Efficient Convolution Operator for Tracking

创建时间： *2017/9/23 18:47*
作者： *XY.Wang*

Efficient Convolution Operator for Tracking

(更新中...)以下均为个人理解，如有错误，欢迎指正！

Abstract

DCF及其相关算法的缺陷：

> 1. 使用CNN特征，卷积过程耗时，实时性不好
> 2. 越来越复杂的模型，使得需要训练的参数变得非常多，容易造成过拟合

ECO针对**计算复杂度**和**过拟合**这两个问题，提出：

> 1. 一个被因式分解的卷积运算： 很大程度上减少了DCF模型中的参数
> 2. 一个用于生成训练样本分布的模型： 降低了存储和时间复杂度，同时提供更高质量的多样性样本
> 3. 一个保守的模型更新策略： 具有更好的鲁棒性和更低的复杂度

与C-COT相比：速度是20倍，EAO增长13%；
使用hand-crafted特征(OTB2015)：AUC=65%

Introduction

> 第一段：视觉跟踪概念+应用于哪些领域+两个评价标准**精确度和鲁棒性**
>
> 第二段：从DCF引出ECO
>
> (1) DCF给跟踪算法带来哪些改善？优点是什么？
>
> (2) 改进历程（大量引用参考文献）
>
> (3) 仍然存在的缺陷
>
> 第三段：为什么会存在以上缺陷
>
> 总结DCF的改进主要归功于**强大的feature（CNN使得速度降低）**和**精致的学习公式**-->model更大更复杂（search reagion变大）-->参数更多-->容易造成过拟合

造成复杂计算和过拟合的原因

> （1）**模型大小：** 高维特征图组合在一起（例如CNN的多层特征），滤波器与特征图尺寸相同，造成需要学习的滤波器参数也非常多；
>
> （2）**训练样本集的大小：** 迭代一次，滤波器参数更新一次。迭代次数越高，学习到的参数越精确，因此需要大量的训练数据，然而又由于存储空间受限，所以选择丢弃老的样本，但只用新样本会造成跟踪器和近期变化的外观拟合的好，并不代表能够精确跟踪最初的目标，可能发生跟丢的现象；

> （3）**模型更新：** 过往的DCF都是连续每帧都对滤波器参数进行更新，但这样的更新方案对目标一点点的外观变化都非常敏感，很容易过拟合。且每帧都更新降低了速度，过拟合造成鲁棒性降低。

Main Idea of ECO

A. 因式分解滤波器，达到降维的目的，减少了需要学习的滤波器参数，降低了CG迭代算法的时间复杂度

> 将滤波器和特征的卷积公式进行因式分解，
>
> $$S_f\{x\} = f * J\{x\} = \sum_{d=1}^{D} f^d * J_d\{x^d\}. \qquad (2)$$
>
> 具体过程为： 先把原来的D个通道的滤波器分解成了一个D乘C的矩阵P和C个通道的滤波器的乘积,即{f1,f2,...,fD}=P{f1,f2,...,fC}。这样就变成了下式
>
> $$S_{Pf}\{x\} = Pf * J\{x\} = \sum_{c,d} p_{d,c} f^c * J_d\{x^d\} = f * P^{\mathrm{T}} J\{x\}. \qquad (6)$$
>
> 这样做的意义在于：原来需要学习D维的滤波器，因式分解以后，只需要学习C维的滤波器和一个D*C的矩阵P就可以了。其中矩阵P在第一帧就可以学习得到。 经过一些列变换，学习过程又变成了利用CG方法求解一个线性的最小二乘问题。

B. 通过高斯混合模型对训练样本进行分类，增加了样本多样性，防止过拟合

DCF模型的训练样本

每一帧更新：新的一帧检测完毕以后，提取一个单个的训练样本，加到训练样本集中。更新时采用就近原则，丢弃老旧样本，选择最近的一些样本进行训练。这样做的缺点是：相邻帧的目标外观非常相似，造成训练样本集的信息冗余，且容易造成与最近的目标拟合的很好，和久远的目标拟合的不好的现象。

> $$E(f) = \sum_{j=1}^{M} \alpha_j \left\| S_f\{x_j\} - y_j \right\|_{L^2}^2 + \sum_{d=1}^{D} \left\| w f^d \right\|_{L^2}^2 . \quad (3)$$

ECO的训练样本：

For this purpose we employ a Gaussian Mixture Model (GMM) such that $p(x) = \sum_{l=1}^{L} \pi_l \mathcal{N}(x; \mu_l; I)$. Here, L is the number of Gaussian components $\mathcal{N}(x; \mu_l; I)$, π_l is the prior weight of component l, and $\mu_l \in \mathcal{X}$ is its mean. The covariance matrix is set to the identity matrix I to avoid costly inference in the high-dimensional sample space.

To update the GMM, we use a simplified version of the online algorithm by Declercq and Piater [14]. Given a new sample x_j, we first initialize a new component m with $\pi_m = \gamma$ and $\mu_m = x_j$ (*concatenate* in [14]). If the number of components exceeds the limit L, we *simplify* the GMM. We discard a component if its weight π_l is below a threshold. Otherwise, we merge the two closest components k and l into a common component n [14],

$$\pi_n = \pi_k + \pi_l \quad , \quad \mu_n = \frac{\pi_k \mu_k + \pi_l \mu_l}{\pi_k + \pi_l}. \tag{11}$$

所以，本文**利用高斯混合模型（GMM）对我们的样本集进行分类**，得到 L 个不同的components,每一个组件都有一个权重和一个特征均值，用它们分别替代原来公式的样本权重和样本。这样就将训练过程中使用的样本数从M减少到了L（节省了存储空间），且增加了样本多样性（不同类别的样本均值代替）。

$$E(f) = \sum_{l=1}^{L} \pi_l \| S_f\{\mu_l\} - y_0 \|_{L^2}^2 + \sum_{d=1}^{D} \| w f^d \|_{L^2}^2 . \tag{12}$$

components的更新策略：每帧更新

新的一帧到来时，若components的个数超过 L，则首先用新样本建立一个独立的组件，然后：如果权值最小的组件的权重小于阈值，则丢弃；否则合并权重最小的那两个组件。

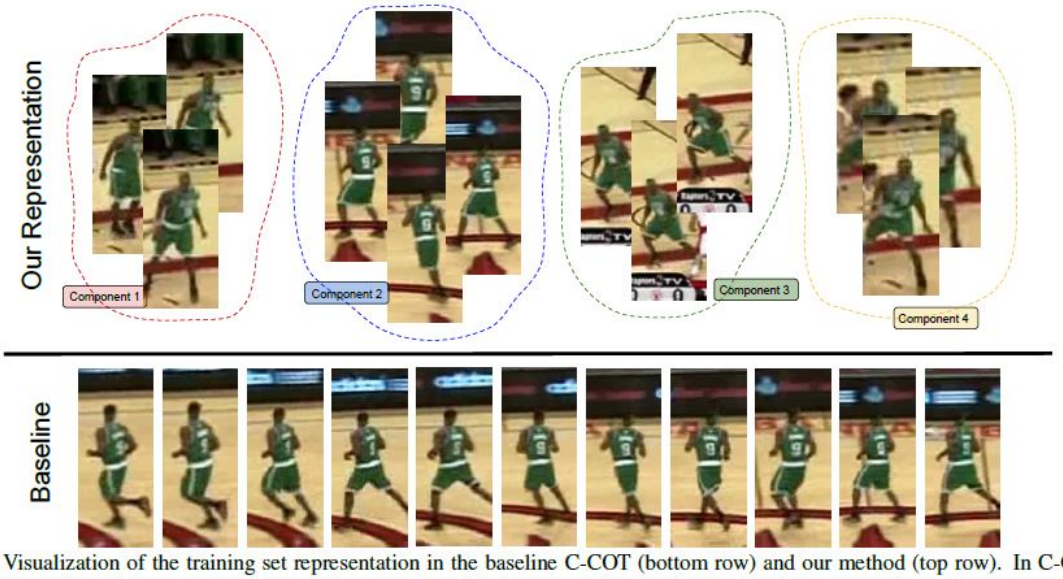

Figure 3. Visualization of the training set representation in the baseline C-COT (bottom row) and our method (top row). In C-COT, the training set consists of a sequence of consecutive samples. This introduces large redundancies due to slow change in appearance, while previous aspects of the appearance are forgotten. This can cause over-fitting to recent samples. Instead, we model the training data as a mixture of Gaussian components, where each component represent a different aspect of the appearance. Our approach yields a compact yet diverse representation of the data, thereby reducing the risk of over-fitting.

两种训练样本的对比：

原来的DCF模型每一帧更新：新的一阵检测完毕后，首先提取一个新样本，添加到训练样本中，然后选择最近的N个样本对滤波器参数进行学习和更新，作为下一帧的迭代初始值；

而ECO选择不那么频繁的更新策略：每Ns（约等于5帧）进行更新。当新的一帧检测完毕后，首先提取新样本，用于更新components构成的训练样本。当间隔时间达到5帧时，利用components构成的训练样本对滤波器参数和矩阵增量进行学习和更新。

相对于之前的训练样本，新增的样本是一个mini-batch而不再只有一个单个的样本，这会使训练出来的滤波器参数更加稳定：因为DCF中，如果新增的单个样本发生突变时，也会用于训练，但这样训练结果将不够准确了。

C. 更新策略：不那么频繁，防止过拟合，鲁棒性更好

滤波器的更新：

固定帧数 Ns(=5)更新

components的更新：

每一帧都提取新样本，更新components