IJESRT

INTERNATIONAL JOURNAL OF ENGINEERING SCIENCES & RESEARCH TECHNOLOGY

# A Survey on Modern Era's Online Object Tracking Algorithms

**Khemraj Deshmukh[*1], Vishal Moyal[2]**
info2khemraj@gmail.com

## Abstract

Object tracking finds many practical applications ranging from robotics, surveillance, augmented reality to human-computer interaction, the state-of-the- art is still far from achieving results comparable to human performance. The goal of this article is to review the state-of-the-art tracking methods. Object tracking remains a challenging problem due to appearance change caused by pose, illumination, occlusion, and motion, among others. An effective appearance model is of prime importance for the success of a tracking algorithm that has been attracting much attention in recent years. In this survey, we empirically demonstrate the performance of the algorithm against various common failure modes in the generic object tracking problem.

## Introduction

Object tracking is a well studied problem in computer vision and has many practical applications. The problem and its difficulty depend on several factors, such as the amount of prior knowledge about the target object and the number and type of parameters being tracked (e.g. location, scale, detailed contour). Although there has been some success with building trackers for specific object classes (e.g. faces [1], humans [2], mice [3], rigid objects [4]), tracking generic objects has remained challenging because an object can drastically change appearance when deforming, rotating out of plane, or when the illumination of the scene changes.

A typical tracking system consists of three components: (1) an appearance model, which can evaluate the likelihood that the object of interest is at some particular location; (2) a motion model, which relates the locations of the object over time; and (3) a search strategy for finding the most likely location in the current frame. Although many tracking methods employ static appearance models that are either defined manually or trained using only the first frame [2], [4], [6], [7], [8], [9], these methods are often unable to cope with significant appearance changes. These challenges are particularly difficult when there is limited a priori knowledge about the object of interest. In this scenario, it has been shown that an adaptive appearance model, which evolves during the tracking process as the appearance of the object changes, is the key to good performance [10], [11], [12]. Training adaptive appearance models, however, is itself a difficult task with many questions yet to be answered. Such models often involve many parameters that must be tuned to get good performance (e.g. "forgetting factors" that control how fast the appearance model can change), and can suffer from drift problems when an object undergoes partial occlusion. Tracking algorithms can be generally categorized as either generative [1, 2, 6, 10, 9] or discriminative [3– 5, 7, 8] based on their appearance models.

Generative tracking algorithms typically learn a model to represent the target object and then use it to search for the image region with minimal reconstruction error. Black et al. [1] learn an off-line subspace model to represent the object of interest for tracking. The IVT method [6] utilizes an incremental subspace model to adapt appearance changes. Recently, sparse representation has been used in the 1-tracker where an object is modeled by a sparse linear combination of target and trivial templates [10]. However, the computational complexity of this tracker is rather high, thereby limiting its applications in real-time scenarios. Li et al. [9] further extend the 1-tracker by using the orthogonal matching pursuit algorithm for solving the optimization problems efficiently. Despite much demonstrated success of these online generative tracking algorithms, several problems remain to be solved. First, numerous training samples cropped from consecutive frames are required in order to learn an appearance model online. Since there are only a few samples at the outset, most tracking algorithms often assume that the target appearance does not change much during this period. However, if the appearance of the target changes significantly at the beginning, the drift problem is likely to occur. Second, when multiple samples are drawn at the current target location, it is likely to cause drift as the appearance model needs to adapt to these potentially mis-aligned examples [8]. Third, these generative algorithms do not use the

background information which is likely to improve tracking stability and accuracy.

Discriminative algorithms pose the tracking problem as a binary classification task in order to find the decision boundary for separating the target object from the background. Avidan [3] extends the optical flow approach with a support vector machine classifier for object tracking. Collins et al. [4] demonstrate that the most discriminative features can be learned online to separate the target object from the background. Grabner et al. [5] propose an online boosting algorithm to select features for tracking. However, these trackers [3–5] only use one positive sample (i.e., the current tracker location) and a few negative samples when updating the classifier. As the appearance model is updated with noisy and potentially misaligned examples, this often leads to the tracking drift problem. Grabner et al. [7] propose an online semi-supervised boosting method to alleviate the drift problem in which only the samples in the first frame are labeled and all the other samples are unlabeled. Babenko et al. [8] introduce multiple instance learning into online tracking where samples are considered within positive and negative bags or sets. Recently, a semi-supervised learning approach [11] is developed in which positive and negative samples are selected via an online classifier with structural constraints.

## Adaptive Appearance Models

An important choice in the design of appearance models is whether to model only the object [12], [23], or both the object and the background [24], [25], [26], [27], [28], [29], [30]. Many of the latter approaches have shown that training a model to separate the object from the background via a discriminative classifier can often achieve superior results. These methods are closely related to object detection – an area that has seen great progress in the last decade. In fact, some of these methods are referred to as "tracking-by-detection" or "tracking by repeated recognition" [31]. In particular, the recent advances in face detection [32] have inspired some successful realtime tracking algorithms [25], [26]. A major challenge that is often not discussed in the literature is how to choose positive and negative examples when updating the adaptive appearance model. Most commonly this is done by taking the current tracker location as one positive example, and sampling the neighborhood around the tracker location for negatives. If the tracker location is not precise, however, the appearance model ends up getting updated with a suboptimal positive example. Over time this can degrade the model, and can cause drift. On the other hand, if multiple positive examples are used (taken from a small neighborhood around the current tracker location), the model can become confused and its discriminative power can suffer. Alternatively, [33] recently proposed a semi-supervised approach where labeled examples come from the first frame only, and subsequent training examples are left unlabeled. This method is particularly well suited for scenarios where the object leaves the field of view completely, but it throws away a lot of useful information by not taking advantage of the problem domain (e.g., it is safe to assume small interframe motion). Object detection faces issues similar to those described above, in that it is difficult for a human labeler to be consistent with respect to how the positive examples are cropped. In fact, Viola et al. [14] argue that object detection has inherent ambiguities that cause difficulty for traditional supervised learning methods. For this reason they suggest the use of a Multiple Instance Learning (MIL) [13] approach for object detection. The basic idea of this learning paradigm is that during training, examples are presented in sets (often called "bags"), and labels are provided for the bags rather than individual instances. If a bag is labeled positive it is assumed to contain at least one positive instance, otherwise the bag is negative. For example, in the context of object detection, a positive bag could contain a few possible bounding boxes around each labeled object (e.g. a human labeler clicks on the center of the object, and the algorithm crops several rectangles around that point). Therefore, the ambiguity is passed on to the learning algorithm, which now has to figure out which instance in each positive bag is the most "correct". Although one could argue that this learning problem is more difficult in the sense that less information is provided to the learner, in some ways it is actually easier because the learner is allowed some flexibility in finding a decision boundary. Viola et al. present convincing results showing that a face detector trained with weaker labeling (just the center of the face) and a MIL algorithm outperforms a state of the art supervised algorithm trained with explicit bounding boxes.

## Online Tracking Algorithms

A typical tracking system[34] is composed of three components: object representation, dynamic model and search mechanism. Since different components can deal with different challenges of object tracking, we analyze recent online tracking algorithms accordingly and show how to choose or design robust online algorithms for specific situations.

**Object Representation**

An object can be represented by either holistic descriptors or local descriptors. Color

histograms and raw pixel values are common holistic descriptors. Color histograms have been used in the mean-shift tracking algorithm and the particle based method. The advantages of histogram-based representations are their computational efficiency and effectiveness to handle shape deformation as well as partial occlusion. However, they do not exploit the structural appearance information of target objects. In addition, histogram-based representations are not designed to handle scale change although some efforts have been made to address this problem.[28, 29] Holistic appearance models based on raw intensity values are used in the Kanade-Lucas-Tomasi algorithm,30 the incremental subspace learning tracking method,1 the incremental tensor subspace learning method[31] and the `1-minimization based tracker.[7] However, tracking methods based on holistic representation are sensitive to partial occlusion and motion blur. Filter responses have also been used to represent objects. Haar-like wavelets are used to describe objects for boosting based tracking methods.4, 8 Porikli et al.32 use features based on color and image gradients to characterize object appearance with update for visual tracking. Local descriptors have also been widely used in object tracking recently due to their robustness to pose and illumination change. Local histograms and color information are utilized for generating confidence maps from which likely target locations can be determined.23 Features based on local histograms are selected to represent objects in the fragments-based method.3 It has been shown that an effective representation scheme is the key to deal with appearance change in object tracking.

**Adaptive Appearance Model**

As mentioned above, it is crucial to update appearance model for ensuring robust tracking performance and much attention has been paid in recent years to address this issue. The most straightforward method is to replace the current appearance model (e.g., template) with the visual information from the most recent tracking result. Other update algorithms have also been proposed, such as incremental subspace learning methods,1, 31 adaptive mixture model,21 and online boosting-based trackers.4, 23 However, simple update with recently obtained tracking results can easily lead to significant drifts since it is difficult to determine whether the new data are noisy or not. Consequently, drifting errors are likely to accumulate gradually and tracking algorithms eventually fail to locate the targets. To reduce visual drifts, several algorithms have been developed to facilitate adaptive appearance models in recent years. [33] propose a tracking method with the Lucas-Kanade algorithm by updating the template with the results from the most recent frames and a fixed reference template extracted from the first frame. In contrast to supervised discriminative object tracking, Grabner et al.5 formulate the update problem as a semi-supervised task where the drawn samples are treated as unlabeled data. The task is then to update a classifier with both labeled and unlabeled data. Specific prior can also be used in this semi-supervised approach6 to reduce drifts. Babenko et al.8 pose the tracking problem within the multiple instance learning (MIL) framework to handle ambiguously labeled positive and negative data obtained online for reducing visual drifts. Recently, Kalal et al.10 also pose the tracking problem as a semi-supervised learning task and exploit the underlying structure of the unlabeled data to select positive and negative samples for update. While much progress has been made on this topic, it is still a difficult task to determine when and which tracking results should be updated in adaptive appearance models to reduce drifts.

**Motion Model**

The dimensionality of state vector, xt, at time t depends on the motion model that a tracking method is equipped with. The most commonly adopted models are translational motion (2 parameters), similarity transform (4 parameters), and ane transform (6 parameters). The classic Kanade-Lucas-Tomasi algorithm16 is designed to estimate object locations although it can be extended to account for ane motion.33 The tracking methods1, 7, 31 account transformation of objects between two consecutive frames. If an algorithm is designed to handle translational movements, the tracking results would not be accurate when the objects undergo rotational motion or scale change even if an adaptive appearance model is utilized. We note that certain algorithms are constrained by their design and it may not be easy to use a different motion model to account for complex object movements. For example, the mean-shift based tracking algorithm17 is not equipped to deal with scale change or in-plane rotation since the objective function is not differentiable with respect to these motion parameters. However, if the objective function of a tracking algorithm is not differentiable with respect to the motion parameters, it may be feasible to use either sampling or stochastic search to solve the optimization problem.

**Dynamic Model**

A dynamic model is usually utilized to reduce computational complexity in object tracking as it describes the likely state transition, between two consecutive frames where xt is the state vector at time t. Constant velocity and constant acceleration models have been used in the early tracking methods such as Kalman filter-based trackers. In these

methods, the state transition is modeled by a Gaussian distribution. Since the assumption of constant velocity or acceleration is rather constrained, most recent tracking algorithms adopt random walk models1, 7 with particle filters.

**Search Mechanism**

Since object tracking can be formulated as an optimization problem, the state search strategy depends mainly on the objective function form. In the literature, either deterministic or stochastic methods have been utilized for state search. If the objective function is differentiable with respect to the motion parameters, then gradient descent methods can be used.16, 17, 33 Otherwise, either sampling4, 8 or stochastic methods1, 7 can be used. Deterministic methods based on gradient descent are usually computationally efficient, but suffer from the local minimum problems. Exhaustive search methods are able to achieve good tracking performance at the expense of very high computational load, and thus seldom used in tracking tasks. Sampling-based search methods can achieve good tracking performance when the state variables do not change drastically. Consequently, stochastic search algorithms such as particle filters are trade-offs between these two extremes, with the ability to escape from local minimum without high computational load. Particle filters have been widely used in recent online object tracking with demonstrated success.

## Performance Evaluation

In this section, we empirically compare tracking methods based on the above discussions and demonstrate how to choose and design effective algorithms. We evaluate 10 state-of-the-art tracking algorithms on 15 challenging sequences using different criteria. The test algorithms include: incremental visual tracker (IVT),1 variance ratio tracker (VRT),2 fragments-based tracker (FragT),3 online boosting tracker (BoostT),4 semi-supervised trackers (SemiT),5 extended semisupervised tracker (BeSemiT),6 `1 tracker (L1T),7 multiple instance learning tracker (MIL),8 visual tracking decomposition algorithm (VTD),9 and track-learning-detection method (TLD).10 Based on the above analysis, we categorize these algorithms in Table 2 which describes their object representation, motion model, dynamic model, search mechanism and characteristics. The challenging factors of the test sequences are listed in Table 3. For fair evaluation, we use the the source codes provided by the authors in all experiments. For the tracking methods which use particle filtering (i.e., IVT, L1T, and VTD), we use 300 particles in all tests. The other parameters of each tracking method are carefully selected in each method for best performance. It is worth noting that the FragT method is not an online method although the experimental comparison shows the necessity of adaptive appearance models

| Algorithm | Motion Model | Object Representation | Dynamic Model | Searching Mechanism | Characteristics |
|---|---|---|---|---|---|
| **IVT** | Affine transform | holistic gray-scale image vector | Gaussian | particle filter | generative |
| **FragT** | Similarity transform | local gray-scale histograms | - | sampling | generative |
| **VRT** | Translational motion | Holistic color histograms | - | mean-shift | discriminative |
| **BoostT** | Translational motion | holistic representation based on Haar-like, HOG and LBP descriptors | - | Sampling | Discriminative |
| **SemiT** | Translational motion | holistic representation based on Haar-like descriptor | - | Sampling | Discriminative |
| **BeSemiT** | Translational motion | holistic representation based on Haar-like, | - | sampling | Discriminative |

| | | | | | |
|---|---|---|---|---|---|
| | | HOG and color histograms | | | |
| **LIT** | Affine transform | holistic gray-level image vector | Gaussian | particle filter | generative |
| **MILT** | translational motion | holistic representation based on Haar-like descriptor | - | Sampling | Discriminative |
| **VTD** | similarity transform | holistic representation based on hue, saturation, intensity, and edge template | Gaussian | particle filter | generative |
| **TLD** | similarity transform | holistic representation based on Haar-like descriptor | - | Sampling | discriminative |

## Conclusions

Object tracking is one of the most important processing blocks in computer vision systems. In recent years, we have been assisting to a proliferation of tracking algorithms. It is a challenging task to develop effective and efficient appearance models for robust object tracking due to factors such as pose variation, illumination change, occlusion, and motion blur. In this article, we have provided an overview of tracking algorithms for such problems. We have pointed out how many traditional methods are relevant here. Existing online tracking algorithms often update models with samples from observations in recent frames. While much success has been demonstrated, numerous issues remain to be addressed. First, while these adaptive appearance models are data-dependent, there does not exist sufficient amount of data for online algorithms to learn at the outset. Second, online tracking algorithms often encounter the drift problems. As a result of self-taught learning, these mis-aligned samples are likely to be added and degrade the appearance models. Hence there is always a need for efficient algorithms. Improving accuracy, robustness and speed of algorithms is a problem that would continue to attract attention.

## References

[1] D. Ross, J. Lim, R.-S. Lin, and M.-H. Yang, "Incremental learning for robust visual tracking," International Journal of Computer Vision 77(1-3), pp. 125–141, 2008.

[2] R. T. Collins, Y. Liu, and M. Leordeanu, "Online selection of discriminative tracking features," IEEE Transactions on Pattern Analysis and Machine Intelligence 27(10), pp. 1631–1643, 2005.

[3] A. Adam, E. Rivlin, and I. Shimshoni, "Robust fragments-based tracking using the integral histogram," in Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, pp. 798–805, 2006.

[4] H. Grabner and H. Bischof, "On-line boosting and vision," in Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, pp. 260–267, 2006.

[5] H. Grabner, C. Leistner, and H. Bischof, "Semi-supervised on-line boosting for robust tracking," in Proceedings of European Conference on Computer Vision, pp. 234–247, 2008.

[6] S. Stalder, H. Grabner, and L. Van Gool, "Beyond semi-supervised tracking: Tracking should be as simple as detection, but not simpler than recognition," in Proceedings of IEEE Workshop on Online Learning for Computer Vision, 2009.

[7] X. Mei and H. Ling, "Robust visual tracking using l1 minimization," in Proceedings of the IEEE International Conference on Computer Vision, pp. 1436–1443, 2009.

[8] B. Babenko, M.-H. Yang, and S. Belongie, "Visual tracking with online multiple instance learning," in Proceedings of IEEE

Conference on Computer Vision and Pattern Recognition, pp. 983–990, 2009.
[9] J. Kwon and K. Lee, "Visual tracking decomposition," in Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, pp. 1269–1276, 2010.
[10] Z. Kalal, J. Matas, and K. Mikolajczyk, "P-n learning: Bootstrapping binary classifiers by structural constraints," in Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, pp. 49–56, 2010.
[11] I. Haritaoglu, D. Harwood, and L. Davis, "W4s: A real-time system for detecting and tracking people," in Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, pp. 962–968, 1998.
[12] M. de La Gorce, N. Paragios, and D. Fleet, "Model-based hand tracking with texture, shading and self-occlusions," in Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, 2008.
[13] Z. Sun, G. Bebis, and R. Miller, "On-road vehicle detection: A review," IEEE Transactions on Pattern Analysis and Machine Intelligence 28(5), pp. 694–711, 2006.
[14] M. Kim, S. Kumar, V. Pavlovic, and H. Rowley, "Face tracking and recognition with visual constraints in real-world videos," in Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, pp. 1–8, 2008.
[15] X. Zhou, D. Comaniciu, and A. Gupta, "An information fusion framework for robust shape tracking," IEEE Transactions on Pattern Analysis and Machine Intelligence 27(1), pp. 115–129, 2005.
[16] B. Lucas and T. Kanade, "An iterative image registration technique with an application to stereo vision," in Proceedings of International Joint Conference on Artificial Intelligence, pp. 674–679, 1981.
[17] D. Comaniciu, V. Ramesh, and P. Meer, "Kernel-based object tracking," IEEE Transactions on Pattern Analysis and Machine Intelligence 25(5), pp. 564–575, 2003.
[18] A. Azarbayejani and A. Pentland, "Recursive estimation of motion, structure, and focal length," IEEE Transactions on Pattern Analysis and Machine Intelligence 17(6), pp. 562–575, 1995.
[19] M. Isard and A. Blake, "Condensation – conditional density propagation for visual tracking," International Journal of Computer Vision 29(1), pp. 5–28, 1998.
[20] M. Black and A. Jepson, "Eigentracking: Robust matching and tracking of articulated objects using a view-based representation," in Proceedings of European Conference on Computer Vision, pp. 329–342, 1996.
[21] A. D. Jepson, D. J. Fleet, and T. F. El-Maraghi, "Robust online appearance models for visual tracking," IEEE Transactions on Pattern Analysis and Machine Intelligence 25(10), pp. 1296–1311, 2003.
[22] S. Avidan, "Support vector tracking," IEEE Transactions on Pattern Analysis and Machine Intelligence 26(8), pp. 1064–1072, 2004.
[23] S. Avidan, "Ensemble tracking," IEEE Transactions on Pattern Analysis and Machine Intelligence 29(2), pp. 261– 271, 2007.
[24] D. Lowe, "Distinctive image features from scale-invariant keypoints," International Journal of Computer Vision 60(2), pp. 91–110, 2004.
[25] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, 2005.
[26] T. Ojala, M. Pietikainen, and T. Maenpaa, "Multiresolution gray-scale and rotation invariant texture classification with local binary patterns," IEEE Transactions on Pattern Analysis and Machine Intelligence 24(7), pp. 971–987, 2002.
[27] P. P´erez, C. Hue, J. Vermaak, and M. Gangnet, "Color-based probabilistic tracking," in Proceedings of European Conference on Computer Vision, pp. 661–675, 2002.
[28] R. T. Collins, "Mean-shift blob tracking through scale space," in Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, pp. 234–240, 2003.
[29] S. T. Birchfield and S. Rangarajan, "Spatiograms versus histograms for region-based tracking," in Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, 2, pp. 1158–1163, 20–25 June 2005.
[30] S. Baker and I. Matthews, "Lucas-Kanade 20 years on: A unifying framework," International Journal of Computer Vision 56(3), pp. 221–255, 2004.
[31] X. Li, W. Hu, Z. Zhang, X. Zhang, and G. Luo, "Robust visual tracking based on incremental tensor subspace learning," in

Proceedings of the IEEE International Conference on Computer Vision, 2007.

[32] F. Porikli, O. Tuzel, and P. Meer, "Covariance tracking using model update based on lie algebra," in Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, pp. 728–735, 2006.

[33] L. Matthews, T. Ishikawa, and S. Baker, "The template update problem," IEEE Transactions on Pattern Analysis and Machine Intelligence 26(6), pp. 810–815, 2004. [34] M. Everingham, L. Van Gool, C. Williams, J. Winn, and A. Zisserman, "The pascal visual object classes (voc) challenge," International Journal of Computer Vision 88(2), pp. 303–338, 2010.

[34] Wang, Qing; Chen, Feng; Xu, Wenli; Yang, Ming-Hsuan "An experimental comparison of online object-tracking algorithms" Wavelets and Sparsity XIV. Edited by Tsakalakos, Loucas. Proceedings of the SPIE, Volume 8138, pp. 81381A-81381A-11 (2011).