Signal Processing: *Image Communication* 34 (2015) 32–44

Contents lists available at ScienceDirect

Signal Processing: *Image Communication*

journal homepage: www.elsevier.com/locate/image

Weak label for fast online visual tracking

Fei Cheng, Kai Liu*, Jin Zhang, Wenwen Ding

School of Computer Science and Technology, Xidian University, Xi'an, China

ARTICLE INFO

Article history:
Received 4 September 2014
Received in revised form
5 March 2015
Accepted 5 March 2015
Available online 13 March 2015

Keywords:
Visual tracking
Weak label
Least square support vector machine (LS-SVM)
Acceleration

ABSTRACT

Tracking by detection is becoming more and more popular in recent years. By training a classifier based on the image patches generated around the target, visual tracking can be formed as a classification problem. Most tracking by detection methods assign a constant label to the sample according to the distance between the sample and the center of the target. However, we find that the criterion to assign the label of each sample can be relaxed as long as some properties are preserved. By using the relaxed criterion which we call weak label, a visual tracking algorithm based on least square support vector machine (LS-SVM) and circulant matrix is proposed in this paper. Unlike the previous algorithm using LS-SVM and circulant matrix, the exploiting of weak label can make the coefficients of the support vectors a constant matrix. The calculation of the coefficients in each frame can be reduced, thus the proposed tracking algorithm can run very fast. On the test video sequences, the proposed tracker implemented in MATLAB runs over 300 frames per second on average on a machine with a 3.0 GHz CPU. Experimental results demonstrated the efficiency and accuracy of the proposed method.

1. Introduction

As a foundation problem in computer vision, visual tracking has found its applications in various fields such as man–machine interaction, visual surveillance, video analysis and augmented reality [1]. Though many tracking approaches [2–14] have been proposed in the literature, visual tracking remains to be a challenge due to background clutter, illumination variation, abrupt motion, etc.

A widely used framework in visual tracking is tracking by detection due to the achievements in object detection [15]. Tracking by detection usually treats visual tracking as a two-class (foreground and background) classification problem. First, a classifier is trained based on the training samples generated in the current frame by a specific criterion. Then, image patches to be classified as foreground or background are generated in a sliding window manner in the next frame. Finally, the image patch produces the maximum classifier response which is predicted as the target is used to update the classifier incrementally.

Grabner et al. [3] proposed to use boosting [15] method and adapted it into an online version for visual tracking. However, in [3] only one positive sample and multiple negative samples are used to train the classifier. The non-balanced training samples may lead to over-fitting and drifting. To alleviate the drifting problem during tracking, Grabner et al. [5] proposed to use online semi-supervised boosting, in which the training samples in the first frame are labeled and unlabeled in the following frames. Babenko

Abbreviations: LS-SVM, least square support vector machine; MIL, multiple instance learning; FFT, fast Fourier transform; FCT, fast compressive tracking; OAB, online adaboost; CSK, circulant structure kernel; WLCSK, weak label circulant structure kernel; DFT, discrete Fourier transform

* Corresponding author. Tel.: +86 13892810532.
E-mail addresses: chengfei8582@163.com (F. Cheng), kailiu@mail.xidian.edu.cn (K. Liu), jinzhang.cv@gmail.com (J. Zhang), dww2048@163.com (W. Ding).

http://dx.doi.org/10.1016/j.image.2015.03.002
0923-5965/

et al. [8] proposed online multiple instance learning (MIL) [16] for tracking. In MIL, the positive and negative samples are put into positive and negative bags. Then a classifier is trained based on these positive and negative bags instead of on the samples directly. The MIL tracker is shown to have good performance, especially in alleviating drifting problem. Recently, Ji et al. [17] proposed to model the appearance of the object with random ferns and template library.

Although tracking by detection framework achieves high tracking accuracy, the tracker may not be so efficient. This is mainly due to the complexity of the classifiers adapted from object detection [2,3,8,11]. These classifiers are suitable for object detection where accuracy is more emphasized than speed. But in visual tracking, speed is as important as accuracy since usually visual tracking is processed online so that the high accuracy would become trivial without the guarantee of speed. Zhang et al. [14] proposed to use random projection to reduce the high dimensional feature space of the training samples. A simple naive Bayes classifier is trained with the training samples in the low dimensional feature space. The low dimensional representation and naive Bayes classifier make the tracker very efficient. Based on CT, Song [18] proposed to select the informative features in compressive domain by the law of maximum entropy energy based on CT. Furthermore, Zhang et al. [19] proposed two improved algorithms named FCT and SFCT. FCT employs a coarse-to-find search and invariant scale features to improve the performance of tracker.

Most of the tracking by detection methods [3,5,8,14,11] generate training samples sparsely, since generating samples in a sliding window manner would result in thousands of samples which would make the training process of the tracker too slow to be applied online. It is clear that there exist a lot of redundancies in the training samples when they are generated in a sliding window manner. Henriques et al. [13] proposed a new theoretical framework to solve the redundancy problem by showing that taking subwindows of an image as training samples induces circulant structure. Then an algorithm based on the fast Fourier transform (FFT) is used to train the LS-SVM classifier from all subwindows. LS-SVM can also be named kernel ridge regression and kernel regularized least-squares in the literature. Another work similar to [13] is the correlation filter proposed by Bolme et al. [9]. The difference between [13] and [9] is that the former one uses kernel trick [20] to map the original feature into a higher dimensional feature space in which the dot product is processed implicitly. Zhang et al. [21] proposes STC which employs a Bayesian view to utilize the surrounding context to determine the position of object and adapts the scale of bounding box during the tracking process. This tracker also runs fast by employing FFT operations and obtains favorable results.

The label for a training sample in [13] is generated according to the distance between the center of the sample and the center of the target. And the label is identical for training samples with the same distance throughout the tracking process. However, we found that by applying the relaxed weak label which will be discussed in the following sections, the coefficient of support vectors in [13] can be a fixed value matrix calculated in advance before tracking. Thus the computation of the coefficient of support vectors in each frame can be omitted while the tracking accuracy can be preserved. The contributions of this paper can be summarized as follows. Firstly, the concept "weak label" is introduced to assign labels for different training samples. Secondly, by treating the kernel function as a similarity measurement, we find that the representation of kernel matrix (will be discussed in the following sections) can be used to approximate the weak label. Lastly, we derive an approximation of the learned coefficient in [13] that accelerates the processing speed significantly.

The rest of this paper is organized as follows. Section 2 introduces some basic conclusions of LS-SVM and circulant matrix discussed in [13]. Section 3 presents a detailed discussion of the proposed method. Experimental results are reported in Section 4. Finally, the conclusion is drawn in Section 5.

2. Circulant structure based tracking

In this section, some related backgrounds on LS-SVM and circulant matrix will be introduced.

2.1. LS-SVM

Given a set of training samples $(\mathbf{x}_i, y_i)_{i=1}^m$, a classifier $f(\mathbf{x}) = \langle \mathbf{w}, \mathbf{x} \rangle + b$ can be trained by the following optimization problem:

$$\underset{\mathbf{w},b}{\operatorname{argmin}} \sum_{i=1}^{m} L(y_i, f(\mathbf{x}_i)) + \lambda \|\mathbf{w}\|^2, \tag{1}$$

where $L(y, f(\mathbf{x}))$ is a loss function. In LS-SVM, the loss function is defined as the quadratic loss $L(y, f(\mathbf{x})) = (y - f(\mathbf{x}))^2$ while in most SVM classifiers, the hinge loss $L(y, f(\mathbf{x})) = \max(0, 1 - yf(\mathbf{x}))$ is used. LS-SVM was shown to have similar performance compared with the hinge loss SVM in many applications in [22].

To extend the classifier to handle non-linear problems, kernel trick is usually used. Kernel trick maps the input feature space into another (usually higher dimensional) feature space by $\phi(\mathbf{x})$, where $\phi(\cdot)$ is the mapping function. Representer theorem [20] states that the solution to (1) can be expressed as a weighted summation of the dot product of the input and support vectors in the mapped feature space:

$$f(\mathbf{z}) = \sum_{i=1}^{m} \alpha_i \kappa(\mathbf{x}_i, \mathbf{z}) + b, \tag{2}$$

where $\kappa(\mathbf{x}_i, \mathbf{z}) = \langle \phi(\mathbf{x}_i), \phi(\mathbf{z}) \rangle$ is the kernel function, $\mathbf{x}_i$ is the i-th support vector and α_i is corresponding coefficient which is to be determined. For quadratic loss LS-SVM, α_i can be obtained by the following closed form [22]:

$$\boldsymbol{\alpha} = (K + \lambda I)^{-1} \mathbf{y}, \tag{3}$$

where $\boldsymbol{\alpha} = [\alpha_1, \alpha_2, \ldots, \alpha_m]^T$, $\mathbf{y} = [y_1, y_2, \ldots, y_m]^T$, K is the kernel matrix with element $K_{ij} = \kappa(\mathbf{x}_i, \mathbf{x}_j)$ and I is the identity matrix that has the same size as kernel matrix K.