Neurocomputing 175 (2016) 310–323

Contents lists available at ScienceDirect

Neurocomputing

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

Regional deep learning model for visual tracking

Guoxing Wu [a,*], Wenjie Lu [b], Guangwei Gao [c], Chunxia Zhao [a], Jiayin Liu [a]

[a] School of Computer Science and Engineering, Nanjing University of Science and Technology (NUST), Nanjing 210096, PR China
[b] Key Laboratory of Ministry of Public Security for Road Traffic Safety, Wuxi 214151, PR China
[c] Institute of Advanced Technology, Nanjing University of Posts and Telecommunications (NUPT), Nanjing 210096, PR China

ARTICLE INFO

Article history:
Received 15 February 2015
Received in revised form
7 August 2015
Accepted 19 October 2015
Communicated by Jiayu Zhou
Available online 30 October 2015

Keywords:
Deep learning
Particle filter
Visual tracking

ABSTRACT

Deep learning has been successfully applied to visual tracking due to its powerful feature learning characteristic. However, existing deep learning trackers rely on single observation model and focus on the holistic representation of the tracking object. When occlusion occurs, the trackers suffer from the contaminated features obtained in occluded areas. In this paper, we propose a regional deep learning tracker that observes the target by multiple sub-regions and each region is observed by a deep learning model. In particular, we devise a stable factor, modeled as a hidden variable of the Factorial Hidden Markov Model, to characterize the stability of these sub-models. The stability indicator not only provides a confidence degree for the response score of each model during inference stage, but also determines the online training criteria for each deep learning model. This online training strategy enables the tracker to achieve more accurate local features compared with those fixed training trackers. In addition, to improve the computational efficiency, we exploit the structurized response property of the customized deep learning model to approximate the final tracking results by the weighted Gaussian Mixture Model under the particle filter framework. Qualitative and quantitative evaluations on the recent public benchmark dataset show that our approach outperforms most state-of-the-art trackers.

1. Introduction

Visual tracking is a fundamental topic in computer vision community as its wide range of practical applications (e.g. autonomous vehicle navigation, video surveillance and motion analysis). Although a plethora of trackers have been proposed in recent years, the tracking problem is still very challenging due to complex real-world environment such as occlusion, severe deformation, abrupt motion, illumination changes and background clutter.

To overcome these negative factors, many tracking methods resort to more complicated observation models [1–5] or motion models [6–8]. However, these methods usually bring problems such as over-fitting and expensive computation, which prohibit them to deal with the tracking task in real-time. One of the promising tracking approaches is the Deep Learning-based Tracker (DLT) [9]. The feature learning scheme in deep learning architecture makes its observation model more robust than most typical trackers [10–13], which are based on hand-crafted features. DLT resorts to transfer learning and online fine-tuning strategy to adapt to appearance variations of the target. However, the iteration number for the fine-tuning training is fixed and set heuristically, which brings a high risk of over-fitting. Besides, the unified online updating scheme for the whole deep learning model usually generates contaminated features when partial occlusion occurs.

In this paper, we present a tracker based on multiple deep learning observation models in a Factorial Hidden Markov Model (FHMM [14]) framework. Specifically, we assume that the state of the target follows the first-order Markov process and observe the target through multiple sub-regions. Each observation model is implemented by a deep learning architecture and focuses on the local feature in its corresponding sub-region. A stable factor is designed to characterize the stability of each observation model, which is updated via the Bayesian formula under the FHMM framework. The stable factor not only provides the confidence degree of each observation model during the inference process of the final tracking result, but also offers the updating criteria for each observation model during the training stage. We build the direct relationship between the stable factor and the iteration number of the fine-tuning stage, which is based on the observation that occlusion region usually leads to unstable performance of the observation model and the online updating scheme for the model should be suppressed to avoid generating contaminated feature. Compared to those permanent updating strategies [9,15], our

* Corresponding author. Tel.: +86 15895820829.
E-mail addresses: leonidwoo@gmail.com (G. Wu), luwenjie0122@msn.cn (W. Lu), zhaochx@mail.nust.edu.cn (C. Zhao), liu.smiton@gmail.com (J. Liu).

http://dx.doi.org/10.1016/j.neucom.2015.10.064
0925-2312/

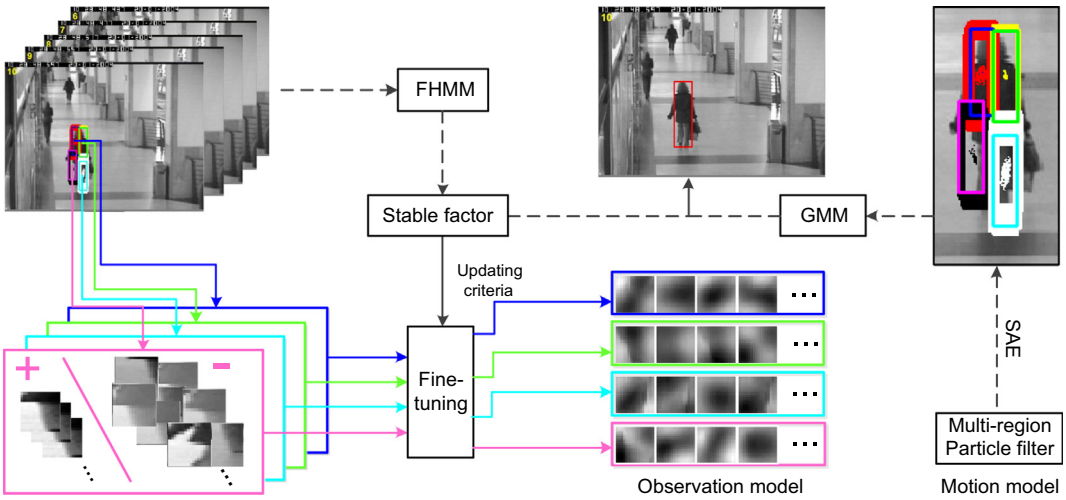

Fig. 1. An overview of regional deep learning tracker. The tracking object is observed by four sub-regions: top-left, top-right, bottom-left and bottom-right. At online training phase (denoted by the solid line), the deep learning observation model is fine-tuned to adapt to the appearance of each sub-region. The training samples are generated in real-time and the training criteria for each model is determined by the stable factor, which is achieved by FHMM. At inference phase (denoted by the dashed line), the trained observation models are evaluated on these regions to get the corresponding response scores. Then, GMM is built upon these response sets. By combining the stable factor and GMM, the final response map is achieved and the position with the highest confidence is considered to be the target. The blurry regions colored with red, yellow, black and white are the particles with high confidence in terms of bounding box. The point clouds are the centers of these particles correspondingly. The rectangles colored with blue, green, pink and cyan are the particles with the highest confidence of each sub-region. (For interpretation of the references to color in this figure caption, the reader is referred to the web version of this paper.)

algorithm reaches a balance between over-fitting and under-fitting. We further exploit the local structurized response property of the customized deep learning model and build a weighted Gaussian Mixture Model (GMM) to approximate the response value of each candidate particle. Compared to the traditional GMM based tracker, our tracker is different mainly in two aspects: (1) our tracker can be deemed as an extension of the GMM, which incorporates the stable factor as a weight for each component; (2) the inference process of the GMM is based on particle filter framework instead of point estimation. Note that, the computational efficiency of our tracker is better than the original DLT [9], because the computational complexity can be reduced significantly by disassembling the entire deep learning network into several shrunken networks. We name our approach as Regional Deep-Learning Tracker (RDLT). Fig. 1 gives an overview of the proposed method.

The rest of this paper is organized as follows. Section 2 summarizes the work most related to ours. Section 3 briefly reviews the particle filter. We elaborate on the proposed method in Section 4. The tracker is evaluated comprehensively and analyzed in detail in Section 5. Section 6 gives the detail discussion about the structure of the proposed method and Section 7 draws conclusion.

2. Related work

There are extensive literatures on visual tracking. The core focus of these literatures can be categorized into three groups: observation model, motion model and search strategy. In this section, the first two groups that are directly related to our method are reviewed. For a comprehensive survey, we refer readers to references [16,17].

Observation model: Observation model can be divided into either generative [18,3,19–28] or discriminant approach [29,30,11, 31,32,10,9]. Generative tracking algorithms pursue a robust representation model for the target and the tracking result is determined by the minimal reconstruction error among the candidate set. Adam et al. [19] use fragment approach combined with integral histogram to characterize the target appearance. Since there is no updating scheme for the observation model, the tracker cannot deal with large appearance changes. Subsequently, a dynamic observation model [18] is proposed to adapt to the target appearance variation. To enhance the robustness of the tracker, Kwon and Lee [3,4] leverage on the sparse principle component analysis technique and decompose the observation model into multiple basic observation models. Although it performs well under various illumination conditions, the method is less effective when coping with heavy occlusion, as it is limited by its holistic observation model. Based on the advanced progresses on sparse representation, $l1$ tracker [20] proposes that the original tracking problem can be cast into the linear regression problem with sparse coefficient constraint. Due to the high computational complexity of $l1$ tracker, various extensions have been made to improve the computational efficiency in the aspect of either optimization algorithm [22,27] or search strategy [21]. In [23] and [33], patch-based observation models are developed by exploiting spatial information, which concentrate on the local feature representation and show impressive performance in the occlusion scenario. Compared to these patch-based methods, the proposed tracker makes full use of spatio-temporal information via FHMM and focuses more on the relationship of multiple sub-regions. In addition, instead of learning the appearance of the tracking object directly, the proposed tracker puts more emphasis on the basic feature learning problem to adapt to the appearance variance.

Discriminant trackers cast the tracking problem as a binary classification task. The location with the highest response score of discriminant function is considered to be the tracked target. Avidan [29] uses the support vector machine classifier to select appropriate features for object tracking. The similar idea is adopted in [30] where the classifier is the online AdaBoost. To alleviate the drift problem, multiple instance learning is employed in [10] for visual tracking. Hare et al. [32] demonstrate that the tracker can be robust to the wrong labeled samples by using structured output support vector machine. The P–N tracker [11] builds the Tracking–Learning–Detection (TLD) framework to cope with the missing case of tracking. Zhang et al. [34,35] propose an extremely efficient method by extracting the feature from the compressed domain. DLT [9] shows promising results by applying deep-learning model to visual tracking, while particular computer