Efficient convolution operator for tracking

Lina Zhu

29 May 2018

1 introduction

In recent years, the method of discriminating correlation filters (DCF) has greatly improved the status quo in tracking. However, in the pursuit of ever-increasing tracking performance, feature speed and real-time capabilities gradually faded. In addition, more and more complex models, a large number of trainable parameters, have introduced serious over-risk. In this work, we address the key reasons behind the problem of computational complexity and overfitting, the goal being to increase both speed and performance. Universal vision tracking is one of the basic issues in computer vision. This is the goal of any image sequence in the estimated trajectory, and online visual tracking plays a vital role in many aspects of real-time vision applications such as intelligent surveillance systems, autopilot, drone surveillance, intelligent traffic control and man-machine interface. With the on-line nature of the tracking, the ideal tracker should be an accurate and robust real-time vision system with

difficult computational constraints.This complex and large model has introduced a serious risk of overfitting (see Figure 1). In this article, we address the issue of over-assembly in recent DCF trackers while restoring its iconic real-time capabilities.
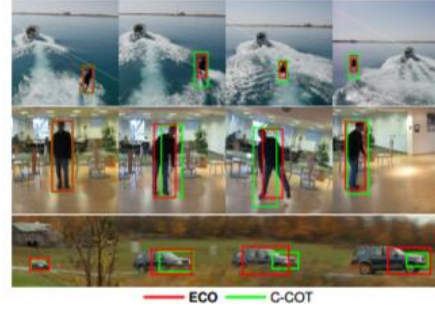

Figure 1: A comparison of our approach ECO with the baseline C-COT [1] on three example sequences.

2 Experiment procedure

2.1 Implementation details

Our tracker is implemented in Matlab. We apply this same feature representation as C-COT, namely the combination The first (Conv-1) and last

(Conv-5) convolutional layers in the VGG-m network, and HOG [6] and Color Name (CN) [2]. Summarize the settings of each function in Table 1. We use the same number of gradient iterations in NCG = 5 conjugates such as C-COT. Please note that the settings for all videos in all parameter datasets remain unchanged.

Table 1: The settings of the proposed factorized convolution approach,as employed in our experiments. For each feature, we show the dimensionality D and the number of filters C.

	Conv-1	Conv-5	HOG	CN
Feature dimension, D	96	512	31	11
Filter dimension, C	16	64	10	3

2.2 Baseline comparison

The experiment shows our contribution analysis. Integrate our decomposed convolution into baseline cues to improve performance and significantly reduce complexity. The sample space model further improved EAO's relative return of 2.9% while reducing the learning complexity by 8 times. In addition, we combined our proposed model update to propose our EAO score of 0.374, resulting in a final relative return of 13.0% higher than the benchmark value. we also show the impact of our contribution on tracker speed. For a fair comparison, we reported that the FPS used to measure one CPU is used for all entries in the table and no billing is required for feature extraction time. Our system contribution increased the speed of the tracker and combined it with the baseline, the final revenue increased by 20 times. Including all steps, GPU version Our tracker runs at 8 FPS.

3 conclusion

We revisit the core DCF formulation to solve the problem of overfitting and computational complexity. We introduce a decomposition convolution operator to reduce the number of parameters in the model. We also propose a compact training sample distribution model to significantly improve the memory and time complexity of learning reduction while enhancing sample diversity. Finally, we propose a simple and effective model updating strategy to reduce the overfitting of recent samples. Experiments on four data sets demonstrate the most advanced performance with improved frame rates.

References

[1] M. Danelljan, A. Robinson, F. Shahbaz Khan, and M. Felsberg. Beyond correlation filters: Learning continuous convolution operators for visual tracking. In *ECCV*, 2016. 1

[2] J. van de Weijer, C. Schmidand, J. J. Verbeek, and

D. Larlus. Learning color names for real-world applications. *TIP*, 18(7):1512–1524, 2009. 2