DEEP CONVOLUTIONAL PARTICLE FILTER FOR VISUAL TRACKING

Reza Jalil Mozhdehi and Henry Medeiros
{reza.jalilmozhdehi, henry.medeiros}@marquette.edu
Department of Electrical and Computer Engineering, Marquette University, Milwaukee, WI, USA

ABSTRACT

This article proposes a novel framework for visual tracking based on the integration of a deep convolutional neural network (CNN) and a particle filter. In the proposed framework, the position of the target at each frame is predicted by a particle filter according to a motion model. Particles around the predicted position are then used as input to the HCFT CNN-based tracker which adjusts their positions to the most likely target positions. The weights of the particles are then determined using the correlation map of the CNN tracker. Finally, the particles and their weights are used to calculate the position of the target in the current frame. We evaluated the performance of the proposed framework using the Visual Tracker Benchmark v1.0. Our results show that this method improves the performance of HCFT in challenging attributes such as deformation, illumination, out-of-plane and in-plane rotations, as well as overall performance.

Index Terms— Deep Convolutional Neural Network, Particle Filter, Visual Tracking, Visual Tracker Benchmark

1. INTRODUCTION

Visual target tracking is a challenging computer vision problem, particularly in situations such as occlusions, target deformations, and in-plane or out-of-plane rotations. Although recent visual tracking approaches based on particle filters such as the Firefly algorithm [1], or those based on correlation filters such as [2, 3] showed acceptable results, they are not as effective as trackers based on convolutional neural networks (CNN). Recently, a powerful visual tracker named Hierarchical Convolutional Feature Tracker (HCFT) has been proposed by Ma et al. in [4]. The tracker employs a deep convolutional neural network and showed substantial performance improvement in comparison with other visual trackers such as MEEM [5], KCF [6], Struck [7] and TLD [8]. The deep convolutional neural network employed in HCFT is considered as a pyramid in which the earlier convolutional layers are used to extract spatial information and the later layers are employed to obtain semantic information [4]. A correlation filter is used on the output of each of the convolutional layers. Then, in a coarse-to-fine manner and by moving back from the semantic information to the spatial information, the exact position of the target is determined [4].

In this article, a particle filter is integrated with HCFT to improve its performance in critical situations such as deformation, illumination, out-of-plane and in-plane rotations. The position of the target at the current frame is utilized in conjunction with a target motion model to predict its position in the next frame. Afterwards, particles are generated around the predicted position and used as inputs to HCFT, which then refines their predicted locations. Also, our framework utilizes the output of the convolutional neural network and correlation filter, which we henceforth call a feature map, to determine the weights of the particles. Finally, the position of the target for the current frame is calculated based on the particles and their weights.

2. RELATED WORK

After being successfully utilized in object detection tasks [9], convolutional neural networks (CNN) have more recently been employed in visual tracking. Several different CNN structures have been proposed so far. Li et al. presented a CNN tracker with only two convolutional layers, which uses three different image cues for each image patch as inputs to the CNN [10]. The output of the CNN is connected to two fully connected layers, which uses three different image cues for each image patch as inputs to the CNN [10]. The output of the CNN is connected to two fully connected layers, which are responsible for estimating the position of the target [10]. Li et al. also proposed a new tracker, which has the same convolutional layers as in [10] but uses a different training model for the two fully connected layers [11]. In the new tracker, by considering different lifespans on every data sequence, different sub-sequences are made. These sub-sequences with different cues are then used for training different tasks [11]. Another interesting CNN tracker is MDNet [12], which shares three convolutional layers for all domains and assigns a specific layer to each domain.

Unlike CNNs, particle filters have been widely employed in visual tracking for several years. Li et al., for example, proposed a visual tracker based on a toadown visual attention computational model and the particle filter [13]. The proposed toadown visual attention detects target-related salient regions. Then, the salient regions are sent to the particle filter to determine the position of the target [13]. Gao et al. presented an algorithm named Firefly in which the number of
meaningful particles was dynamically increased to improve tracking performance [1]. Another visual tracker was proposed in [14] based on a novel iterative particle filter (IPF) which iteratively samples the particles with the search scope contracted. Additionally, Kim et al. employed the particle filter in conjunction with a target appearance model based on a Gaussian mixture [15].

Recently, visual tracking methods combining particle filters with deep neural networks such as stacked de-noising autoencoders [16] or a two-layer CNN [17] have been proposed. However, unlike the framework proposed in this paper, these approaches do not take into consideration the possibility of utilizing correlation filters to determine the likelihood of the observations.

3. STRUCTURE OF HCFT

The CNN used in HCFT was originally proposed by Ma et al. in [18] for object detection. It includes five convolutional layers and five pooling layers. The outputs of the different layers of HCFT for a specific frame of the motor-rolling data sequence are illustrated in Fig. 1. The deconvolutional neural network proposed by Zeiler et al. in [19] clarified that layers 1 and 2 respond to edges, layer 3 recognizes similar textures, layer 4 illustrates significant variation, and layer 5 shows entire objects with significant pose variation. Thus, semantic information of the target is extracted from the later layers of the CNN and spatial details are obtained from the early layers. HCFT uses the correlation filter over the outputs of each convolutional layer. Then, by applying a coarse-to-fine method it moves back from the fifth layer to the third layer to determine the exact position of the target. Fig. 1 also shows the coarse-to-fine method applied in [4].

4. PARTICLE FILTER DESIGN

The Particle filter (PF) is a sequential Monte Carlo method that employs a set of random weighted samples called particles to represent the posterior distribution of the target state [20]. The filtering posterior distribution of the target state at time $t$ is then given by

$$
\hat{P}_r(x(t) | Y_t) \approx \sum_{i=1}^{M} W_i(t) \delta(x(t) - x_i(t))
$$

(1)

where $x_i$ represent the samples, $M$ the number of particles and $W_i$ the normalized particle weights. $x(t)$ is the target state, which is given by

$$
x(t) = [u(t), v(t), \dot{u}(t), \dot{v}(t)]^T
$$

(2)

where $u(t)$ and $v(t)$ are the locations of the target on the horizontal and vertical image axes and $\dot{u}(t)$ and $\dot{v}(t)$ are the corresponding velocities. In the proposed framework, the previous target state $x(t-1)$ is given to the motion model to calculate the predicted position $\hat{x}(t)$. The motion model is given by

$$
\hat{x}(t) = Ax(t-1) + q(t)
$$

(3)

where $q(t)$ is the process noise and $A$ is the process matrix defined by

$$
A = \begin{bmatrix}
1 & 0 & 1 & 0 \\
0 & 1 & 0 & 1 \\
0 & 0 & 1 & 0 \\
0 & 0 & 0 & 1
\end{bmatrix}
$$

(4)

The predicted target position is then disturbed by additive Gaussian noise to generate an initial set of particles $\tilde{x}_i$. These particles are then used as inputs to the HCFT tracker to generate a new set of particles $x_i$. The weights, $w_i$, of the particles $x_i$ are given by the sum of all the elements of the feature
map corresponding to that particular image patch. The intuition behind this choice is that feature maps that correspond to the target tend to show substantially higher correlation values than background patches [21]. We then compute the normalized weights $W_i(t)$ which are used to update the target posterior [20]. At every iteration of the filter, we compute the effective sample size $\hat{N}$ and compare it with a fixed threshold $N_{thr}$ to determine if resampling should be carried out. The effective sample size is given by [20]

$$\hat{N} = \frac{1}{\sum_{i=1}^{M} w_i^2(t)}$$ (5)

If $\hat{N} \leq N_{thr}$, then resampling is performed. Finally, the position of the target for the current frame is estimated according to [20]

$$x(t) \approx \sum_{i=1}^{M} W_i(t)x_i(t)$$ (6)

Algorithm 1 summarizes the proposed visual tracker.

5. RESULTS AND DISCUSSION

We evaluate our algorithm using the well known Visual Tracker Benchmark v1.0 [22], which contains 50 data sequences that are annotated with 9 attributes representing challenging aspects of tracking, such as occlusions, deformations, and illumination variations. It benchmarks trackers against a one-pass evaluation (OPE), a spatial robustness evaluation (SRE), and a temporal robustness evaluation (TRE). In the OPE metric, the tracker is initialized with the ground truth location at the first frame of the image sequence while in the SRE metric, the initialization is subject to some disturbance. TRE focuses on short-term tracking, and since the main benefit of employing our proposed method is to extend the long-term tracking ability of the HCFT algorithm, we only present results for the OPE and SRE metrics. The improvements with respect to the TRE metric are negligible, as expected. For additional details on the benchmark procedure, we refer the reader to [22].

Fig. 2 shows a qualitative illustration of the performance of our tracker in comparison with HCFT on some sequences in which HCFT fails. The three sequences on the left show OPE results and the three sequences on the right illustrate SRE results. As the sequences indicate, the baseline tracker gets easily confused in situations such as deformation, occlusion, blurring, and out-of-plane rotations. The particle filter is able to sample several image patches and it is hence capable of overcoming these difficulties.

Fig. 3 shows a quantitative evaluation of the performance of the proposed approach in comparison with HCFT for the attributes in which our approach shows the most significant improvement. In the figure, precision plots correspond to the average Euclidean distance between the tracked locations and the ground truth while success plots correspond to the amount of overlap between the target bounding box and the corresponding ground truth [22]. As the figure shows, the proposed framework improves the performance of HCFT on several attributes without degrading its performance on other attributes not shown in the figure. In attributes such as temporary deformation, illumination, in-plane and out-of-plane rotations in which the correlation filter loses track of the target, the motion model allows the successful prediction of the position of the target and the particles are then able to recover using the weights generated by the CNN. Under challenging conditions that include deformation, illumination variation, out-of-plane and in-plane rotations, our method shows improvements of approximately 7.5%, 4.5%, 4% and 3.5%, respectively. The overall OPE success rate improvement is
approximately 3.5%.

6. CONCLUSION

This article proposes a novel framework for visual tracking based on the integration of a deep convolutional neural network and a particle filter. In the proposed framework, a motion model predicts the position of the target at each frame and the HCFT CNN-based tracker is used for refining the particle positions and generating the corresponding weights. We evaluated the performance of the proposed framework using the Visual Tracker Benchmark v1.0 and the results show that our method improves the performance of HCFT in challenging conditions such as deformation, illumination, in-plane and out-of-plane rotations. The motion model in conjunction with the particle filter’s ability to sample several image patches allow it to overcome temporary target losses caused by dramatic temporary appearance changes or occlusions.

Sequential Monte Carlo methods have been used to incorporate temporal information and hence develop robust object tracking algorithms that build on features that range from simple color histograms to support vector machines [23, 24]. It is only natural to integrate such temporal robustness with recent state-of-the-art feature extraction techniques such as correlation filter based deep convolutional neural networks. Considering the progress observed over the past several decades in the area of recursive Bayesian estimation and the recent significant machine learning breakthroughs made possible by deep learning techniques, we believe this work is just the first step in that direction.
7. REFERENCES


