

# 简介

提出了一种新的在测量方法来衡量target和exemplar的距离，获取多个具有代表性的exemplar。这个方法不需要训练，而且比较efficient，不会降低太多速度。

本文的主要目的在于讨论如何提取exemplar来最大化追踪的准确性。

作者认为提取好的exemplar可以有效解决遮挡、明暗变化、运动模糊和形状变化等问题。

作者提出两个模块，Long-term module(LTM)和short-term module(STM)，其中LTM主要代表物体在不同环境下(光线，形状等)，用于物体长期追踪和re-detect。

STM的作用主要是利用短期的特征进行追踪。

## Long-term Module

这个模块的目的是存储一些代表性的exemplar，能够最大化追踪物体的多样性。

首先作者比较多样性是在卷积后的隐空间进行比较，也就是Siamese网络所提取的特征空间。

### Allocation Strategy

从数学的角度，LTM的目的是最大化超平行体的体积  $\Gamma(f_1, \dots, f_n)$ 。其中的特征向量  $f_i$  代表对应的 template  $T_i$ 。

其中  $T_1$  代表template在特征空间交叉相关的conv kernel。

因为孪生网络实际是在特征空间里通过卷积来衡量template kernel  $f_1$  和当前search image区域的相似性。

因此作者认为如果要衡量两个template T1和T2的相似性，可以计算  $f_1 * f_2$ ，对于内存中所有的 template，可以构造出一个Gram矩阵：

$$G(f_1, \dots, f_n) = \begin{bmatrix} f_1 * f_1 & f_1 * f_2 & \cdots & f_1 * f_n \\ \vdots & \vdots & \ddots & \vdots \\ f_n * f_1 & f_n * f_2 & \cdots & f_n * f_n \end{bmatrix} \quad (1)$$

gram矩阵的行列式可以代表基于  $f_1, \dots, f_n$  的超平行体的体积  $\Gamma$ 。

$$\max_{f_1, f_2, \dots, f_n} \Gamma(f_1, \dots, f_n) \propto \max_{f_1, f_2, \dots, f_n} |G(f_1, f_2, \dots, f_n)| \quad (2)$$

如果某个template替换了memory其中的一个template导致体积更大，那么将这个template当成一个basis vector of the feature space，代表追踪物体在当前embedding space的流形。

### lower bound

这个策略可能会导致后期LTM都是一些和target无关的物体，所以需要有一个分数的下线。这个

lower bound是候选template  $T_c$  和gt的template  $T_1$ ，因此需要满足条件  $f_c * f_1 > l \cdot G_{11}$

然后有时候静态的boundary过于保守，作者又采取了两个策略：

1. 动态的lower bound，主要是考虑到短时间物体外表变化引入一个diversity measure  $\gamma$  给STM，因此有  $f_c * f_1 > l \cdot G_{11} - \gamma$

2. ensemble lower bound。bound是静态的，但是相似性的计算会用内存中所有的template进行计算。这样valid template满足  $f_c * f_{1:n} > l \cdot \text{diag}(G)$

## Template Masking

在计算特征的相似性之前，会对feature vector  $f_i$  计算一下tapered cosine window来减少背景的影响。

## Short-term Module

这个模块的作用主要是处理突然的运动变化和遮挡(abrupt movements and partial occlusion)。

这个模块的元素以队列的方式储存，总容量为一个固定值  $K_{st}$

这个模块元素的判别标准如果直接沿用LTM的判别标准效果并不好，会有非常大的波动，而且没有一个normalize的标准。

因此其多样性的计算标准为：

$$\gamma = 1 - \frac{2}{N(N+1)G_{st,max}} \sum_{i < j}^N G_{st,ij}$$

计算Gram矩阵的上三角之后然后根据其最大值进行normalize。这样得到的  $\gamma$  取值范围为[0-1]。越接近1那么越能代表STM元素的多样性(diverse)。

## Inference Strategy

为了得到精确的bounding box，主要有两个模块：

1. Modulation module用于平衡STM和LTM的信息。
2. ST-LT Switch。默认情况下是一直用STM模块的template去做追踪但是因为STM模块的templat经常变化很容易导致drift。因此作者会计算IoU(STM, LTM)，如果这个值小于一个阈值就认为STM的结果产生了较大的偏移，因为作者认为LTM更加稳定。

## Implementation Details

为了使得内存更新更快运行速度更快采取了两个策略：

1. parallelization: 首先内存更新的时候需要计算每个template和内存元素的相似性来构造gram矩阵。由于这个计算其实是2D conv，所以是可以在GPU里进行并行计算的
2. dilation: 由于相邻帧之间的template相似性其实是很高，所以没必要每一帧都计算，可以设置一个间隔值dilation，文章中dilation=10，每隔10帧更新一次STM和LTM。

## 实验

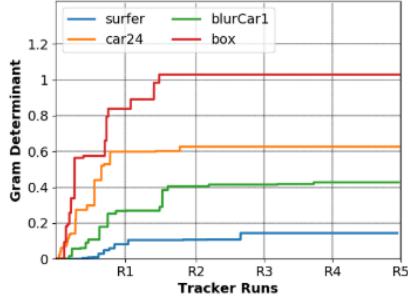
实验主要为了证明以下四点：

1. Gram矩阵的行列式在同一个seq上是否能逐渐增加直到收敛

2. THOR对本身tracker影响的速度如何
3. THOR是能提高tracker的performance
4. 每个模块对于最后performace的影响

作者用了一些sota的tracker和一些之前的tracker即你想那个了实验，具体的，超参数也比较多，具体可以看原文。

## Gram矩阵能否收敛



	R1		R5	
	$ G_{norm} $	AUC	$ G_{norm} $	AUC
surfer	0.0812	0.6169	0.1441	0.6562
car24	0.5983	0.8299	0.6258	0.8312
blurCar	0.2680	0.8125	0.4273	0.8141
box	0.8377	0.5837	1.0286	0.7542

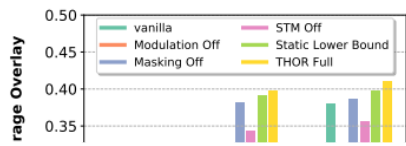
Figure 3: **Proof of Concept.** Left: Convergence of the Gram determinant after repetitively re-running the tracker with THOR. Right: Gram determinant and area under curve evaluated (AUC) at the end of first and last runs ( $R_1$  and  $R_5$ ) of the experiment.

## 速度和精度

Tracker	Lower Bound	VOT2018				OTB2015		
		Accuracy $\uparrow$	Robustness $\downarrow$	EAO $\uparrow$	Speed (FPS) $\uparrow$	AUC $\uparrow$	Precision $\uparrow$	Speed (FPS) $\uparrow$
SiamFC	—	<b>0.5194</b>	0.6696	0.1955	<b>219</b>	0.5736	0.6962	<b>214</b>
THOR-SiamFC	dynamic	0.4977	0.4448	0.2562	99	<b>0.5990</b>	<b>0.7347</b>	97
THOR-SiamFC	ensemble	0.4846	<b>0.3746</b>	<b>0.2672</b>	69	0.5971	0.7291	80
SiamRPN	—	<b>0.5858</b>	0.3371	0.3223	<b>133</b>	0.6335	0.7674	<b>137</b>
THOR-SiamRPN	dynamic	0.5818	0.2341	<b>0.4160</b>	112	<b>0.6477</b>	<b>0.7906</b>	106
THOR-SiamRPN	ensemble	0.5563	<b>0.2248</b>	0.3971	105	0.6407	0.7867	110
SiamMask	—	<b>0.6096</b>	0.2810	0.3804	<b>95</b>	0.6204	0.7683	<b>97</b>
THOR-SiamMask	dynamic	0.5891	0.2388	0.3846	60	<b>0.6397</b>	0.7900	78
THOR-SiamMask	ensemble	0.5903	<b>0.2013</b>	<b>0.4104</b>	70	0.6319	<b>0.7929</b>	66

Table 1: **Tracking benchmarks.** The attained performances of the trackers on VOT2018 and OTB2015. The main metric for ranking trackers is EAO (expected average overlay) on VOT2018, and AUC (area under curve) on OTB2015.

## ablation study



	lower bound	
	dynamic	ensemble
Mean of $ G_{norm}  \uparrow$	0.0261	<b>0.25164</b>