, motion smoothness, confusion, occlusion and shaking camera. 65 Sequences have been reported earlier in the PETS workshop [32], and 250 are new, for a total of 315 video sequences. The main source of the data is real-life videos from YouTube with 64 different types of targets ranging from human face, a person, a ball, an octopus, microscopic cells, a plastic bag or a can. The collection is categorized for thirteen aspects of difficulty with many hard to very hard videos, like a dancer, a rock singer in a concert, complete transparent glass, octopus, flock of birds, soldier in camouflage, completely occluded object and videos with extreme zooming introducing abrupt motion of targets.

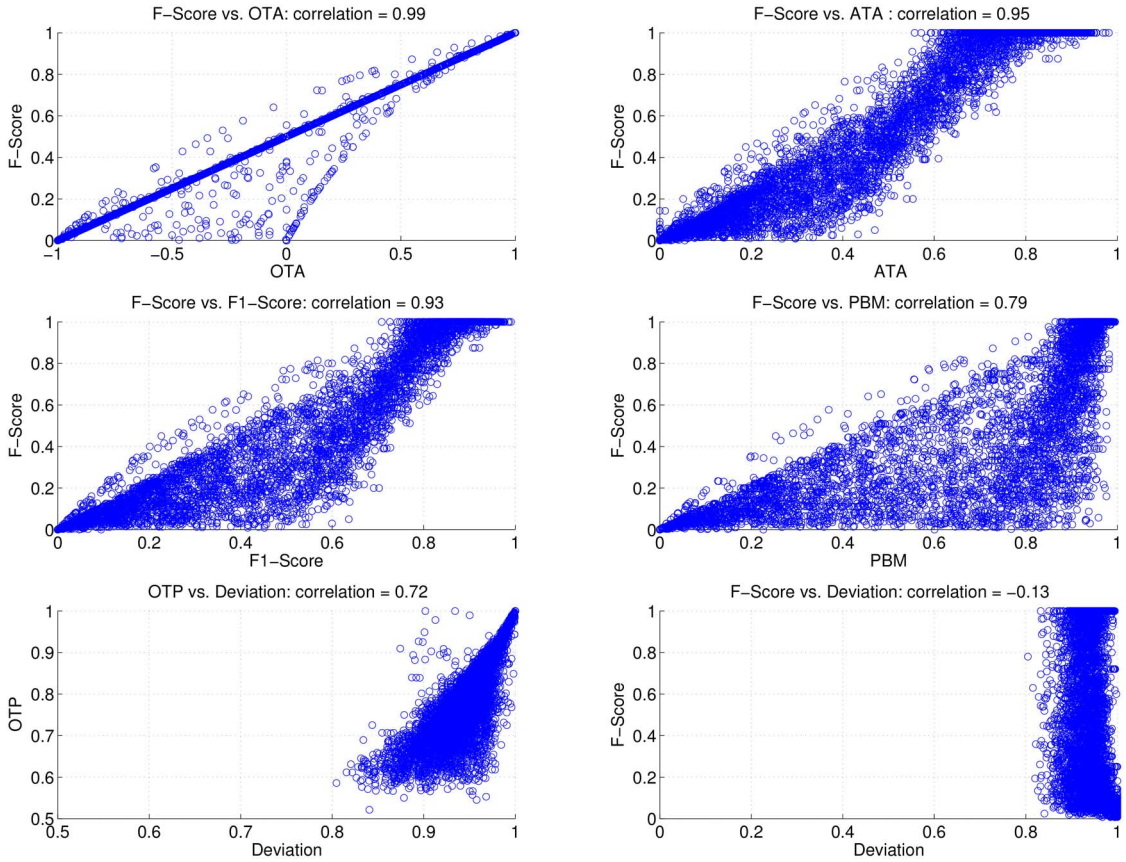

Fig. 6. Scatterplots for all trackers and all video sequences of the F-score (y-axis) versus OTA, ATA, $F1$-score and PBM, the Deviation versus OTP, and F-score versus Deviation.

To maximize the diversity, most of the sequences are short. The average length of the short videos is 9.2 seconds with a maximum of 35 seconds. One additional category contains ten long videos with a duration between one and two minutes. This category comprise of three videos from [1] with a fast-moving motorbike in the desert, a low contrast recording of a car on the highway, and a car-chase; three videos from the 3DPeS dataset [28] with varying illumination conditions and with complex object motion; one video from the dataset in [107] with surveillance of multiple people; and three complex videos from YouTube.

The total number of frames in ALOV++ is 89364. The data in ALOV++ are annotated by a rectangular bounding box along the main axes of flexible size every fifth frame. In rare cases, when motion is rapid, the annotation is more frequent. The ground truth has been acquired for the intermediate frames by linear interpolation. The ground truth bounding box in the first frame is specified to the trackers. It is the only source of target-specific information available to the trackers. ALOV++ is made available through http://www.alov300.org/.

5.3 Evaluation

By having a large video dataset we can provide a comprehensive performance evaluation without discussing each video separately. Thus we avoid the risk of being trapped in the peculiarity of the single video instance. We visualize the performance of a set of videos by sorting the videos according to the outcomes of the evaluation metric. By sorting the videos, the graph gives a bird's eye view in cumulative rendition of the quality of the tracker on the whole dataset. Note that the performance order of videos becomes different for each tracker. These types of plots are referred to as *survival curves* [108], a terminology borrowed from medicine to test the effectiveness of treatments on patients [109]. The survival curve indicates how many sequences, and to what percentage of the frames' length the tracker survives (by the Pascal 50% overlap criterion). The survival curve is estimated by the Kaplan-Meier estimator [110]. The significance of differences between survival curves is measured by log-rank statistics [111], [112]. Given survival curve of tracker i, $\{sc_i(1), sc_i(2), \ldots, sc_i(K)\}$, define $m_i(k) = sc_i(k) - sc_i(k+1)$ as the performance drop at the k-th video. Under the null hypothesis that two survival curves originate from the same distribution, the expected drop is

$$e_i(k) = \big(m_i(k) + m_j(k)\big)\,\frac{sc_i(k)}{sc_i(k) + sc_j(k)}. \tag{9}$$

The sum of the observed drop minus the expected drop $O_i - E_i = \sum_{k=1}^{N} (m_i(k) - e_i(k))$ is analyzed by log-rank statistics:

$$S_{ij} = \frac{(O_i - E_i)^2}{var(O_i - E_i)}. \tag{10}$$

Under the null hypothesis, S_{ij} approximately follows the chi-square distribution with one degree of freedom [111].

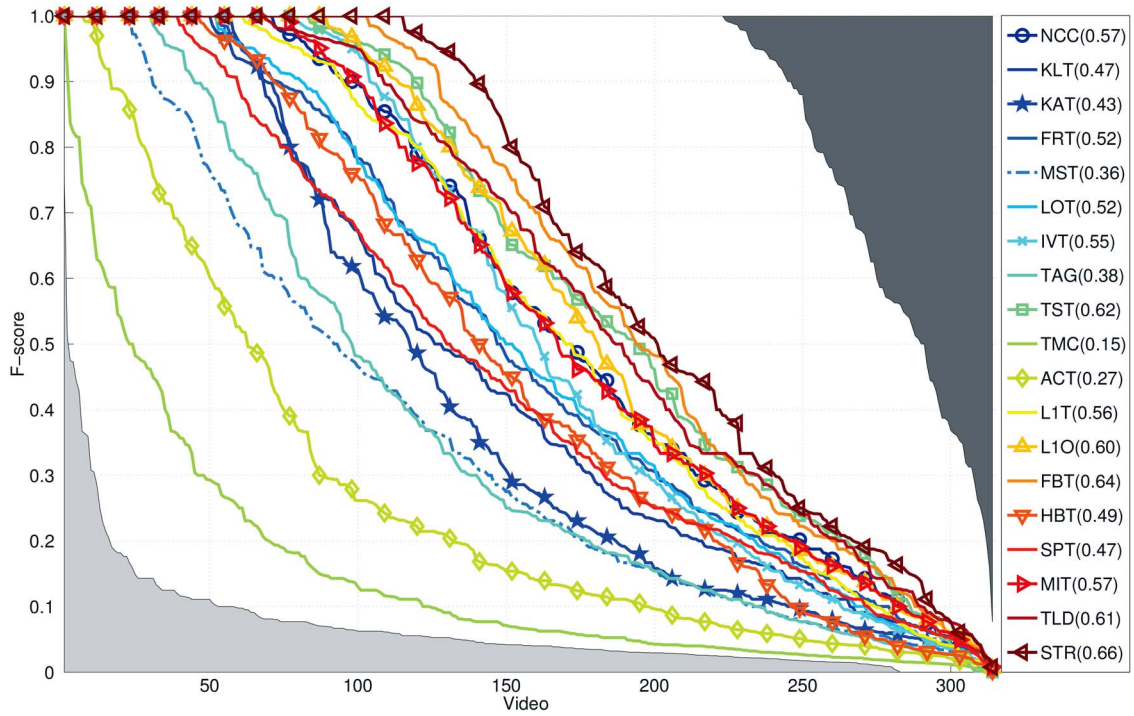

Fig. 7. Survival curves of the trackers with respect to F-scores. Also indicated in the legend are the average F-scores. The shaded area in the upper right corner represents the fraction of the videos in which none of the nineteen trackers was able to track objects in any of the videos. The shaded area on the left bottom represents the fraction of the videos in which all trackers were able to track correctly.

To evaluate the relative performance of the nineteen trackers on each of the 315 video sequences objectively, we have performed the Grubbs outlier test [113] per video to look for an outstanding performance of trackers. We use the one-sided Grubbs statistics:

$$G = \frac{Y_{max} - \bar{Y}}{\sigma}, \tag{11}$$

where $\bar{Y}$ and σ denote the average F-score and standard deviation overall trackers on that video. We label a tracker 'outstanding' when it is significantly better than the rest with a confidence of 99% and its F-score being at least 0.5.

6 Results

6.1 Evaluation Measures

Before getting to the actual tracker performances, we evaluate the effectiveness of evaluation metrics. Fig. 6 shows a plot of the metrics derived from all sequences and all trackers. We obtained similar figures also by testing the videos of the fourteen aspects separately and testing the trackers individually. Thus we include here only the whole results over 5985 runs (nineteen trackers on 315 videos).

The correlation between the F-score and OTA is 0.99. The correlation between F and ATA is 0.95. The correlation between F and $F1$ is 0.93. Hence, it is concluded that F, ATA and $F1$ essentially measure the same performance. We prefer to use the F-score to have a clear distinction between success and failure which makes it easier to evaluate a large dataset.

The use of the thresholded version F is supplemented with a separate measure, Deviation, to measure the tightness of tracking for the frames where the tracking is successful. Fig. 6 also demonstrates that the Deviation metric has a low correlation with the OTP-score. And, the figure shows that the Deviation metric is not correlated with the F-score. The correlation between the F-score and the PBM-score, expressing the relative distance between centroids, is 0.79. This figure suggests that the PBM may contain useful information when the centroid distance is at stake. We will use Deviation as it is uncorrelated to F.

6.2 The Overall Performance of Trackers

The survival curves show large differences in the overall performances of trackers, as shown in Fig. 7. The best overall performances are by STR, FBT, TST, TLD and L1O, in that order. According to the Kaplan Meier log-rank statistics these five are also significantly better than the rest (data not shown). It is remarkable that the leading group of five is each based on a different method: STR is based on a discriminative structured classifier; FBT is a discriminative classifier; TST selects the best of many small trackers; and TLD combines tracking with detection. The target region representation, the appearance representation, the motion model and the update model also vary between the different trackers. Given their very different methods, it is interesting to observe that they are all capable of tracking correctly approximately 30% of the dataset. Number five in performance is L1T based on constrainted optimization, followed by NCC which uses on a plain matching by correlation. This is remarkable

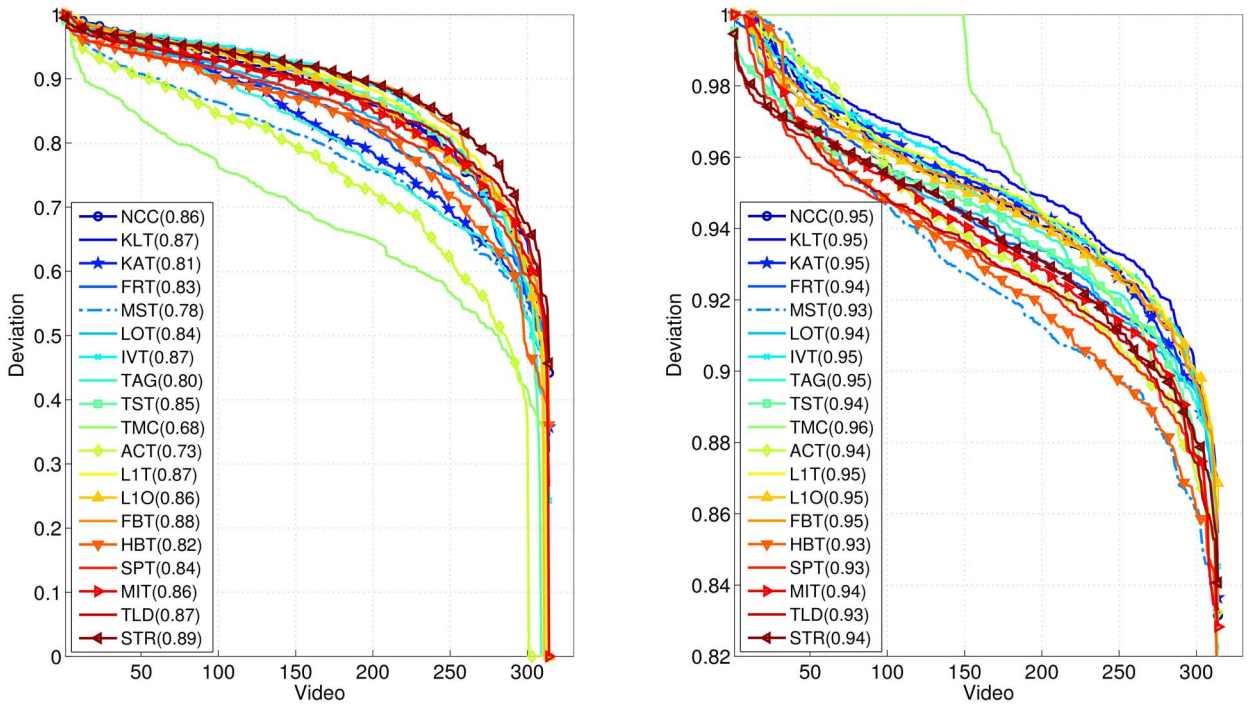

Fig. 8. Survival curves of the trackers with respect to the Deviation Metric. Left subplot: whenever the target and the ground truth overlap. Right subplot: when the target is being tracked correctly, that is with an overlap $> 50\%$. The flat part of the TMC-curve represents the videos for which the tracker loses track in the second frame.

as NCC scores 6th overall while it was designed even before 1995.

Bounding box visualization of the nineteen trackers on all videos can be viewed from website: http://www.alov300.org/

The overlap between survival curves is limited to neighboring curves in the vast majority of cases. This justifies the ordering of the survival curves by one number: the average *F*-score, see the table in Fig. 7. For STR the average *F*-score is 0.66, for FBT 0.64, for TST 0.62, for TLD 0.61 and for L1O 0.60.

Four-fold repetition for the stochastic trackers TMC, IVT, L1T and L1O, on one quarter of the data has shown that the survival curves are similar to the degree that the spread in the mean *F*-score is no more than 0.02 for three trackers and 0.05 for L1T. This indicates that the trackers take sufficiently many samples to achieve stable results.

The result of selecting the best and the worst among the nineteen trackers is also visualized in Fig. 7. The upper right shaded area can be characterized as the too-hard-track-yet videos. The boundary of this area results when taking the ideal combination over all nineteen trackers. And the lower shaded area gives the videos for which all trackers are capable of following the target correctly. This portion of the ALOV++ dataset is less than 7%, whereas the number of videos too hard to track by any tracker is 10%. The dataset offers a good portion of not-yet-conquered challenges for the nineteen trackers.

Comparing the ideal combination with the best individual trackers indicates that there is ample room for improvement. Although performance is impressive, there is no current evident single best solution for general tracking.

6.3 The Accuracy of Tracking

The Deviation Measure measures the accuracy. We observe no correlation of the Deviation Measure with the *F*-score. As can be seen in Fig. 8 KLT, NCC and L1T perform best. KLT and NCC search to maximize the direct match of the target, while L1T aims for the same via a constrainted optimization. The trackers with the least grip on the target are MST, TLD and SPT due to the limited translation-only motion model. As differences are very small, from now on we focus on *F*-scores only.

6.4 Analysis by Aspects of Difficulty

A large dataset allows for an objective evaluation at two levels of trackers rendering an analysis of the factors behind the performance.

The first analysis is by dividing the dataset in fourteen groups of sequences, one for each aspect of difficulty for tracking. Fig. 9 gives the average score over the videos for each aspect. It does so after correcting the scores for the overall difficulty varying from aspect to aspect by calculating the normalized *F*-score of tracker i in aspect j as $t_{ij} = \left(F_{ij} - \mu(F_i)\right)/\sigma(F_i)$; where F_{ij} is the average absolute *F*-score of tracker i in aspect j; $\mu(F_i)$ is the mean of F_{ij} over j and $\sigma(F_i)$ is the standard deviation of F_{ij} over j. The ensemble of t-scores per aspect gives an impression of the strengths and weaknesses of the tracker.

A second important set of objective facts is found in the sequences for which a tracker outperforms all other trackers by a wide margin. A tracker is designated 'outstanding' when, according to Grubbs' test, it is significantly better than all other trackers on that sequence and it has an *F*-score of at least 50%. The list of outstanding results is in Table 3.

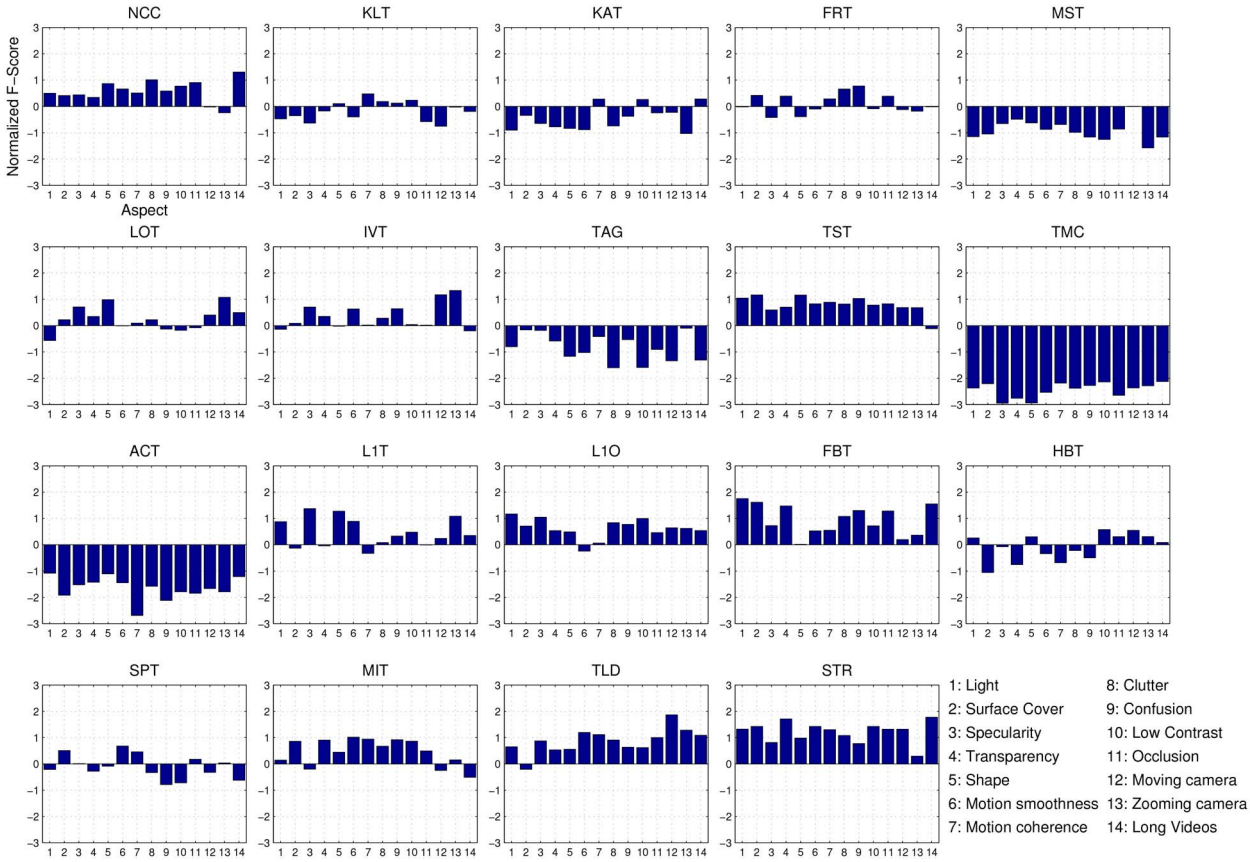

Fig. 9. Normalized F-score t_{ij} of each tracker across data aspects.

6.4.1 The Influence of Illumination Conditions

An important aspect of illumination is the appearance of a shadow in the path of the object. This can be overcome by keeping a model of the target as well as of the local background. When the nineteen trackers are divided into the group which keeps a model of the target only and the alternative group which maintains a model of the target and local background, the performance differs notably. The average F-score for the target-only trackers on the category 'light' in the data is 0.45, while the overall average for the target-and-background group of trackers[1] is 0.58, significant at the 95%-level according to the t-test. FBT is the best with an average F-score of 0.73 in this category. The (local) background is often included successfully in successful methods. As an example, consider the videos 01-32 and 01-33 (see Fig. 10), where the only difficulty is a person traversing from a deep shadow into the full sun or back. The transition is lost by many of the trackers.

6.4.2 The Influence of Changes in the Target's Surface Cover

Keeping a model for the background separately from a model of the target is also advantageous when the surface cover changes drastically over time. For an example see Fig. 10. The average F-score for the target-only trackers on the category 'surface cover' in the data is 0.48, where the overall average for the target-and-background group of trackers FBT, TLD, MIT, HBT, SPT, and STR is 0.62, significant at the 95%-level according to the t-test. FBT is highest, again, with an average F-score of 0.82 in this category.

6.4.3 The Influence of Specularities

In the group of videos with a specular reflection, the L1T tracker achieves outstanding performance on three videos, see Table 3. They contain relatively small reflections, but sufficiently disturbing to cause all others to lose track. L1T is capable of handling small specularities because of the use of single pixel templates which then put off small specular pixels without influencing neighboring pixels. The result is remarkable as L1O is designed as an extension of L1T for the purpose to handle occlusions. The tracker will declare specularities as occlusion and falsely stop tracking.

6.4.4 The Influence of Shape and Size

In general, the performance of trackers on drastic changes in shape is poor. Of all aspects of difficulty, the overall performance on videos with shape changes is lowest. Videos of gymnasts like 06-01, 06-02, 06-19 and 06-20, are lost by all trackers, while the surveillance scene in 06-12 is lost by many. Tracker L1T which is very flexible - i.e. it places few shape constraints in its appearance model - is the best overall in this category.

To evaluate the effect the size of the target may have on the performance of the tracker, we sort the videos in ten equally filled buckets by the width of the target. The results show a substantial effect for MST and ACT, and a noticeable effect for IVT, SPT and LOT, see Fig. 11. The effect

1. Target-and-background trackers are: FBT, TLD, MIT, HBT, SPT, and STR.

TABLE 3
List of Outstanding Cases Resulted from the Grubbs' Outlier Test and with $F \geq 0.5$

Sequence	Tracker	Sequence	Tracker	Sequence	Tracker	Sequence	Tracker
0112	TLD	0411	ACT	1102	TLD	1203	MIT
0115	STR	0510	L1T	1103	HBT	1206	STR
0116	KAT	0512	STR	1104	TLD	1210	TLD
0122	TLD	0601	STR	1107	HBT	1217	TLD
0203	FBT	0611	MST	1112	STR	1218	TLD
0301	L1T	0705	TLD	1116	TLD	1221	TLD
0305	L1T	0901	HBT	1119	TLD	1303	TLD
0312	L1T	0916	STR	1128	TLD	1402	TLD
0314	KAT	0925	STR	1129	FBT	1409	STR
0404	FBT	1020	FBT	1134	FRT		

for MST is attributed to the increased likelihood of getting stuck in a local minimum when the target is small. For IVT and LOT the number of free parameters of the appearance model is among the largest of all trackers. Therefore, they are likely profiting from having a larger target to their disposal to learn an image model. In SPT and LOT, we find some evidence that super pixel representations are less suited for small widths. In contrast, none of the discriminative trackers are found to be sensitive to target size, demonstrating the capacity to normalize the size by going after the difference between the target and its background.

6.4.5 *The Influence of the Target's Motion*

Many challenges remain to be answered still in the motion of the target, both in the smoothness of the motion and the coherence of the motion, as can be observed from Fig. 9 in the relatively poor performances on videos of these types. In tracking, the apparent motion of the target relative to the camera plays an important role. For all seven videos with no apparent motion[2], the average *F*-score of the top

2. The videos with no apparent motion and occlusion are: 11-01, 11-17, 11-21, 11-23, 11-24, 11-25 and 11-30.

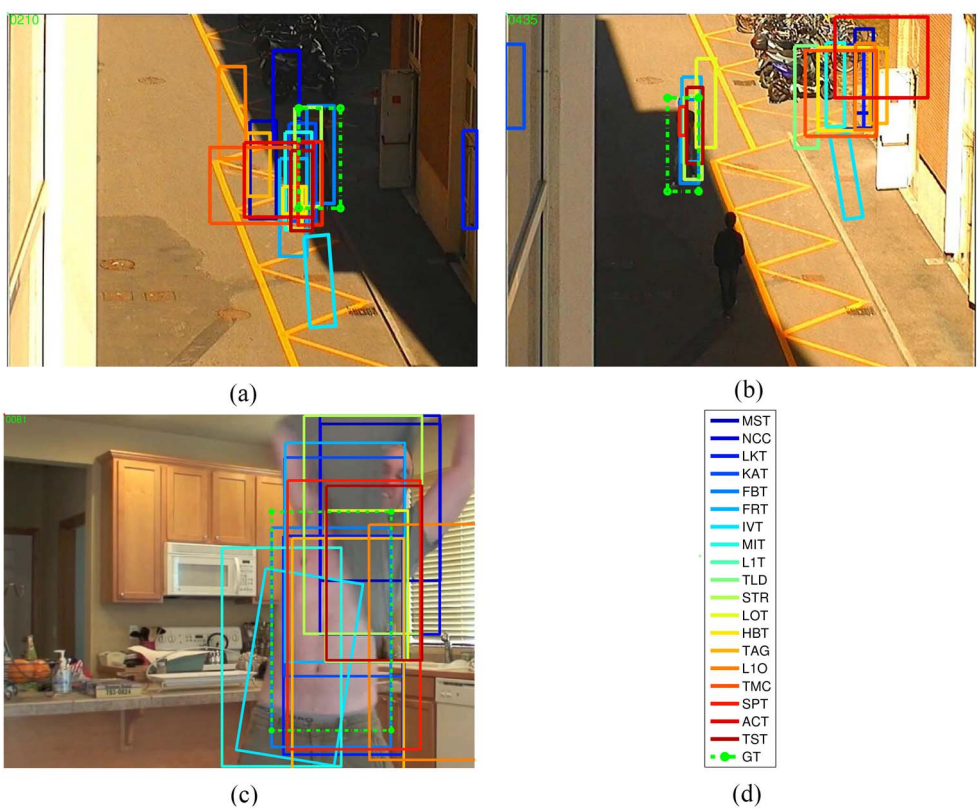

Fig. 10. Illustration of the effect of shadow on the performance of trackers for (a) videos 01-32 ([28]), and (b) 01-33 ([28]), for (c) 02-04 for the change of surface cover. Subplot (d) depicts the color codes of the trackers and the ground truth.

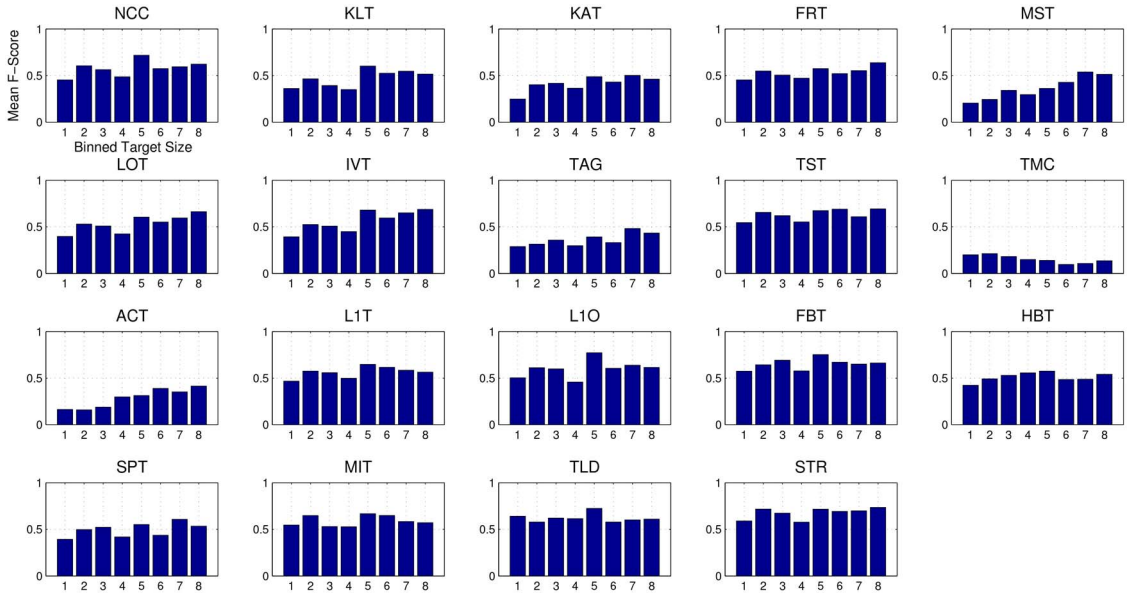

Fig. 11. Correlation of trackers' performance against the target size. The x-axis shows the videos sorted by target size which is distributed into eight equally filled buckets, while the y-axis shows the average F-score.

five trackers STR, FBT, TST, and TLD and L1T on these seven sequences is very high, being 0.77, 0.60, 0.80, 0.90, 0.98, 0.80 and 0.87, respectively. We conclude that no or little apparent motion, even full (but short) occlusions and extreme appearance changes is not much of a problem for modern trackers.

6.4.6 The Influence of Clutter and Confusion

The performance of many trackers in cluttered scenes is generally good even where humans would consider tracking difficult. This is illustrated in video 08-02 where a singer moves before a crowd, or even a cut-out version of Waldo in video 08-06 or the camouflage in 06-12 is followed by many trackers. When there are simultaneous other circumstances such as change of illumination in video 08-09, clutter is a complicating factor (see Fig. 12). For clutter there is no clearly winning method of tracking.

The presence of similar objects like other members of a marching band in sequence 09-03, or sheep in sequence 09-04 are less of problem to many modern trackers to the point where some trackers succeed in following individuals in the New York marathon in video 09-25 where humans may find it difficult to follow (see Fig. 12). As with clutter there is no clearly winning method of tracking for confusion.

6.4.7 The Influence of Occlusion

In the design of trackers, occlusion often plays an important role. We find that in the success of handling occlusions, the motion of the target relative to the camera plays an important role. For all seven videos with no such motion, the occlusion ranges from partial occlusion to short full occlusion. Still the average F-score of the top five trackers STR, FBT, TST, and TLD and L1T on these seven sequences are 0.77, 0.60, 0.80, 0.90, 0.98, 0.80 and 0.87, respectively, very good indeed. We conclude that under no or little motion relative to camera, even full short occlusions are not much of a problem for modern trackers.

For the four videos with relative motion and with light occlusion[3] the average F-score of the top five trackers are 0.88, 1.00, 0.76 and 0.84, respectively. This indicates that occlusion with less than 30% may be considered a solved problem. In contrast, for all seven videos[4] with relative motion and full occlusion, most trackers have difficulty in reacquiring the target when it reappears.

As can be seen in Table 3 there are many different trackers which achieve an outstanding performance by the Grubbsï£¡ test. TLD does this on five different sequences, whereas HBT performs outstandingly on two videos, and three more trackers each on one video. This indicates that TLD is a successful strategy for occlusion, generally. Especially the recovery after the long intermediate shot from a different camera in 11-16, see Fig. 13 is impressive here. At the same time, to be successful under for all occlusions appears to be a big challenge. There are five videos for which a different tracker is outstanding, like the small and fast-moving motor bicycle in 11-07 tracked perfectly by HBT. TLD is best overall on occlusion but there is work left to do.

6.4.8 The Influence of Camera Motion

Camera motion is the one aspect of difficulty for which there is a successful strategy indeed. The detection scheme of TLD in combination with its optical flow tracker is outstanding for four sequences in the category of 'camera motion', see Table 3 and Fig. 9. The camera shakes, introducing very abrupt movements of the object. The model is attractive as the detector searches for the most obvious appearance anywhere in the frame whereas the tracker gives preference for a close candidate. And, the tracker has the ability to recover when it loses track. The performance

3. Videos with apparent motion and no more than 30% occlusion are 11-10, 11-20, 11-22 and 11-27.

4. The videos with relative motion and full occlusion are 11-02, 11-07, 11-28, 11-29, 11-31, 11-33 and 11-34.

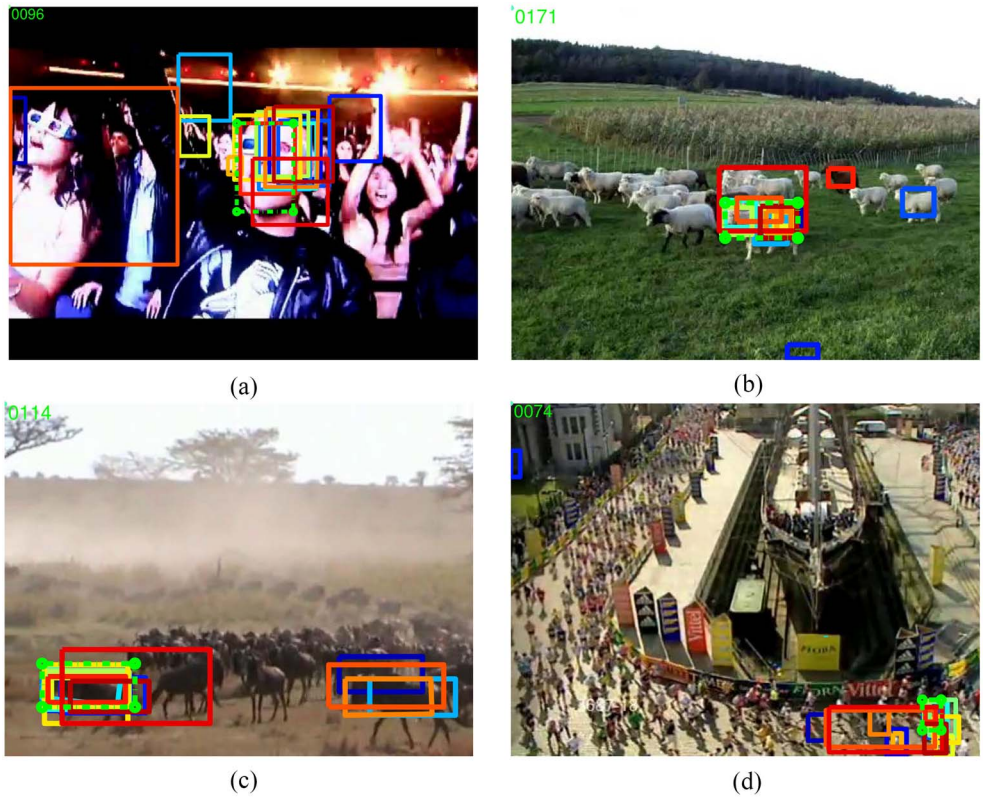

Fig. 12. Illustration of the effect of clutter on the performance of trackers for (a) video 08-09, for (b) 09-04, (c) 09-08, and (d) 09-25 ([96]) for confusion. For the color codes, see Fig. 10.

of TLD is best illustrated by sequence 12-10 where it is the only tracker capable of following the fast panning of the camera on the race track, or sequence 12-17 with a fast rotation in the woods, and sequence 12-18 climbing with a handheld camera. As can be seen in Fig. 9, IVT is relatively good in moving cameras like many other trackers with Gaussian based motion models.

6.4.9 The Influence of Zoom

A zooming camera, or, equivalently an object disappearing into the depth is handled differently by many trackers. As can be seen in Fig. 9, IVT is relatively good in this aspect of tracking. To investigate this further we have selected all twelve videos[5] for which the target shows a gradual change of scale. The eight trackers with a built-in mechanism for changes in scale IVT, TST, L1O, TLD and even including TAG, L1T, HBT, SPT, have a good (relative) performance for zooming, significantly better compared to the average *F*-score of the remaining eleven trackers. We conclude that the implementation of a mechanism to handle a gradual scale change is beneficial for the performance of all trackers.

6.4.10 The Influence of the Length of the Video

The ten of the videos in the category 'long' are between one and two minutes in length. The results are in Fig. 14.

As can be seen in the figure, the best performance is by STR, FBT, NCC and TLD which are also good performers in general. TLD is the only tracker capable of making something out of long sequence 1, which is the back of the motor biker in the desert, while all other trackers fail immediately after the start. In the sequence of the soap, all trackers lose the target when the camera changes, nor are they able to restore tracking when the view returns to the first camera. A most remarkable result on long videos is achieved by NCC. It is the simplest of all trackers under consideration and it does not perform any learning over time or any updating of the model, while performing 3rd overall on the long videos (and 6th over all videos)! This supports the conclusion that simplicity in updating is advised.

6.4.11 Trackers with an Even Performance Overall

As can be seen in Fig. 9, STR has an even performance over most aspects of difficulty, bringing it to the number one position over the entire dataset. It performs well on all aspects but one, the change of scale of the target as it has no mechanism to adjust to that. Also TST has an even, solid performance throughout showing the potential of combining many weak trackers. There is a general breakdown, however, for longer videos which cannot be attributed to occlusion, clutter or shadow. For this tracker, we attribute the weak performance on long videos to the many parameter updates, which in the end inserts false information into the model. L1O performs evenly as well but misses out when the target's motion is no longer smooth. TLD performs reasonably on all aspects apart from changes in surface cover. Finally, NCC which is very simple matching by correlation has a reasonable performance throughout,

5. The videos with gradual changes of the scale of the target are: 13-01, 13-02, 13-06, 13-07, 13-09, 13-10, 13-13, 13-15, 13-17, 13-20, 13-27, and 13-29.

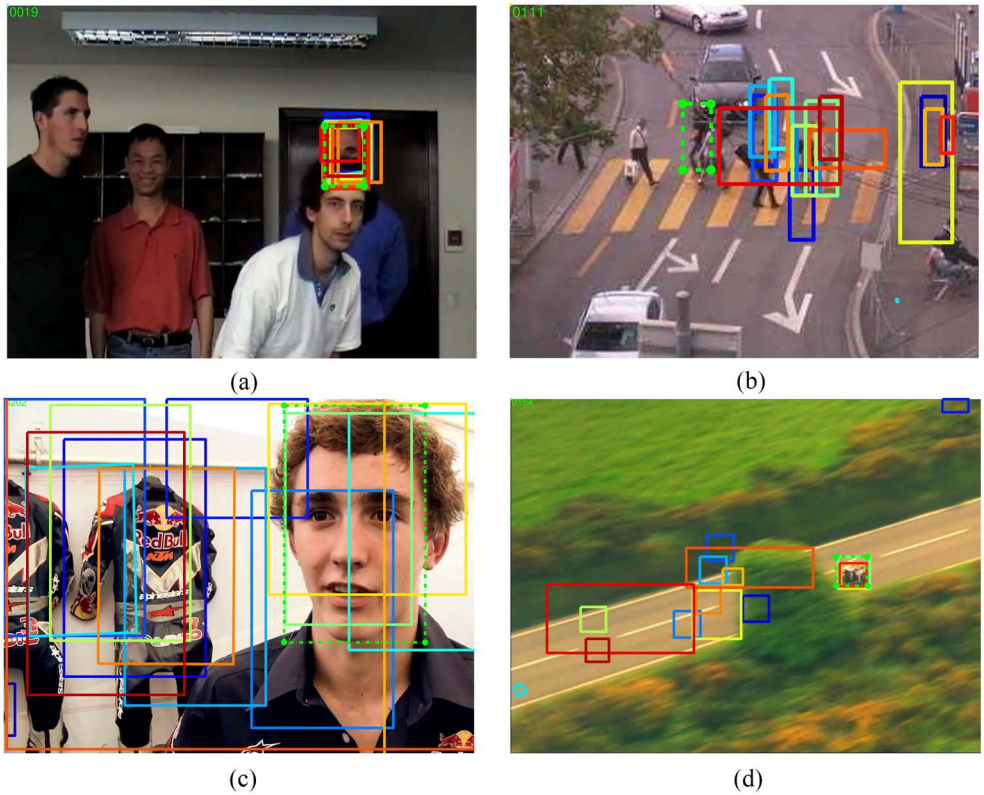

Fig. 13. Illustration of the effect of occlusion for (a) video 11-27 which is easy, and (b) 11-33 is hard. (c) 11-16 with a long intermediate shot, and (d) 11-07 with small and regular occlusions. For the color codes, see Fig. 10.

apart when the camera moves or zooms, as it has no extension of the model to handle that.

Due to the build-in capability for handling scale by warping, KLT performs well on zooming camera videos. Still, however, the performance of IVT, TST, L1O and TLD is superior in this regard, choosing the best fitting scale and rotation.

The methods TMC, TAG, and ACT perform poorly, supporting the necessity to evaluate trackers on data sets with largest possible variety. TMC is a complex method based on the star-connection between patches to allow for maximum shape flexibility. This capacity is both the strength and the weakness as it introduces a high dependency on the quality of the initial bounding box. Patches are sampled around

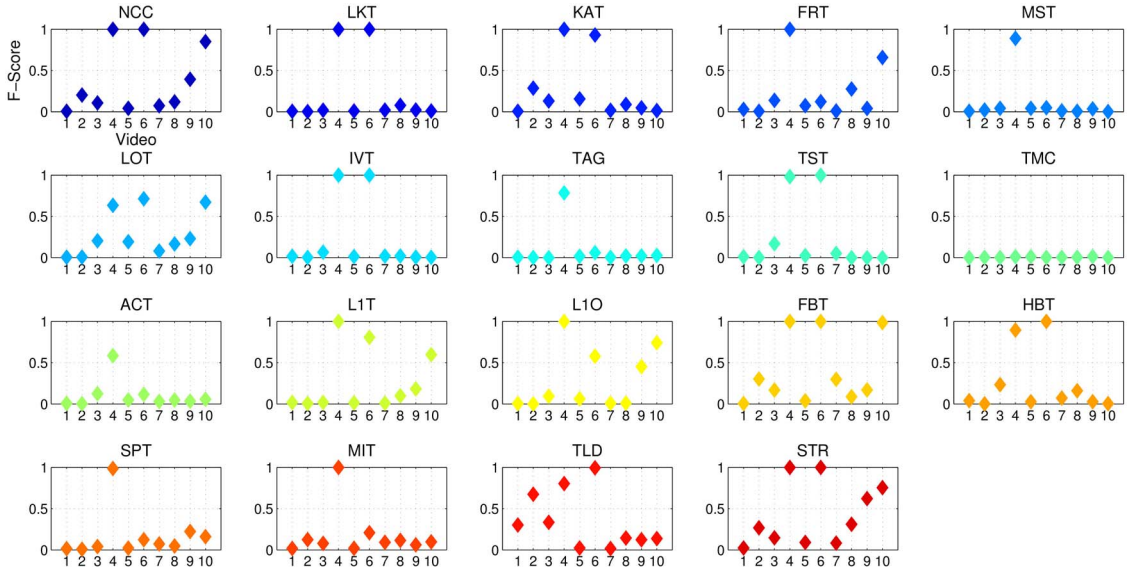

Fig. 14. Performance of the trackers on long videos. The x-axis shows the length of the video in number of frames, and the y-axis displays the F-score. For all subplots, from left to right: 1: back of the motor biker in the desert (947 frames), 2: car following car (3090), 3: car chase (2816), 4: rear mirror (2576), 5: soap (3881), 6: motor bike meter (3076), 7: shopping lane from the Tower dataset (2386), 8: surveillance with occluding pillars (1686), 9: surveillance with heavy shadow and reappearances (1241), 10: surveillance video (2156).

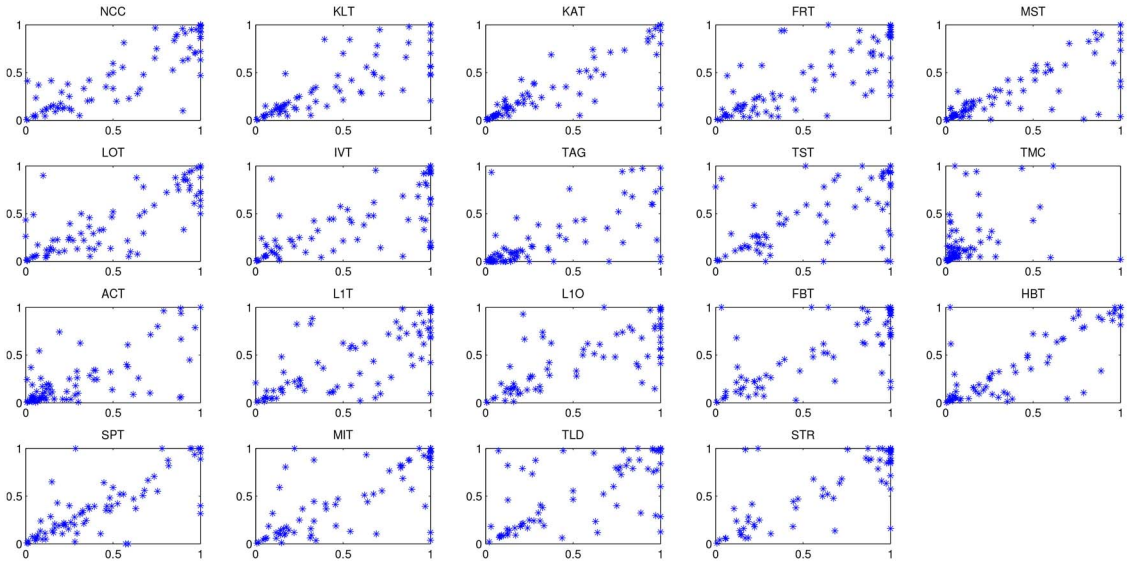

Fig. 15. F-score of the run with a properly initial bounding box on the x-axes, and of the 20%-shifted initial bounding box on the y-axes.

the object by empirical rules and a Hessian condition number, so that their amount depends on intensity variations inside the region. However, there is no guarantee that the samples cover the whole target. When the initial bounding box coincides only partially with the object region, the sampled set will include falsely labeled background elements. False labels will occur frequently, explaining the poor general performance as illustrated further in the bounding box shift experiment. With TAG designed to handle affine deformations, we have observed that the tracker performs worse when compared to trackers without that capacity. The tracker drifts away from the target in the first few frames in many sequences, especially when the camera moves and the tracker fails to estimate the transformation correctly. The appearance it learns from incremental PCA is noisy. The model is too complicated for many real-life circumstances with too many parameters to be estimated reliably. In ACT, many global parameters for appearance, shape and motion need to be estimated. Local patches are predicted by a constant-velocity Kalman filter. Then they are tuned according to the estimates. A small error in the global estimates can lead to a magnified error in the approximation of the target location. The improper estimate will gradually introduce a derailing of the tracking.

6.4.12 The Influence of the Accuracy of the Initial Bounding Box

The initial bounding box is an important element in tracking, the automatic specification of which has demonstrated a considerable progress recently, e.g., [114].

In a side-experiment we have investigated the stability of the result of the trackers on a 25% random selection of the dataset, for each of which videos the initial target bounding box was shifted by 20% of the width to the right so that the target was partially covered. The experiment gives an indication of trackers' robustness to misalignment. The results of the relative decrease in performance over the reduced dataset are given in Fig. 15.

We note an average loss in performance over all trackers of 4.8% in their overall F-score equal to a 10% relative loss in performance. This may be considered a moderate loss when compared to the F-score values implying that a disturbance of 20% is within reach of automated methods for localization [114]. However, the loss varies considerably among trackers. The most sensitive trackers are IVT and TAG with a loss in performance to an average F-score of 0.45 compared to 0.57 on the selection with the proper alignment at the start. This can be understood as IVT relies heavily on the appearance of the target and all variations therein, while TAG only differs from IVT in the motion model. MST goes down from 0.40 to 0.31 because it relies on a histogram, which in this experiment is heavily polluted at the start with background. At the positive end of the spectrum, STR drops from 0.69 to 0.67 due to the fact that the classifier does not make a difference on the basis of foreground and background at all. HBT drops from 0.48 to 0.46 by the back projection, effectively starting anew with the segmentation at every frame. The many trackers of TST also produce a similar drop. The strategies of STR, TST and HBT are robust to initial bounding box shift. An amazing result is scored by TMC which is weak in all aspect which improves considerably from 0.10 to 0.17.

The variation per video is much bigger than the averages show (see Fig. 15). Note the surprising amount of videos for which almost all trackers gain in performance.

7 Conclusion

7.1 The Circumstances of Tracking

The analysis and the experiments highlighted many circumstances which affect the performance of a tracker. Occlusion and clutter are well recognized for which several solutions demonstrate convincing results. Also confusion with similar objects in the scene is studied recently [89]. Of the nineteen trackers we have considered in this survey, eight have a mechanism for handling apparent changes in the scale of the object, whether it is due to a zoom in the camera or a change of the target's distance to the camera. In

the experiments we were able to demonstrate that inclusion of such a mechanism is advantageous for all these trackers, regardless their overall performance level. We have illustrated a large progress in handling the motion of the camera such as by TLD. Motion of the target, including the smoothness of the motion and its coherence, remain hard problems to solve, however. Changes in the appearance due to redressing of the target by rotation or otherwise, the presence of specularities and changes in illumination are a group of circumstances which are closely linked. Trackers based on discriminative methods are favorable here. Where size is rarely a design consideration in tracking, we have been able to demonstrate a dependence of the size of the target in many trackers. And finally, the length of the video is an important factor in the distinction of trackers testing their general ability to track and their model update mechanisms.

In this paper we have proposed the use of a wide variety of videos is important to obtain a good, differentiated impression of the performance of trackers in the many different circumstances. The current practice is that evidence of performance heavily relies on the choice of a few very long videos. No matter how impressive the performance may be, proof can only be anecdotal. We put forward that it is better to invest in many short videos supplemented with a few long ones, with a single defined target to understand better the trackers' behavior.

7.2 The Methods of Tracking

In this paper, trackers are divided according to their main method of tracking. From the papers it is not always clear how the proposed method fits in the history of ideas. As a consequence of the wide variety of circumstances and the hardness of the problem, the number of methods is very diverse. We have divided the methods in five main groups, see Fig. 5. In the first group are the trackers which optimize the direct match between a single model of the target and the incoming image on the basis of a representation of its appearance. The second group of trackers also seeks to maximize the match between the target and the image, but in this case the tracker holds more than one model of the target. This is an essential difference as the tracker is now capable of holding a long term memory of the target's appearances at the expense that the matching and the updating both have much more degrees of freedom (to go wrong). The third, recent, group of trackers performs a maximization of the match, but this time with explicit constraints derived from the motion, the coherence, the elasticity of the target and so on. An essentially different group of trackers do not perform matching on the basis of appearance per se, but rather maximization on the discrimination of the target from the background. The discriminative trackers learn a classifier to distinguish the object with several solutions for the problem how to label pixels as target pixels and background pixels after the first frame. Our final group has only one entry, STR, as it can be conceived as discriminative maximization with explicit constraints again. It is remarkable that the best performing trackers in this survey originate from all five groups. This demonstrates that they all solve some part of the (hard) problem of tracking. In this survey it has become evident from the large distance between the best trackers and the ideal combination of trackers in Fig. 7 that in many of the proposed methods there is value for some of the circumstances of tracking. But their ideal combination would solve a much larger part, as is demonstrated in the same figure showing the margin between the trackers and the best possible combination.

In the representation of the target a variety of representations are in use in our selection of trackers, as illustrated in Fig. 2. The bounding box is still widely in use, and part of the most successful trackers in this survey. While there is a general trend to give up an exact representation of the target and focus on parts and patches, such a distributed representation has not yet proven itself, at least not in our experiments. HBT projects the internal representation back into the image to arrive at a new target region. The inclusion of local background information in the representation of the target makes trackers robust to the illumination and appearance changes of the target as demonstrated in this survey.

In the representation of the appearance, so many different features are in use that we have preferred to group them in three fundamental categories: a two dimensional array, a one dimensional histogram and a feature vector. We have not been able to demonstrate the superiority of any of these representations. A deeper exploration of a possible inclusion of local features like SURF, HoG, and Daisy would be beneficial as they have played such a pivotal role in the area of object detection and classification. An indication is that FBT using these features performs best in the categories for illumination effects and surface change.

There are two major motion models. Uniform and dense sampling in a box around the current position of the target which we have set to the value of twenty pixels for all applicable trackers to permit a fair comparison, has no prejudice on the motion. Instead, the alternative Gaussian sampling which has been set to $\sigma = 11.55$ for all applicable trackers, corresponding to the size of the uniform sample area, favors the current position in the recorded frame. One consequence is that when there is no apparent motion of the projected target in the recorded frame, tracking is easy as shown in the experiments here. Even when there is complete short occlusion trackers are generally capable of following. In other words, videos showing static targets can be considered solved.

The updating of the model is a delicate topic quickly gaining importance in tracking. A general concern here is the complexity of the model update, and the potential infusion of the wrong information into the model. In general, the most complex update schemes - i.e. those with the greatest number of degree of freedom - perform the poorest in long videos. An example is the trackers based on incremental PCA holding extended memory representation of the past appearances of the target. At the other end of the scale, we have found most remarkably that NCC which is the oldest, the simplest model considered and performs no updating at all still performs 6th overall and retains that position in the ten long videos alone. We consider these facts as evidence for keeping the complexity of the update model low.

7.3 The Performance of Trackers

Overall STR performs the best. Furthermore, it has an even performance on all aspects of difficulty (see Fig. 9). Although it has a reasonable performance in many complex videos, it rarely achieves an outstanding performance in which it tracks significantly better than any of the other trackers. The solid performance is attributed to the precisely motivated use of S-SVM which optimizes the displacement directly in a discriminative setting. It only fails on scale changes of the target as it has no mechanism to detect that.

TLD performs remarkably well on camera motion due to its well-designed detection and motion model. The number of outstanding performances in occlusion and camera motion is much larger than any other tracker. Also the overall performance is good. Only in categories related to illumination and appearance the performance is limited.

FBT has good overall performance, while specializing in appearance changes of the target and changes in illumination. It does not know how to handle changes in shape or camera motions as it cannot handle the smearing of the feature values in its discriminative approach.

Good overall performance is also delivered by TST which is a collection of many weak trackers with a solid performance on all aspects, apart from long videos as the model updating has too big a complexity. L10 is the improved version of L1T designed to handle occlusions, but - as we have found - it improves also on other aspects of L1T.

In general, when trackers specialize on one specific aspect of tracking their performance goes down on others.

7.4 The Role of the Data and Evaluation

Given the large variety of data and trackers, we have been able to evaluate the effectiveness of evaluation metrics as proposed in literature. We have ensembled a large dataset for visual tracking evaluation, comprising of a variety of videos in nearly 90,000 individual frames, with every 5th frames annotated with ground truth. This represents an excellent resource for the evaluation of trackers and the evaluation of tracking metrics.

We conclude that the F-score and the OTA-score are highly correlated, while F and the ATA-score or the $F1$-score are still so much correlated that no additional information is to be expected from using both. Rather than using PBM, we prefer to use the F-score and the Deviation Metric as they measure different dimensions of performance.

The F-score permits comparison of trackers by survival curves and the associated Kaplan-Meier statistic. The volume of the data set makes it easy to demonstrate significant differences. Occasionally we have performed a t-test to demonstrate the superiority of a group of trackers on videos with a common property.

As we have nineteen trackers we are able to select videos of which one of the trackers delivers an outstanding performance as measured by Grubbs' outlier test. Survival and outliers test are valuable tools.

7.5 The Perspective of History

We have considered single object tracking, where the object is represented by a given bounding box. Recently, automatic object detection has become quite successful to the degree that tracking of a single object may be achieved by performing detection in every frame.

Several successful trackers evaluated in this paper, which have appeared during the last five years, perform tracking using discriminative classification. Widely used machine learning techniques like semi-supervised learning, multiple-instance learning and structured SVM have been successfully introduced in tracking. The key point to note here is that the background pixels surrounding the object bounding box do also play an important role in tracking.

Another trend that has influenced tracking is the locality. In previous generations of trackers the target was considered a single patch with global descriptors. Now the object is decomposed into segments, super-pixels, parts, patches and structures. Due to this the knowledge of local descriptors, (SIFT, SURF, LBP) and their robustness in matching and recognition, are infusing localized information into tracking. There has also been attempt to combine local and global information in two different layers in order to address rapid and significant appearance changes. Byproduct of decomposition of objects into super-pixels and parts is that the better segmentation and outline of the tracked object can also be achieved. In addition, local patches or segment can help to deal with occlusion.

During the last few years sparse representation or L-1 minimization has been employed in solving many computer vision problems including tracking. In order express the target in terms of its observations in the previous frames, sparse representation is an appropriate method capable of dealing with occlusion and some illumination variation.

New applications (such as sport analsis, concept-detection over time, social video classification, and crowd control) call for new methods, more and more robust to unconstraint circumstances including illumination changes, specularity, transparency, occlusion, clutter, low contrast, camera motion etc. No easy model is readily available, not even brute force. Influx of new methods from the computer vision field is still to be expected for these applications.

The conclusion we can offer is that tracking unknown objects in an unknown scenario is still an open problem as testified by the 50% error of all trackers in the experiments. We believe to have contributed to a valuable starting point for the second half.

7.6 The Difficulty of Tracking

One could wonder how it is possible that the tracker is capable of discriminating among sudden shadows, sudden appearance changes by rotation, sudden appearance of similar targets, wild unpredicted motion, long occlusion and all this after taking one sample. In this sense, tracking must be a hard topic, harder than object detection and classification where usually 50 well-chosen examples are used to train the detector and the classifier. In this sense many of the results are impressive, sometimes even approaching the level which is hard for humans.

One could wonder whether tracking may progress substantially. We have provided evidence that tracking may profit from making combinations of the existing successful models, the perfect combination being very distant from the current best performances. On the other hand, we have

seen evidence that simple models with a low complexity perform best. So the way to combine is not immediately obvious. In this sense, some of the videos deliver disappointing results for most of the trackers when compared to the level of human performance.

At the heart of the current state of the art, trackers which solve (label) drifting, small selections of large sets of features, and low-complexity update mechanisms without losing too much on the current achievements will progress the field.

7.7 The Limitations of This Survey

Most Computer Vision problems are the mirror of the collective scientific knowledge, often reinforced by the inspiration from some killer application. As we have seen in this survey, methods specializing in one type of target have a vast advantage as it limits the target appearance variation. Surveillance is such an application.

We have aimed for generic tracking where the target is unknown. This joint effort has collected the evaluation of nineteen publicly accessible trackers on 315 video sequences in an experimental survey we believe to be the first of its scale. As tens of trackers are appearing each year, we are aware that the survey will never be complete. Tens of compute years have gone in the survey, possible only with a parallel exploitation of clusters in three labs, as well as a considerable programming and annotation effort. We have aimed to present new objective methods for evaluating trackers more than being complete. The dataset, ground truth, trackers implementations and tools to compute the evaluation metrics are published on http://www.alov300.org/ and they are available for download. The website also supports visualization of the tracker results.

ACKNOWLEDGMENTS

The work in this paper was funded by COMMIT, the National Dutch Program for public private ICT research in the Netherlands, by EU FESR 2008 15 from the region of Emilia Romagna Italy, and by the U.S. Army Research Laboratory and the U.S. Army Research Office under grant W911NF-09-1-0255. We are grateful to the reviewers and editor for their careful and close reading of the manuscript.

REFERENCES

[1] Z. Kalal, J. Matas, and K. Mikolajczyk, "P-N learning: Bootstrapping binary classifiers by structural constraints," in *Proc. IEEE CVPR*, San Francisco, CA, USA, 2010.
[2] W. Hu, X. Zhou, W. Li, W. Luo, X. Zhang, and S. Maybank, "Active contour-based visual tracking by integrating colors, shapes, and motions," *IEEE Trans. Image Process.*, vol. 22, no. 5, pp. 1778–1792, May. 2013.
[3] X. Gao, Y. Su, X. Li, and D. Tao, "A review of active appearance models," *IEEE Trans. Syst., Man, Cybern. C,*, vol. 40, no. 2, pp. 145–158, 2010.
[4] U. Prabhu, K. Seshadri, and M. Savvides, "Automatic facial landmark tracking in video sequences using kalman filter assisted active shape models," in *Proc. ECCV*, Heraklion, Greece, 2010.
[5] J. Berclaz, F. Fleuret, E. Turetken, and P. Fua, "Multiple object tracking using k-shortest paths optimization," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 9, pp. 1806–1819, Sept. 2011.
[6] J. Henriques, R. Caseiro, and J. Batista, "Globally optimal solution to multi-object tracking with merged measurements," in *Proc. ICCV*, Barcelona, Spain, 2001.
[7] A. R. Zamir, A. Dehghan, and M. Shah, "GMCP-tracker: Global multi-object tracking using generalized minimum clique graphs," in *Proc. 12th ECCV*, Florence, Italy, 2012.
[8] S. Pellegrini, A. Ess, and L. van Gool, "Improving data association by joint modeling of pedestrian trajectories and groupings," in *Proc. 11th ECCV*, Heraklion, Greece, 2010.
[9] L. Zhang, Y. Li, and R. Nevatia, "Global data association for multi-object tracking using network flows," in *Proc. IEEE CVPR*, Anchorage, AK, USA, 2008.
[10] A. Yilmaz, O. Javed, and M. Shah, "Object tracking: A survey," *ACM CSUR*, vol. 38, no. 4, Article 13, 2006.
[11] T. F. Chan and L. A. Vese, "Active contours without edges," *IEEE Trans. Image Process.*, vol. 10, no. 2, pp. 266–277, Feb. 2001.
[12] Y. Li, H. Ai, T. Yamashita, S. Lao, and M. Kawade, "Tracking in low frame rate video: A cascade particle filter with discriminative observers of different life spans," in *Proc. IEEE CVPR*, Minneapolis, MN, USA, 2007.
[13] J. Kwon and K. Lee, "Tracking of abrupt motion using Wang-Landau Monte Carlo estimation," in *Proc. 10th ECCV*, Marseille, France, 2008.
[14] W. C. Siew, K. P. Seng, and L. M. Ang, "Lips contour detection and tracking using watershed region-based active contour model and modified," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 22, no. 6, pp. 869–874, Jun. 2012.
[15] B. Ristic and M. L. Hernandez, "Tracking systems," in *Proc. IEEE RADAR*, Rome, Italy, 2008, pp. 1–2.
[16] J. Fiscus, J. Garofolo, T. Rose, and M. Michel, "AVSS multiple camera person tracking challenge evaluation overview," in *Proc. 6th IEEE AVSS*, Genova, Italy, 2009.
[17] C. B. Santiago, A. Sousa, M. L. Estriga, L. P. Reis, and M. Lames, "Survey on team tracking techniques applied to sports," in *Proc. AIS*, Povoa de Varzim, Portugal, 2010, pp. 1–6.
[18] J. C. McCall and M. M. Trivedi, "Video-based lane estimation and tracking for driver assistance: Survey, system, and evaluation," *IEEE Trans. Intell. Transp. Syst.*, vol. 7, no. 1, pp. 20–37, Mar. 2006.
[19] S. Y. Chen, "Kalman filter for robot vision: A survey," *IEEE Trans. Ind. Electron.*, vol. 59, no. 11, pp. 4409–4420, Nov. 2012.
[20] T. B. Moeslund, A. Hilton, and V. Krüger, "A survey of advances in vision-based human motion capture and analysis," *CVIU*, vol. 104, no. 2–3, pp. 90–126, 2006.
[21] R. Poppe, "Vision-based human motion analysis: An overview," *CVIU*, vol. 108, no. 1–2, pp. 4–18, 2007.
[22] Z. Jia, A. Balasuriya, and S. Challa, "Recent developments in vision based target tracking for autonomous vehicles navigation," in *Proc. IEEE ITSC*, Toronto, ON Canada, 2006, pp. 765–770.
[23] O. Demigha, W. Hidouci, and T. Ahmed, "On energy efficiency in collaborative target tracking in wireless sensor network: A review," *IEEE Commun. Surv. Tuts.*, vol. 15, no. 99, pp. 1–13, 2012.
[24] J. Popoola and A. Amer, "Performance evaluation for tracking algorithms using object labels," in *Proc. USA ICASSP*, Las Vegas, NV, USA, 2008.
[25] D. A. Klein, D. Schulz, S. Frintrop, and A. B. Cremers, "Adaptive real-time video-tracking for arbitrary objects," in *Proc. IEEE IROS*, Taipei, Taiwan, 2010, pp. 772–777.
[26] [Online]. Available: /home/skynet/a/sig/kng/dataset/CAVIAR
[27] A. Nilski, "An evaluation metric for multiple camera tracking systems: The i-LIDS 5th scenario," in *Proc. SPIE*, Cardiff, Wales, 2008.
[28] D. Baltieri, R. Vezzani, and R. Cucchiara, "3DPes: 3D people dataset for surveillance and forensics," in *Proc. Int. ACM Workshop MA3HO*, Scottsdale, AZ, USA, 2011, pp. 59–64.
[29] J. Ferryman and J. L. Crowley, *Proc. IEEE Int. Workshop PETS*, Boston, MA, USA, Aug. 2010.
[30] C.-H. Kuo, C. Huang, and R. Nevatia, "Multi-target tracking by on-line learned discriminative appearance models," in *Proc. IEEE CVPR*, San Francisco, CA, USA, 2010, pp. 685–692.
[31] B. Karasulu and S. Korukoglu, "A software for performance evaluation and comparison of people detection and tracking methods in video processing," *MTA*, vol. 55, no. 3, pp. 677–723, 2011.
[32] D. M. Chu and A. W. M. Smeulders, "Thirteen hard cases in visual tracking," in *Proc. IEEE Int. Workshop PETS*, 2010.

[33] C. Erdem, B. Sankur, and A. M. Tekalp, "Performance measures for video object segmentation and tracking," *IEEE Trans. Image Process.*, vol. 13, no. 7, pp. 937–951, Jul. 2004.

[34] S. Salti, A. Cavallaro, and L. di Stefano, "Adaptive appearance modeling for video tracking: Survey and evaluation," *IEEE Trans. Image Process.*, vol. 21, no. 10, pp. 4334–4348, Oct. 2012.

[35] J. C. SanMiguel, A. Cavallaro, and J. M. Martinez, "Adaptive on-line performance evaluation of video trackers," *IEEE Trans. Image Process.*, vol. 21, no. 5, pp. 1828–1837, May 2012.

[36] A. T. Nghiem, F. Bremond, M. Thonnat, and V. Valentin, "Etiseo, performance evaluation for video surveillance systems," in *Proc. AVSS*, London, U.K., 2007, pp. 476–481.

[37] P. Carvalho, J. S. Cardoso, and L. Corte-Real, "Filling the gap in quality assessment of video object tracking," *IVC*, vol. 30, no. 9, pp. 630–640, 2012.

[38] F. Bashir and F. Porikli, "Performance evaluation of object detection and tracking systems," in *Proc. IEEE Int. Workshop PETS*, 2006.

[39] R. Kasturi *et al.*, "Framework for performance evaluation of face, text, and vehicle detection and tracking in video: Data, metrics, and protocol," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 31, no. 2, pp. 319–336, Feb. 2009.

[40] K. Bernardin and R. Stiefelhagen, "Evaluating multiple object tracking performance: The clear MOT metrics," *EURASIP J. IVP*, vol. 2008, no. 1, p. 246309, Feb. 2008.

[41] M. Everingham, L. J. V. Gool, C. K. I. Williams, J. M. Winn, and A. Zisserman, "The Pascal visual object classes VOC challenge," *IJCV*, vol. 88, no. 2, pp. 303–338, 2010.

[42] D. L. Shaul Oron, Aharon Bar-Hillel, and S. Avidan, "Locally orderless tracking," in *Proc. IEEE CVPR*, Providence, RI, USA, 2012.

[43] J. Kwon and K. M. Lee, "Tracking of a non-rigid object via patch-based dynamic appearance modeling and adaptive basin hopping monte carlo sampling," in *Proc. IEEE CVPR*, Miami, FL, USA, 2009.

[44] E. Maggio and A. Cavallaro, *Video Tracking: Theory and Practice*. 1st ed. Oxford, U.K.: Wiley, 2011.

[45] E. Maggio and A. Cavallaro, "Tracking by sampling trackers," in *Proc. IEEE ICCV*, Barcelona, Spain, 2011, pp. 1195–1202.

[46] B. Babenko, M.-H. Yang, and S. Belongie, "Visual tracking with online multiple instance learning," in *Proc. IEEE CVPR*, Miami, FL, USA, 2009.

[47] A. Sanin, C. Sanderson, and B. C. Lovell, "Shadow detection: A survey and comparative evaluation of recent methods," *PR*, vol. 45, no. 4, pp. 1684–1695, 2012.

[48] A. Amato, M. G. Mozerov, A. D. Bagdanov, and J. Gonzalez, "Accurate moving cast shadow suppression based on local color constancy detection," *IEEE Trans. Image Process.*, vol. 20, no. 10, pp. 2954–2966, Oct. 2011.

[49] K. Briechle and U. D. Hanebeck, "Template matching using fast normalized cross correlation," in *Proc. SPIE*, vol. 4387. 2001, pp. 95–102.

[50] S. Baker and I. Matthews, "Lucas-Kanade 20 years on: A unifying framework," *IJCV*, vol. 56, no. 3, pp. 221–255, 2004.

[51] H. T. Nguyen and A. W. M. Smeulders, "Fast occluded object tracking by a robust appearance filter," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 26, no. 8, pp. 1099–1104, Aug. 2004.

[52] A. Adam, E. Rivlin, and I. Shimshoni, "Robust fragments-based tracking using the integral histogram," in *Proc. IEEE CVPR*, Washington, DC, USA, 2006.

[53] D. Comaniciu, V. Ramesh, and P. Meer, "Real-time tracking of non-rigid objects using mean shift," in *Proc. IEEE CVPR*, Hilton Head Island, SC, USA, 2000.

[54] D. A. Ross, J. Lim, and R. S. Lin, "Incremental learning for robust visual tracking," *IJCV*, vol. 77, no. 1–3, pp. 125–141, 2008.

[55] M. Isard and A. Blake, "A mixed-state condensation tracker with automatic model-switching," in *Proc. 6th ICCV*, Bombay, India, 1998.

[56] J. Kwon and F. C. Park, "Visual tracking via geometric particle filtering on the affine group with optimal importance functions," in *Proc. IEEE CVPR*, Miami, FL, USA, 2009.

[57] L. Čehovin, M. Kristan, and A. Leonardis, "An adaptive coupled-layer visual model for robust visual tracking," in *Proc. IEEE ICCV*, Barcelona, Spain, 2011.

[58] X. Mei and H. Ling, "Robust visual tracking using L1 minimization," in *Proc. IEEE 12th ICCV*, Kyoto, Japan, 2009.

[59] X. Mei, H. Ling, Y. Wu, E. Blasch, and L. Bai, "Minimum error bounded efficient l1 tracker with occlusion detection," in *Proc. IEEE CVPR*, Providence, RI, USA, 2011.

[60] H. T. Nguyen and A. W. M. Smeulders, "Robust track using foreground-background texture discrimination," *IJCV*, vol. 68, no. 3, pp. 277–294, 2006.

[61] D. M. Chu and A. W. M. Smeulders, "Color invariant surf in discriminative object tracking," in *Proc. IEEE ECCV*, Heraklion, Greece, 2010.

[62] M. Godec, P. M. Roth, and H. Bischof, "Hough-based tracking of non-rigid objects," in *Proc. IEEE ICCV*, Barcelona, Spain, 2011.

[63] L. Breiman, "Random forests," *ML*, vol. 45, no. 1, pp. 5–32, 2001.

[64] D. H. Ballard, "Generalizing the hough transform to detect arbitrary shapes," *PR*, vol. 13, no. 2, pp. 111–122, 1981.

[65] C. Rother, V. Kolmogorov, and A. Blake, ""GrabCut": Interactive foreground extraction using iterated graph cuts," *ACM Trans. Graphics*, vol. 23, no. 3, pp. 309–314, Aug. 2004.

[66] S. Wang, H. Lu, F. Yang, and M.-H. Yang, "Superpixel tracking," in *Proc. IEEE ICCV*, Barcelona, Spain, 2011.

[67] T. G. Dietterich, R. H. Lathrop, and T. Lozano-Pérez, "Solving the multiple instance problem with axis-parallel rectangles," *AI*, vol. 89, no. 1–2, pp. 31–71, 1997.

[68] Z. Kalal, J. Matas, and K. Mikolajczyk, "Online learning of robust object detectors during unstable tracking," in *Proc. IEEE 12th ICCV*, Kyoto, Japan, 2009.

[69] M. Ozuysal, P. Fua, and V. Lepetit, "Fast keypoint recognition in ten lines of code," in *Proc. IEEE CVPR*, Minneapolis, MN, USA, 2007, pp. 1–8.

[70] S. Hare, A. Saffari, and P. H. S. Torr, "Struck: Structured output tracking with kernels," in *Proc. IEEE ICCV*, Barcelona, Spain, 2011.

[71] J. R. R. Uijlings, A. W. M. Smeulders, and R. J. H. Scha, "What is the spatial extent of an object?" in *Proc. IEEE CVPR*, Miami, FL, USA, 2009.

[72] D. Terzopoulos and R. Szeliski, "Tracking with Kalman snakes," MIT Press, 1992.

[73] H. T. Nguyen, M. Worring, R. van den Boomgaard, and A. W. M. Smeulders, "Tracking nonparameterized object contours in video," *IEEE Trans. Image Process.*, vol. 11, no. 9, pp. 1081–1091, Sept. 2002.

[74] X. Zhou, W. Hu, Y. Chen, and W. Hu, "Markov random field modeled level sets method for object tracking with moving cameras," in *Asian Conference on Computer Vision*, Y. Yagi, S. Kang, I. Kweon, and H. Zha, Eds. Berlin, Germany: Springer, 2007, pp. 832–842, LNCS 4843.

[75] A. Senior, "Tracking people with appearance models," in *Proc. Int. Workshop PETS*, 2002.

[76] R. Vezzani, C. Grana, and R. Cucchiara, "Probabilistic people tracking with appearance models and occlusion classification: The ad-hoc system," *PRL*, vol. 32, no. 6, pp. 867–877, 2011.

[77] S. Calderara, R. Cucchiara, and A. Prati, "Bayesian-competitive consistent labeling for people surveillance," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 30, no. 2, pp. 354–360, Feb. 2008.

[78] G. Shu, A. Dehghan, O. Oreifej, E. Hand, and M. Shah, "Part-based multiple-person tracking with partial occlusion handling," in *Proc. IEEE CVPR*, Providence, RI, USA, 2012.

[79] D. Ramanan, D. A. Forsyth, and K. Barnard, "Building models of animals from video," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 28, no. 8, pp. 1319–1334, Aug. 2006.

[80] J. M. Geusebroek, R. van den Boomgaard, A. W. M. Smeulders, and H. Geerts, "Color invariance," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 23, no. 12, pp. 1338–1350, Dec. 2001.

[81] J. M. Geusebroek, A. W. M. Smeulders, and J. J. van de Weijer, "Fast anisotropic gauss filtering," *IEEE Trans. Image Process.*, vol. 12, no. 8, pp. 938–943, Aug. 2003.

[82] H. Bay, A. Ess, T. Tuytelaars, and L. van Gool, "Speeded-up robust features (SURF)," *CVIU*, vol. 110, no. 3, pp. 346–359, 2008.

[83] C. R. Wren, A. Azarbayejani, T. Darrell, and A. P. Pentland, "Pfinder: Real-time tracking of the human body," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 19, no. 7, pp. 780–785, Jul. 1997.

[84] D. Koller *et al.*, "Towards robust automatic traffic scene analysis in real-time," in *Proc. IEEE ICPR*, Jerusalem, Israel, 1994.

[85] C. Stauffer and W. E. L. Grimson, "Adaptive background mixture models for real-time tracking," in *Proc. IEEE CVPR*, Fort Collins, CO, USA, 1999.

[86] P. W. Power and J. A. Schoonees, "Understanding background mixture models for foreground segmentation," in *Proc. IVCNZ*, 2002.
[87] R. Cucchiara, C. Grana, M. Piccardi, and A. Prati, "Detecting moving objects, ghosts, and shadows in video streams," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 25, no. 10, pp. 1337–1342, Oct. 2003.
[88] A. Humayun, O. Mac Aodha, and G. J. Brostow, "Learning to find occlusion regions," in *Proc. IEEE CVPR*, Providence, RI, USA, 2011.
[89] T. B. Dinh, N. Vo, and G. Medioni, "Context tracker: Exploring supporters and distracters in unconstrained environments," in *Proc. IEEE CVPR*, 2011.
[90] S. Lin, Y. Li, S. Kang, X. Tong, and H.-Y. Shum, "Diffuse-specular separation and depth recovery from image sequences," in *Proc. ECCV*, London, U.K., 2002.
[91] Z. Qigui and L. Bo, "Search on automatic target tracking based on PTZ system," in *Proc. IEEE IASP*, Hubei, China, 2011, pp. 192–195.
[92] R. E. Kalman, "A new approach to linear filtering and prediction problem," *J. Basic Eng.*, vol. 82, no. 1, pp. 34–45, 1960.
[93] G. Welch and G. Bishop, "An introduction to the Kalman filter," Univ. North Carolina, Chapel Hill, NC, USA, Lecture, 2001.
[94] B. Ristic, S. Arulampalam, and N. Gordon, *Beyond the Kalman Filter: Particle Filters for Tracking Applications*. Artech House, 2003.
[95] S. Ali and M. Shah, "Floor fields for tracking in high density crowd scenes," in *Proc. 10th ECCV*, Marseille, France, 2008.
[96] M. Rodriguez, S. Ali, and T. Kanade, "Tracking in unstructured crowded scenes," in *Proc. IEEE 12th ICCV*, Kyoto, Japan, 2009.
[97] X. Song, X. Shao, H. Zhao, J. Cui, R. Shibasaki, and H. Zha, "An online approach: Learning-semantic-scene-by-tracking and tracking-by-learning-semantic-scene," in *Proc. IEEE CVPR*, San Francisco, CA, USA, 2010.
[98] D. Baltieri, R. Vezzani, and R. Cucchiara, "People orientation recognition by mixtures of wrapped distributions on random trees," in *Proc. 12th ECCV*, Florence, Italy, 2012.
[99] "Multiple-shot person re-identification by chromatic and epitomic analyses," *PRL*, vol. 33, no. 7, pp. 898–903, 2012.
[100] D. Coppi, S. Calderara, and R. Cucchiara, "Appearance tracking by transduction in surveillance scenarios," in *Proc. 8th IEEE AVSS*, Klagenfurt, Austria, 2011.
[101] B. D. Lucas and T. Kanade, "An iterative image registration technique with an application to stereo vision," in *Proc. IJCAI*, vol. 3. Vancouver, BC, Canada, 1981, pp. 674–679.
[102] F. Porikli, O. Tuzel, and P. Meer, "Covariance tracking using model update based on Lie algebra," in *Proc. IEEE CVPR*, Washington, DC, USA, 2006.
[103] Y. Wu, J. Cheng, J. Wang, and H. Lu, "Real-time visual tracking via incremental covariance tensor learning," in *Proc. IEEE 12th ICCV*, Kyoto, Japan, 2009.
[104] S. Avidan, "Support vector tracking," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 26, no. 8, pp. 1064–1072, Aug. 2004.
[105] E. Maggio, E. Piccardo, C. Regazzoni, and A. Cavallaro, "Particle PHD filtering for multi-target visual tracking," in *Proc. ICASSP*, vol. 1. Honolulu, HI, USA, 2007, pp. 1101–1104.
[106] A. Ellis, A. Shahrokni, and J. Ferryman, "Overall evaluation of the PETS 2009 results," in *Proc. IEEE Int. Workshop PETS*, vol. 7. 2009.
[107] B. Benfold and I. Reid, "Stable multi-target tracking in real-time surveillance video," in *Proc. IEEE CVPR*, Providence, RI, USA, 2011.
[108] J. F. Lawless, *Statistical Models and Methods for Lifetime Data*. Hoboken, NJ, USA: Wiley, 2003.
[109] D. Collett, *Modelling Survial Data in Medical Research*. Boca Raton, FL, USA: Chapman Hall, 2003.
[110] E. L. Kaplan and P. Meier, "Nonparametric estimation from incomplete observations," *J. Amer. Statist. Assoc.*, vol. 53, no. 282, pp. 457–481, 1958.
[111] N. Mantel, "Evaluation of survival data and two new rank order statistics arising in its consideration," *Cancer Chemother. Rep.*, vol. 50, no. 3, pp. 163-170, Mar. 1966.
[112] D. G. Kleinbaum and M. Klein, "Kaplan-Meier survival curves and the log-rank test," in *Survival Analysis*. New York, NY, USA: Springer, 2012, pp. 55–96.
[113] F. E. Grubbs, "Procedures for detecting outlying observations in samples," *Technometrics*, vol. 11, no. 1, pp. 1–21, 1969.
[114] K. E. A. van de Sande, J. R. R. Uijlings, T. Gevers, and A. W. M. Smeulders, "Segmentation as selective search for object recognition," in *Proc. ICCV*, Barcelona, Spain, 2011.

Arnold W. M. Smeulders graduated from the Delft Technical University, Delft, The Netherlands, in physics on texture in medical images. He received the Ph.D. degree from Leiden Medical Faculty, Leiden, The Netherlands. Since 1994, he is a Professor at the University of Amsterdam, Amsterdam, The Netherlands, leading The ISIS Group on Visual Search Engines. The search engine has been a top three performer in the TRECvid competition for the last 8 years. ISIS came out best in the 6-yearly international review in 2003 and 2009 (shared the maximum with few). In 2010, he co-founded Euvision, a university spin-off. In addition, he is recently a Director of COMMIT, the large public private-ICT-Research Program of The Netherlands. He is the past Associate Editor of the *IEEE Transactions on Pattern Analysis and Machine Intelligence* and current Associate Editor of *IJCV*. He is an IAPR fellow and a Honorary member of NVPHBV. He was a Visiting Professor in Hong Kong, Tsukuba, Modena, Cagliari, and Orlando. He is a recent member of the Academia Europaea.

Dung M. Chu received the master's degree in computer science from the University of Amsterdam, Amsterdam, The Netherlands, in 2008. He is pursuing the Ph.D. degree with the Intelligent Systems Lab Amsterdam, University of Amsterdam. His current research interests include video understanding, object recognition, and object tracking.

Rita Cucchiara received the laurea degree in electronic engineering and the Ph.D. degree in computer engineering from the University of Bologna, Bologna, Italy, in 1989 and 1992, respectively. Since 2005, she is a Full Professor at the University of Modena and Reggio Emilia, Italy. She heads the Imagelab Laboratory (http://imagelab.ing.unimore.it) and is the Director of the Research Center in ICT SOFTECH-ICT (www.softech.unimore.it). Since 2011, she is the scientific responsible of the ICT platform of the High Tecnology TV Network of Emilia Romagna region. Her current research interests include pattern recognition, computer vision, and multimedia. She has been a coordinator of several projects in video surveillance mainly for people detection, tracking, and behavior analysis. In the field of multimedia, she works on annotation, retrieval, and human-centered searching in images and video big data for cultural heritage. She is the author of over 200 papers in journals and international proceedings, and is a reviewer for several international journals. Since 2006, she is a fellow of ICPR. Prof. Cucchiara is a member of the Editorial Boards of *Multimedia Tools and Applications and Machine Vision and Applications* journals and Chair of several workshops and conferences. She has been the General Chair of the 14th International Conference on ICIAP in 2007 and the 11th National Congress AI*IA in 2009. She is a member of the IEEE Computer Society, ACM, GIRPR, and AI*IA. Since 2006, she has been a fellow of IAPR.

Simone Calderara received the M.S.I. degree in computer science from the University of Modena and Reggio Emilia, Reggio Emilia, Italy, in 2004, and the Ph.D. degree in computer engineering and science in 2009. He is currently a Assistant Professor with the Imagelab Computer Vision and Pattern Recognition Laboratory, University of Modena and Reggio Emilia and SOFTECH-ICT Research center. His current research interests include computer vision and machine learning applied to human behavior analysis, video surveillance and tracking in crowd scenarios, and time series analysis for forensic applications. He was in national and international projects on video surveillance and behavior analysis as a Project Leader. He was the Program Chair of the International Workshop on Pattern Recognition for Crowd analysis and a member of the Program Committee of international conferences AVSS, ICDP, ACM Multimedia and active reviewer of several *IEEE Transaction* journals. He is the co-author of over 50 publications in journals and international conferences.

Afshin Dehghan received the B.S. degree in electrical engineering from the University of Tehran, Tehran, Iran, in 2011. He is currently pursuing the Ph.D. degree at UCF's Center for Research in Computer Vision (CRCV). He has authored several papers published in conference such as CVPR and ECCV. His current research interests include object tracking, object detection, event recognition, and face verification.

Mubarak Shah is a Agere Chair Professor of Computer Science, and is the Founding Director of the Computer Vision Lab at the University of Central Florida, Orlando, FL, USA. He is a co-author of three books and has published extensively on topics related to visual surveillance, tracking, human activity and action recognition, object detection and categorization, shape from shading, geo registration, and visual crowd analysis. Dr. Shah is a fellow of IEEE, IAPR, AAAS, and SPIE. He is an ACM Distinguished Speaker. He was an IEEE Distinguished Visitor speaker from 1997 to 2000, and received IEEE Outstanding Engineering Educator Award in 1997. He is an Editor of international book series on Video Computing; Editor-in-Chief of *Machine Vision and Applications* journal, and an Associate Editor of *ACM Computing Surveys* journal. He was an Associate Editor of the *IEEE Transactions on Pattern Analysis and Machine Intelligence* and the Program Co-Chair of *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2008.