Recent Trends in Computational Intelligence & Image Processing - RICP 2017
23 February 2017, India
College of Engineering, Vadakara, Calicut, India

Visual Tracking; A Brief Survey

Sreelekha M[1], Sufyan P[2]

[1]MEA Engineering College, State Highway 39, Nellikunnu-Vengoor, Perinthalmanna, Malappuram, Kerala

[2]Assistant Professor, MEA Engineering College, State Highway 39, Nellikunnu-Vengoor, Perinthalmanna, Malappuram, Kerala

Abstract: ***Visual tracking is very essential task in many application of computer vision such as autonomous robot navigation, surveillance, vehicle navigation etc. Tracking objects in real life scenarios is a difficult question; therefore, it is still the most active areas of research in computer vision. Good trackers should perform well in large number of videos involving lighting changes, occlusion, clutter, the motion of the camera, a low contrast etc. Visual tracking involves the detection of moving objects of interest as well as tracking of such objects from one frame to another. Its main task is to find and track moving objects or multiple objects in the image sequence. There are typically three stages of video analysis; object detection, object tracking, and object reassembly. This paper presents a brief survey of various video object tracking methods based on sparse representations.***

Keywords: Visual Tracking, Occlusion, Clutter, Sparse Representation

1. Introduction

Visual tracking is an important and challenging task in the field of computer vision. The availability of high-quality and cheap cameras, the rise of high-power computers and the growing demand for automated video analytics have generated considerable interest in visual tracking algorithms.

Visual tracking in video can be defined as a problem [1] in estimating the trajectory of an object in the image plane as it moves around the scene. It is a process of segmenting an object of interest from a video scene and tracking its motion, orientation, and occlusion. In addition, the tracker may also give information centered on the object, such as orientation, region, or contour, according to the tracking domain. Video object detection and tracking has a wide range of applications in video processing, such as medical imaging, vision-based control, video compression, human-computer interface, video surveillance, augmented reality and robotics.

The tracking algorithm may be selected based on object detection techniques, features, object representations, and object tracking algorithms. Common visual characteristics are color, texture, optical flow and edge. The purpose of visual tracking is to create objects over time by finding its position in the video sequence. The correspondence between detection of objects and creation of objects through frames may be done individually or collectively. In the first stage, the region of interest in each frame is implemented by the object detection algorithm, and then the objects corresponding to the cross-frames are tracked. In the final stage, the object region is projected by iteratively updating the position of the object obtained from the previous frame.

Recently, tracking methods based on sparse representations have been proposed. Mei and Ling [2] and Mei *et al.* [3]) use a number of holistic templates of a target object as an appearance model and determine the most likely object regions by solving one *l*1 minimization problem for each drawn particle. Most of these methods employ holistic representation schemes and hence do not perform well when target objects are heavily occluded. Liu *et al*. [4] employ a local appearance model based on histograms of sparse coefficients and the mean-shift algorithm for object tracking. Jia *et al.* [5] develop coarse and fine structural local appearance models based on sparse representation of both multiple templates and local appearance models.

2. Literature Survey

In paper [2], the author develops a robust visual tracking framework by transforming the tracking problem into finding a sparse approximation in the template subspace, and proposes using trivial templates to deal with the occlusion, so that each trivial template has only one non- zero element. Then, during the trace, the target candidate is represented as a linear combination of template templates consisting of two target templates (obtained from the previous frame) and trivial templates. The number of target templates is far less than the number of ordinary templates. Intuitively, a good target candidate can be effectively represented by the target template. Since coefficients corresponding to trivial coefficients tend to be zero, this results in a sparse coefficient vector. In the case of occlusion a limited number of trivial coefficients will be activated, but the entire coefficient vector will remain sparse. In contrast, bad target candidates often result in dense representations. The sparse representation is achieved by solving the *l*1 regularized least squares problem, which can be done efficiently by convex optimization. The candidate with the minimum target template projection error is then selected as the tracking result. The tracking is then guided by the Bayesian State Inference Framework, where the particle filter is used to propagate the sample distribution over time. This approach includes two additional components to further enhance robustness. First, they perform a non-negative constraint on the sparse representation. These constraints are particularly useful for eliminating clutter similar to a target template having a reverse intensity pattern. Constraints are implemented by including positive and negative generic templates in the template set. Second, the target template set is dynamically updated to maintain a representative template throughout the tracking process. This is done by using the coefficients in the sparse representation to adjust the template weights.

In paper [3], the authors propose an efficient tracking algorithm with minimum error bound and occlusion detection. The first contribution is to greatly improve the run-time efficiency of the *l*1 tracker by using the error bounds derived from the least squares method. In particular, it is observed that the computationally expensive reconstruction error in the sparse constraint *l*1 minimization is limited by the least squares reconstruction error, which can be effectively calculated. Reconstruction error observation facilitates the design of a Boundary Particle Re-sampling (BPR) algorithm, which greatly improves the speed of the tracking algorithm without sacrificing re-sampling accuracy. In particular, the probability of tracking the samples is calculated in two phases. In the first stage, the sample is reconstructed by simply projecting the sample onto the target template subspace. Reconstruction is solved by running a linear least squares equation several steps faster than the typical *l*1 minimization function. In the second stage, only the dynamically selected samples with smaller reconstruction errors from the previous stage are reconstructed by *l*1 minimization, and most samples are filtered out without solving *l*1 minimization. By this two-stage reconstruction, the computational cost is greatly reduced.

The second contribution is through the study of the reconstruction coefficient of the occlusion detection method and then used to improve the template update process. In order to prevent the wrong information from contaminating the template set, a robust occlusion detection method is introduced here. The idea is to first establish an occlusion map from the trivial coefficient, which indicates pixel-by-pixel image contamination in a given candidate. Then, if an occlusion is detected, the occlusion map is used for occlusion detection, and the candidate is not added to the template set.

In paper [4], the authors propose and test a robust tracking algorithm with local sparse appearance model (SPT). The key component of the algorithm is the static sparse dictionary, which is used to limit the drift and maintain its linearity across subspaces. The dynamic base distribution is represented by the sparse coding histogram and updated online. The voting map based on sparse representation, the reconstruction error regularity and mean shift, which is used to locate the center of the final object. In addition to all of these contributions, a new sparse dictionary learning method, called K-choice, was introduced to learn the sparse representation of the library of objects. Figure 1 shows an overview of the proposed algorithm.

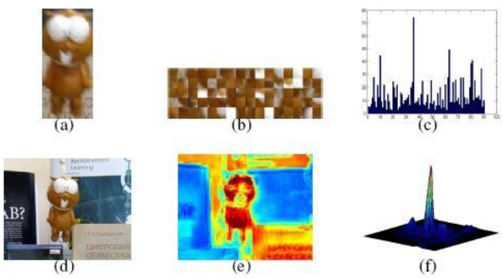

Figure 1: The target appearance (a) is modeled with a dictionary (b) and a sparse coding histogram (c). The confidence map (e) of the image (d) is the inverse of the reconstruction error from the learned dictionary. Target center is found by voting and sparse constraint with regularized mean-shift on the probability map (f).

The contributions of this paper are:

- A natural combination of static sparse dictionaries and dynamic on-line update basis distributions that take into account the adaptability and stability.
- A new sparse dictionary learning method by directly selecting the most representative base.
- Sparse Representation Voting Map and Sparse Constrained Regularized Mean Shift Object Tracking.

In paper [5], the authors propose an efficient tracking algorithm based on rough and fine structure local sparse model. The proposed method utilizes the local appearance uniformity while the global appearance changes over time. An image patch in a fixed-space layout in a target area is extracted and encoded using a dictionary made up of patches of a plurality of target templates having sparsity constraints. The coding coefficients of the blocks spanning multiple templates are integrated by averaging and alignment pooling to obtain a robust representation of the target object. This operation helps to locate the object accurately and efficiently handles the partial occlusion by exploring the consistent parts of the object in the sequence of images. To make the representation more unique and robust, the likelihood of candidate regions is calculated here based on a combination of patches extracted with a coarse and fine strategy. The dictionary for local sparse coding is generated from a collection of collected templates that are sequentially updated based on the incremental subspace learning method [6]. A scheme of detecting the occlusion portion is also introduced to update the template more accurately without the occlusion pixels. The update module facilitates the proposed tracker account for target appearance changes caused by posture changes and lighting changes. In this paper, firstly, the sparse code of the local region is calculated by the average and alignment pool to simulate the appearance of the object for visual tracking. A new algorithm for constructing coarse and fine dictionaries is proposed for robust representation. Secondly, a template updating scheme based on incremental subspace learning is proposed to describe the appearance change of objects. The template update module is equipped with an occlusion detection module to include pixels belonging to the foreground object. Third, extensive experiments were conducted on a large baseline data set to evaluate the performance of the proposed algorithm for prior art approaches.

3. Conclusion

In this article, we present an extensive survey of visual tracking methods based on sparse representations and also gave a brief review of related topics. In first paper they propose using a sparse representation for robust visual tracking. They model tracking as a sparse approximation problem and solve it through an *l*1-regularized least squares approach. In second paper they propose an efficient BPR-*l*1 tracker with minimum error bound and occlusion detection. They employ a two stage sample probability scheme, where most samples with small probabilities from first stage are filtered out without solving the computational expensive *l*1 minimization. In third paper they have developed and tested a robust tracking algorithm with a static sparse dictionary and dynamic online updated basis distribution, which can adapt to appearance changes and limit the drifting. In last paper, they propose an efficient tracking algorithm based on coarse and fine structural local sparse appearance models with adaptive update.

References

[1] Yilmaz, O. Javed and M. Shah, "Object tracking: A survey", ACM Computing Surveys", 2006.
[2] X. Mei and H. Ling, "Robust visual tracking using *l*1 minimization," in *Proc. IEEE 12th Int. Conf. Comput. Vis.*, Sep./Oct. 2009, pp. 1436–1443.
[3] B. Liu, J. Huang, L. Yang, and C. A. Kulikowsk, "Robust tracking using local sparse appearance model and K-selection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2011, pp. 1313–1320.
[4] X. Mei, H. Ling, Y. Wu, E. Blasch and L. Bai, "Minimum error bounded efficient ℓ1 tracker with occlusion detection," *CVPR 2011*, Providence, RI, 2011, pp. 1257-1264.
[5] X. Jia, H. Lu and M. H. Yang, "Visual Tracking via Coarse and Fine Structural Local Sparse Appearance Models," in *IEEE Transactions on Image Processing*, vol. 25, no. 10, pp.4555-4564, Oct. 2016.
[6] D. A. Ross, J. Lim, R.-S. Lin, and M.-H. Yang, "Incremental learning for robust visual tracking," *Int. J. Comput. Vis.*, vol. 77, nos. 1–3, pp. 125–141, 2008.

Authors Profile

Sreelekha M received the B.Tech degree in Information Technology from MEA Engineering College, Perinthalmanna in 2015 and doing M.Tech in Computer Science from MEA Engineering College, Perinthalmanna.

Sufyan P received the B.Tech degree in Computer Science from College of Kidangoor, Kottayam in 2013 and received the M.Tech degree in Computer Science from MES College of Engineering, Kuttipuram in 2015. He is currently working as an Asst. Professor in MEA Engineering College, Perinthalmanna.