DOI: 10.1051/matecconf/20166102020

Visual Tracking Using L2 Minimization

Zhijun Pei[1,a] and Lei Han[1]

[1]*Department of Electronic Engineering, Tianjin University of Technology and Education, Tianjin, China*

Abstract. Visual tracking has been an active research topic in the computer vision applications. By modeling the target appearance with a sparse approximation over a template set, sparse representation has been applied to the visual tracker, which called L1 tracker. Due to the need to solve the L1 norm related minimization problem for many times, this L1 tracker is very computationally demanding. Although various fast numerical solver is developed to solve the resulting L1 norm related minimization problem, the framework is still a L1 norm related minimization model. Similar to the face recognition problem, sparse approximations may not deliver the desired robustness and a simple L2 approach to the visual tracking problem is not only robust, but also much faster. It may be possible to apply the L2 minimization, instead of L1 minimization, to the visual tracking problems, which has been verified by experiments on challenging sequences in the paper.

1 Introduction

Visual tracking is a critical task in many computer vision applications and has been widely applied in various fields such as the automated surveillance, vehicle navigation robotics. The challenges in designing a robust visual tracking algorithm are caused by the presence of noise, occlusion, varying viewpoints, background clutter, and illumination changes [1]. Although a variety of tracking algorithms have been proposed to overcome these difficulties in last two decades, many challenging problems still remain when designing a practical visual tracking system. An effective appearance model is of prime importance for the success of a tracking algorithm. Tracking algorithms can be generally categorized as either discriminative or generative based on their appearance models [2]. Tracking problem is formulated as a binary classification problem in discriminative tracking methods. Discriminative trackers locate the object region by finding the best way to separate object from background. In another way, 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. And the appearance model is often dynamically updated during the tracking to adapt to pose and illumination changes of the object.

Recently, motivated by recent advances in sparse representation and its applications in computer vision such as robust face recognition, sparse representation have been applied to tracking problem, where a tracking candidate is sparsely represented by target templates and trivial templates [3]. In these approaches, the sparse representation is obtained via solving a L0-norm related minimization or L1-norm related minimization problem [4]. However, the computational complexity of this tracker is rather high, thereby limiting its applications in real time scenarios. It is well known that L0-norm related minimization is an NP-hard problem and the large scale L1-norm related minimization is also a challenging problem. There have been great progresses on fast numerical methods for solving large-scale L1-norm related minimization problems such as the so-called accelerated proximal gradient (APG) method [5]. Although the introduction of very fast numerical method to solve the resulting L1 norm minimization problems may lead to a real time L1 tracker, it still built on the same framework of the L1 tracker.

In the L1 tracker framework, hundreds of L1-norm related minimization problems need to be solved for each frame during the tracking process. An efficient solver for the L1 norm related problems has been the key in practice. As for the application of compressive sensing (CS) to the problem of face recognition, it has been shown that faster, more accurate, and more robust methods may be achieved by modelling outliers explicitly and using the L2-norm instead of L1-norm [6]. In many applications of compressive sensing, the sparsity is assumed, rather than proven or measured. However, the sparsity assumption may not supported by data of the practical problems. This implications are important for the application of compressive sensing to face recognition but also to other problems where sparsity is assumed rather than proven, such as visual tracking problems. So the possibility of the application the L2 minimization for the visual tracking problems is discussed, and which has been verified by the experiments in the paper.

2 L1 minimization tracking

A brief review on the L1 tracker within the particle filter framework is first given bellow.

[a] Corresponding author: peizj@tute.edu.cn

© The Authors, published by EDP Sciences. This is an open access article distributed under the terms of the Creative Commons Attribution License 4.0 (http://creativecommons.org/licenses/by/4.0/).

2.1 Particle filter

Tracking can be considered as an estimation of the state for a time series state space model. The problem is formulated in probabilistic terms. The particle filter, also known as the sequential Monte Carlo method, is one of the most popular approaches to provide solutions. It recursively constructs the posterior probability density function of the state space using Monte Carlo integration. It has been developed in the computer vision community and applied to tracking problems [7].

The particle filter is a Bayesian sequential importance sampling technique for estimating the posterior distribution of state variables characterizing a dynamic system. It provides a convenient framework for estimating and propagating the posterior probability density function of state variables regardless of the underlying distribution. It consists of prediction step and update step. Let x_t denote the state variable describing the affine motion parameters of an object at time t. Given all available observations $z_{1:t-1}=\{z_1, z_2, \cdots, z_{t-1}\}$ up to time t−1, the predicting distribution of x_t is recursively computed as

$$p(x_t | z_{1:t-1}) = \int p(x_t | x_{t-1})\, p(x_{t-1} | z_{1:t-1})\, dx_{t-1}$$

At time t, the observation z_t is available and the state vector is updated using the Bayes rule

$$p(x_t | z_{1:t}) = p(z_t | x_t)\, p(x_t | z_{1:t-1}) / p(z_t | z_{1:t-1})$$

In the particle filter, the posterior $p(x_t | z_{1:t})$ is approximated by a finite set of N samples $\{x_{it}\}$, i=1,⋯ ,N with importance weights w_{it}. The candidate samples x_{it} are drawn from an importance distribution $q(x_t | x_{1:t-1}, z_{1:t})$ and the weights of the samples are updated. The samples are resampled to generate a set of equally weighted particles according to their importance weights to avoid degeneracy.

In the tracking framework, the object motion of two consecutive frames is model with the affine image warping. The state variable x_t is modeled by the parameters of the affine transformation parameters. The image region of interest z_t is cropped from the image as the target templates. A Gaussian distribution is employed to model the state transition distribution $p(x_t | x_{t-1})$. The observation model $p(z_t | x_t)$ reflects the similarity between a target candidate and the target templates, which is formulated from the error approximated by the target templates using L1 minimization.

2.2 L1 minimization tracking

The global appearance of one object under different illumination and viewpoint conditions is known to lie approximately in a low dimensional subspace. With template image columns are stacked to form a one dimension vector, the target template set can represented as

$$T = [t_1 ... t_n] \in R^{d \times n}, (d \gg n)$$

Then a tracking result $y \in R^d$ approximately lies in the linear span of T, that is

$$y \approx Ta$$

where *a* is the target coefficient vector.

In many visual tracking scenarios, target objects are often corrupted by noise or partially occluded, which may affect any part of the image and appear at any size on the image. The locations of corruption can differ for different tracking images and are unknown to the computer. The trivial templates I = [i_1, i_2, ...,i_d] can be used to capture the occlusion. To incorporate the effect of occlusion and noise, a tracking object can represent as

$$y=Ta+Ie = As \tag{1}$$

where the trivial template is a vector with only one nonzero entry, here *I* can be chosen as an identity matrix, and *e* is called trivial coefficient vector.

The system in equation (1) is underdetermined and does not have a unique solution for the vector *s*. The error caused by occlusion and noise typically corrupts a fraction of the image pixels. Therefore, for a good target candidate, there are only a limited number of nonzero coefficients in vector *e* that account for the noise and partial occlusion. In order to have a sparse solution, considering the problem as a L1-regularized least squares problem,

$$\min_s \|As - y\|_2^2 + \lambda \|s\|_1 \tag{2}$$

Where $\|.\|_1$ and $\|.\|_2$ denote the L1 norms and L2 norms respectively.

Then the tracking result can be found by finding the smallest residual after projecting on the target template subspace. Therefore, the tracking result is the sample of states that obtains the smallest error or the largest probability.

3 L2 minimization tracking

Just as discussed above, the visual L1 tracking frame caste the tracking problem as finding a sparse approximation in a template subspace, where handling occlusion using trivial templates such that each trivial template has only one nonzero element. During tracking, a target candidate is represented as a linear combination of the template set composed of both target templates obtained from previous frames and trivial templates. The number of chosen target templates are far fewer than the number of trivial templates. Intuitively, a good target candidate can be efficiently represented by the target templates. This leads to a sparse coefficient vector, since coefficients corresponding to trivial templates, named trivial coefficients, tend to be zeros. In the case of occlusion or other unpleasant issues such as noise corruption or background clutter, a limited number of trivial coefficients will be activated, but the whole coefficient vector remains sparse. A bad target candidate, on the contrary, often leads to a dense representation. The sparse representation is achieved through solving an L1-regularized least squares problem. Then the candidate with the smallest target template projection error is chosen as the tracking result. After that, tracking is led by the Bayesian state inference framework in which a particle filter is used for propagating sample distributions over time. The target template set is dynamically updated to keep the representative templates throughout the tracking procedure. This is done by adjusting template weights by using the coefficients in the sparse representation.

In the proposed L1 tracker, hundreds of L1-norm related minimization problems need to be solved for each frame during the tracking process. An efficient solver for the L1 norm related problems has been the key to use the L1 tracker in practice. Based on the interior point method, the solver for the L1 norm minimizations turns out to be too slow for tracking. In order to reduce the number of particles, which equal to the number of the 1 norm minimizations for solving, a minimal error bounding strategy can be introduced, but it is still far away from being real time. Although using accelerated proximal gradient approach can further improve the L1 Tracker, L1-norm related minimization problems still need to be solved for each frame [5].

Similar to the face recognition problem, one argument with the analysis of the L1 tracker might be that the L1 term is intended to achieve robustness, rather than indicating a belief in the sparsity of the coefficients. This is an important distinction. The L1-norm is used in compressive sensing as a tractable alternative to the L0-norm. Sparseness does not necessarily lead to robustness to the presence of outliers. To achieve robustness, one could use L1-regression as follows:

$$\min_{\alpha} \|y - A\alpha\|_{L1} \qquad (3)$$

which avoids overly penalizing gross outliers.

The L1-norm in compressive sensing and L1-norm in L1-regression are, however, two unrelated uses of the same norm. The two applications differ in the quantities to which the L1 norm is applied. Robustness cannot therefore be used to justify applying the L1-norm to the coefficients without further explanation. No such explanation is given, however. The problem with (3) is that solving such a linear program can be computationally expensive as the data size grows. Rather than resort to approaches, however, a faster, more accurate, and more robust methods may be achieved by modelling outliers explicitly and using the L2-norm.

In contrast to the L1 case, it is possible to estimate using the L2 norm by solving

$$\arg\min \|y - A\alpha\|_{L2}^{2}$$

Even when the system is overdetermined, the optimal solution, in the sense of the smallest reconstruction error, can be recovered by

$$\alpha = (\mathrm{A}^{T}\mathrm{A})^{-1}\mathrm{A}^{T}y$$

To efficiently solve the related pseudo-inverse, by QR factorisation of A, then A = QR. where Q forms a orthonormal basis, and R is an upper triangle matrix. Therefore, we can have

$$\alpha = R^{-1}\mathrm{Q}^{T}y$$

Here $R^{-1}\mathrm{Q}^{T}$ remains the same for all y. So it just need to computed once and stored for each frame during the tracking process. Once the coefficients are estimated, one can find the state location of identified object via minimizing the residuals

$$c^{*}(y_t^i) = \arg\min_{x_t^i} \left\| y_t^i - T_t c_T^i \right\|_{L2}^{2}$$

where i is the particles index of *t* frame; c_T^i is the sub-vector consisting of components of a and T_t is a submatrix of A, both corresponding to the basis of particle i.

The particle filter provides an estimate of posterior distribution of random variables related to Markov chain. In visual tracking, it gives an important tool for estimating the target of next frame without knowing the concrete observation probability. At the frame t, denote x_t which describes the location and the shape of the target, $y_{1:t-1} = \{y_1, y_2, \cdot\cdot\cdot, y_{t-1}\}$ denotes the observation of the target from the first frame to the frame t − 1. The optimal state for the frame t is obtained according to the maximal approximate posterior probability. The observation model reflects the similarity between a target candidate and the target templates, and the observation likelihood of state x_t^i is given as

$$p(y_t \mid x_t^i) = \exp\{-a\left\| y_t^i - T_t c_T^i \right\|_{L2}^{2}\} / \Gamma$$

where α is a constant controlling the shape of the Gaussian kernel, Γ is a normal factor. Then, the optimal state x_t^* of frame t is obtained by

$$x_t^* = \arg\max_{x_t^i \in S_t} p(y_t \mid x_t^i)$$

In addition, a template update scheme is also adopted to overcome pose and illumination changes.

4 Experiments and discussions

In order to evaluate the performance of the proposed tracking framework with L2 minimization method, two videos are used in the experiments. The proposed tracker is implemented in MATLAB2013b on an Intel Core(TM) i3 CPU with 2.13 GHz.

The first test sequence is an infrared (IR) image sequence from VIVID benchmark dataset PkTest02 [8]. Some samples of the final tracking results are demonstrated in Figure 1, where six representative frames of the video sequences are shown. The frame indexes are 26, 46, 56, 66, 76 and 96. The target to background contrast is very low and the noise level is high for these IR frames. From Figure 1, it is shown that the tracker is capable of tracking the object all the time even with severe occlusions by the trees on the roadside. It keeps tracking it in the rest of the sequences. There is no case that fails to track the target in the some index frame. Similar to the proposed L1 tracker, our proposed method can also avoids fails problem and is effective under low contrast and noisy situation. The proposed tracker with L2 minimization achieves similar results to L1 method but more fast.

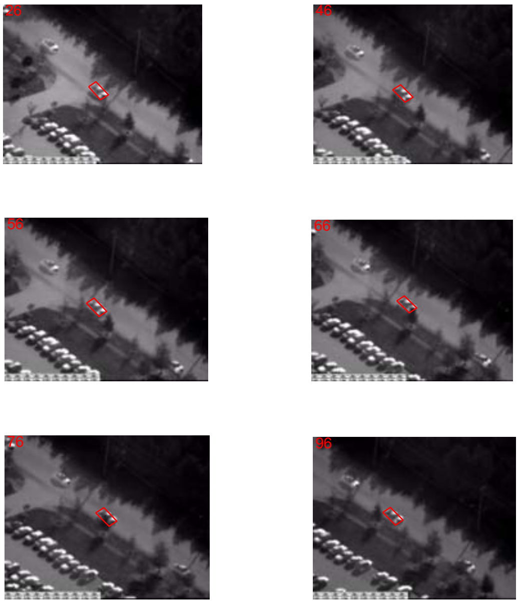

Figure 1. Some samples of the final tracking results

We also evaluate our tracking algorithm with the challenging sequences such as David indoor sequence, where there exits scale, pose and illumination change [2]. For the David indoor sequence shown in Figure 2, the illumination and pose of the object both change gradually. When using the feature templates proposed by [2], where the adopted features are similar to generalized Haar-like features which have been shown to be robust to scale and orientation change, the proposed L2 methods perform well on this sequence. Even if a shift happen, it can recover the failure and track the target properly again. The proposed tracker is robust to pose and illumination changes as the object appearance can be modelled well by random projections, based on compressed sampling, and templates online update is used. Our tracker achieves similar results to compressing tracking method.

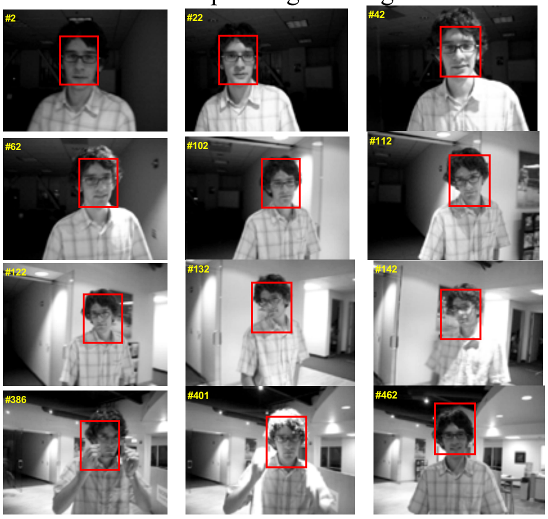

Figure 2. some tracking results of David sequence

5 Conclusion

Despite that numerous algorithms have been proposed, object tracking remains a challenging problem. As for L1 trackers, sparse representation has been applied to visual tracker by modeling the target appearance using a sparse approximation over a template set, which needs to solve an L1 norm related minimization problem for many times. While these L1 trackers showed impressive tracking accuracies, they are very computationally demanding and the speed bottleneck is the solver to L1-norm minimizations. Based on the accelerated proximal gradient approach, a fast numerical solver is developed to solve the resulting L1 norm related minimization problem, But it still build on a L1 norm related minimization model. Among the face recognition researches, it has been shown that sparse approximations may not deliver the robustness or performance desired and a simple L2 approach to the face recognition problem is not only more accurate, it is also more robust, and much faster. So the possibility of the application the L2 minimization for the visual tracking problems is investigated. The running time efficiency and tracking accuracy of the proposed tracker is validated with the evaluation involving two challenging sequences. The proposed tracking algorithm with L2 minimization have demonstrated good robustness in various tracking environments.

Acknowledgement

This work is supported by the Science and Technology Development Foundation of Tianjin Colleges and Universities under Grant No. 20140716. And this work is also partially supported by Research Foundation of Tianjin University of Technology and Education under Grant No. KJY1312.

References

1. A. Yilmaz, O. Javed, and M. Shah, ACM Computing Survey, Vol .38, No.4 (2006)
2. Kaihua Zhang, Lei Zhang, and Ming-Hsuan Yang, European Conference on Computer Vision (ECCV 2012), Florence, Italy, October (2012)
3. X. Mei and H. Ling, IEEE Trans. on Pattern Analysis and Machine Intelligence (PAMI), 33(11) (2011)
4. X. Mei and H. Ling, IEEE International Conference on Computer Vision (ICCV), Kyoto, Japan (2009)
5. C. Bao, Y. Wu, H. Ling and H. Ji, IEEE Conf. on Computer Vision and Pattern Recognition (CVPR), Rhode Island, 157(10) (2012)
6. Qinfeng Shi, Anders Eriksson, Anton van den Hengel, Chunhua Shen, IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 11), Colorado Springs, USA, June 21-23, 42(7) (2011)
7. S. K. Zhou, R. Chellappa, and B. Moghaddam, IEEE Trans. Image Processing, 11 (2004)
8. R. Collins, X. Zhou, and S. K. Teh, IEEE Workshop on Performance Evaluation of Tracking and Surveillance, (PETS'05), Breckenridge CO,Jan (2005)