**在线目标跟踪：一种评估基准**

吴毅 Jongwoo Lim 杨明玄

**摘要**

目标跟踪是计算机视觉大量应用中的重要组成部分之一。近年来，尽管在分享源码和数据集方面的努力已经取得了许多进展，开发一套库和标准用于评估当前最先进的跟踪算法仍然是极其重要的。在简单回顾近年来在线目标跟踪的研究进展后，我们以多种评价标准进行了大量的实验，用于研究这些算法的性能。为了便于性能评估和分析，测试图片序列分别被标注了不同的特性。通过定量分析结果，我们得出了实现鲁棒性跟踪的有效方法，并给出了目标跟踪领域潜在的未来研究方向。

1. **介绍**

在计算机视觉的许多不同应用之中，目标跟踪是最重要的组成部分之一。比如监视、人机交互和医疗图像[60,12]。在视频的一帧中，目标的初始状态（比如位置和尺寸）是给定的，跟踪的目的就是在随后的帧中评估出目标的状态。尽管目标跟踪问题已经被研究了数十年，近年来也取得了许多进展[28, 16, 47, 5, 40, 26, 19]，但它仍是一个非常具有挑战性的问题。大量因素影响着跟踪算法的性能，比如光照变化、遮挡和杂乱的背景，目前也不存在单一跟踪方法可以成功地处理所有的应用场景。因此，评估这些最先进的跟踪器的性能以展示其优势和弱点是至关重要的，而且有助于明确该领域未来的研究方向，从而设计出更加鲁棒的算法。

对于综合性能评估，收集具有代表性的数据集是极其重要的。目前已经存在一些针对监控场景中视觉跟踪的数据集，比如VIVID [14]，CAVIAR [21]和PETS数据集。然而，这些监控序列中的目标对象通常是人或小尺寸的车，并且背景通常都是静态的。尽管有一部分普通场景下的跟踪数据集[47, 5, 33]被标定好了边界框，但是绝大部分并没有被标定好。对于那些没有被标注ground truth的序列，评估跟踪算法是困难的，因为结果是基于不一致的目标标注位置。

近年来，更多的跟踪源码已经可以公开获得，比如OAB [22]， IVT [47]，MIL [5]，L1 [40]和TLD [31]算法，这些算法通常都被用于评估。然而，大部分跟踪器的输入输出的格式是不同的，因此对于大规模的性能评估是不方便的。在本工作中，我们建立了一个代码库和一个测试数据集，其中代码库包括了大部分公开可得到的跟踪器，测试数据集标注了ground truth以有助于评估工作。另外，数据集中的每一个序列都被标记了多种特性，比如遮挡、快速运动和光照变化，这些特性通常会影响跟踪的性能。

评估跟踪算法时一个常见的问题是，结果报告仅仅基于少量不同初始条件和参数的序列。因此，这些结果并未体现算法的整体性能。为了公平和综合的性能评估，我们提出从空间和时间上扰乱真实目标位置的初始状态。尽管初始化对鲁棒性的重要性在跟踪领域是众所周知的，但是在文献中却很少被提及。据我们所知，本文是首次提及并分析目标跟踪中初始化问题的综合性工作。我们使用基于位置误差度量的精确度图和基于重叠度量的成功率图来分析每一个算法的性能。

本工作的贡献体现在以下三个方面：

**数据集** 我们建立了一个跟踪数据集，其中包含50个完全标注好的序列以便于跟踪算法的评估。

**代码库** 在代码库里，我们整合了大部分公开可获得的跟踪器，并且统一了输入输出格式以便于大规模的算法性能评估。目前，代码库已经包含了29个跟踪算法。

**鲁棒性评估** 跟踪中的初始边界框在时空上被抽样用于评估跟踪器的鲁棒性和特点。每一个跟踪器通过分析超过660000个边界框输出结果被全面地评估。

本研究工作主要专注于单目标的在线[[1]](#footnote-1)跟踪。代码库、标注好的数据集和所有的跟踪结果在本网站上可以获得http://visualtracking.net。

1. **相关研究**

在本部分，我们从几个主要的模块上来回顾近年来的目标跟踪算法：目标表示方式、搜索机制和模型更新。另外，一些基于联合几个跟踪器或者挖掘上下文信息的跟踪方法也已经被提出来了。

**表示方式** 在任何视觉跟踪器中，目标表示都是一个主要的组成部分，并且大量的方法已经被提出[35]。由于Lucas和Kanade的开创性研究[37, 8]，全局模板（原始灰度值）在跟踪中已经被广泛地使用[25, 39, 2]。随后，基于子空间的跟踪方法[11, 47]被提出来，它能够更好地反映表观变换。此外，梅和凌[40]提出了基于稀疏表示的跟踪方法来处理损坏的目标外观，并且这一研究最近已经被进一步完善了[41, 57, 64, 10, 55, 42]。除了模板，许多其他视觉特征也已经被用于跟踪算法，比如颜色直方图[16]，方向梯度直方图（HOG）[17, 52]、协方差区域描述子[53, 46, 56]和类哈尔特征[54, 22]。最近，判别模型被广泛用于跟踪[15, 4]，其通过在线学习的二分类器将目标从背景中辨别出来。大量学习方法已经被改造用于处理跟踪问题，比如SVM[3]，结构化输出SVM[26]，排序式SVM[7]，boosting算法[4, 22]，semiboosting[23]和多例boosting[5]。为了使跟踪器对姿态变化和部分遮挡更加鲁棒，可以对目标进行分块，每一个小块用描述子或者直方图进行描述。[1]中使用了几个直方图在预定义网格结构中描述目标。Kwon和Lee[32]提出了一种自动更新局部块的拓扑结构来处理较大姿态变化的方法。为了更好地处理外观变化，最近一些集成了多种表示方式的方法已经被提出[62, 51, 33]。

**搜索机制** 为了评估目标对象的状态，确定性和随机性方法已经被使用。鉴于跟踪问题是在最优化框架下提出的，假设目标函数对运动参数是可微的，梯度下降方法可以被用于有效地定位目标位置[37, 16, 20, 49]。然而，这些目标函数通常是非线性的并且包含许多局部极小值。为了减轻该问题，密集抽样方法被采用了[22, 5, 26]，而其代价是高计算负荷。另一方面，随机性搜索算法如粒子滤波[28, 44]已经被广泛地使用，因为此类算法对局部极小值相对不太敏感而且计算非常有效[47, 40, 30]。

**模型更新** 为了反映表观变化，更新目标表示或目标模型是至关重要的。Matthews等人[39]提出了Lucas-Kanade算法[37]中的模板更新问题，其模板通过联合从第一帧中抽取的固定参考模板和最近一帧的结果来更新。一些有效的更新算法是基于在线混合模型[29]、在线boosting[22]和增量子空间更新[47]提出来的。对于判别式模型，主要问题是改善样本收集环节，使在线训练得到的分类器更加鲁棒[23, 5, 31, 26]。尽管已经取得了许多进步，寻找一种自适应的表观模型来避免跟踪漂移仍是非常困难的。

**上下文和跟踪器的融合** 上下文信息对于跟踪也是相当重要的。近年来一些跟踪算法利用辅助对象或目标周围的局部视觉信息来协助跟踪[59, 24, 18]。当目标被完全遮挡或者离开图像区域[24]时，上下文信息尤其有用。为了提高跟踪性能，一些跟踪器融合方法最近被提出来了。Santner等人[48]提出联合静态、适度自适应和高度自适应跟踪器的方法来反映表观的变化。在贝叶斯框架下，（有些跟踪算法）甚至维持和选择多种跟踪器[34]或多重特征[61]来更好地反映表观的变化。

1. **被评估的算法和数据集**

为了公平的评估，我们测试的跟踪算法都是公开的源码或二进制代码，因为所有测试都不可避免地涉及到技术细节和特殊的参数设置[[2]](#footnote-2)。表1列出了被评估算法的清单。我们也在VIVID测试平台[14]评估了这些跟踪器，包括均值迁移(MS-V)，模板匹配(TM-V)，比值迁移(RS-V)和峰值差(PD-V)方法。

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Method** | **Representation** | **Search** | **MU** | **Code** | **FPS** |
| CPF [44] | L, IH | PF | N | C | 109 |
| LOT [43] | L, color | PF | Y | M | 0.70 |
| IVT [47] | H, PCA, GM | PF | Y | MC | 33.4 |
| ASLA [30] | L, SR, GM | PF | Y | MC | 8.5 |
| SCM [65] | L, SR, GM+DM | PF | Y | MC | 0.51 |
| L1APG [10] | H, SR, GM | PF | Y | MC | 2.0 |
| MTT [64] | H, SR, GM | PF | Y | M | 1.0 |
| VTD [33] | H, SPCA, GM | MCMC | Y | MC-E | 5.7 |
| VTS [34] | L, SPCA, GM | MCMC | Y | MC-E | 5.7 |
| LSK [36] | L, SR, GM | LOS | Y | M-E | 5.5 |
| ORIA [58] | H, T, GM | LOS | Y | M | 9.0 |
| DFT [49] | L, T | LOS | Y | M | 13.2 |
| KMS [16] | H, IH | LOS | N | C | 3,159 |
| SMS [13] | H, IH | LOS | N | C | 19.2 |
| VR-V [15] | H, color | LOS | Y | MC | 109 |
| Frag [1] | L, IH | DS | N | C | 6.3 |
| OAB [22] | H, Haar, DM | DS | Y | C | 22.4 |
| SemiT [23] | H, Haar, DM | DS | Y | C | 11.2 |
| BSBT [50] | H, Haar, DM | DS | Y | C | 7.0 |
| MIL [5] | H, Haar, DM | DS | Y | C | 38.1 |
| CT [63] | H, Haar, DM | DS | Y | MC | 64.4 |
| TLD [31] | L, BP, DM | DS | Y | MC | 28.1 |
| Struck [26] | H, Haar, DM | DS | Y | C | 20.2 |
| CSK [27] | H, T, DM | DS | Y | M | 362 |
| CXT [18] | H, BP, DM | DS | Y | C | 15.3 |

表1. 被评估的跟踪算法（MU: model update, FPS: frames per second)。对于目标表示方法， L: local（局部），H: holistic（整体），T: template（模板），IH: intensity histogram（灰度直方图），BP: binary pattern（二值模式），PCA: principal component analysis（主成分分析），SPCA: sparse PCA（稀疏PCA），SR: sparse representation（稀疏表示），DM: discriminative model（判别模型），GM: generative model（生成模型）。对于搜索机制，PF: particle filter（粒子滤波），MCMC: Markov Chain Monte Carlo（马尔科夫链蒙特卡洛方法），LOS: local optimum search（局部最优搜索），DS: dense sampling search（密集采样搜索）。对于模型更新，N: No，Y: Yes。在代码栏中，M: Matlab，C:C/C++，MC: Matlab和C/C++混合编程， suffix E: executable binary code（可执行二进制代码）。

近些年，针对各种视觉问题开发了许多标准数据集，比如伯克利分割数据集[38]，FERET人脸识别[45]和光流数据集[9]。也有一些数据集用于监视场景下的跟踪，比如VIVID [14]和CAVIAR [21]数据集。对于一般的视觉跟踪，评估中使用了更多的序列[47，5]。然而，大部分序列没有作准确值标注，而且定量的评估结果可能是通过不同的初始条件生成的。为了便于公平的性能评估，我们收集并标注了大部分常用的跟踪测试序列。图1展示了每个序列的第一帧，其中目标对象用一个边界框进行初始化标记。



图1. 用于评估的跟踪序列。图中展示了目标对象在每个序列的第一帧中的边界框。这些序列根据我们的结果排名进行排序（参考补充材料）：左上角的序列比右下角的序列更难于跟踪。注意我们在jogging序列中标注了两个目标。

**测试序列的特征** 由于诸多因素会影响跟踪的性能，评估跟踪算法是困难的。为了更好地评估和分析跟踪方法的优点和缺点，我们用11种属性标注所有序列来进行分类，这些属性列在表2中。

我们的数据集中的属性分布如图2(a)所示。某些属性，如OPR和IPR，出现得比其他属性更频繁。该表也说明了一个序列通常被标记了多种属性。除了在整个数据集上总结算法的性能，我们也给对应的属性分别创建了几个子集，用于描述特定的具有挑战性的条件下算法的性能。比如，OCC子集包括29个序列，可用于分析跟踪器处理遮挡的能力。OCC子集中的属性分布如图2(b)所示，其余的子集报表可以在补充材料中获取。

|  |  |
| --- | --- |
| **属性** | **描述** |
| IV | 光照变化 - 目标区域内的光照剧烈变化 |
| SV | 尺度变化 - 第一帧中和当前帧中的边界框尺寸之比的范围超过[1/ts, ts]， ts >1 (ts=2) |
| OCC | 遮挡 - 目标被部分或全部遮挡 |
| DEF | 形变 - 非刚体目标发生形变 |
| MB | 运动模糊 - 目标或摄像机的运动导致的目标区域变模糊 |
| FM | 快速运动 - ground truth的运动大于tm个像素点(tm=20) |
| IPR | 平面内旋转 - 目标在图像平面内发生旋转 |
| OPR | 平面外旋转 - 目标在图像平面外发生旋转 |
| OV | 超出视野 - 目标的一部分离开视野 |
| BC | 背景杂斑 - 目标附近的背景具有和目标类似的颜色或纹理 |
| LR | 低分辨率 - ground-truth边界框内的像素点个数少于tr(tr =400) |

表2. 测试序列所标注的各种属性的清单。表中也列出了测试中使用的阈值。

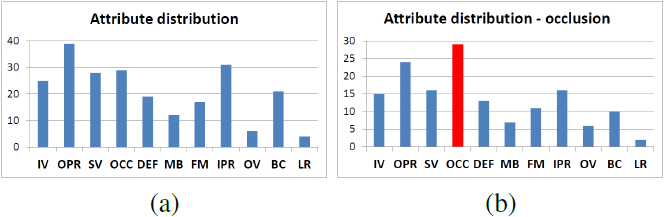


图2. (a)为整个测试数据集的属性分布， (b)为遮挡（OCC）特性在序列中的分布。

1. **评估方法**

在本研究工作中，我们使用精度和成功率做定量分析。另外，我们从两个方面评估了跟踪算法的鲁棒性。

**精确度图** 在跟踪精度评估中，一个被广泛使用的标准是中心位置误差，其被定义为跟踪目标的中心位置和手工标定的准确位置之间的平均欧氏距离。一个序列中所有帧的平均中心位置误差被用于慨括跟踪算法对该序列的总体性能。但是，当跟踪器丢失目标时，输出的跟踪位置是随机的，此时的平均误差值可能无法正确估量跟踪的性能[6]。近年来，精确度图[6, 27]已经被用于测量跟踪的整体性能。精确度图能够显示出评估的位置在给定的准确值的阈值距离之内的帧数占总帧数的百分比。对于每个跟踪器具有代表性的精度评分，我们使用的分数阈值等于20个像素点[6]。

**成功率图** 另一个评估标准是边界框的重叠率。假设跟踪的边界框为γt，准确的边界框是γa，重叠率被定义为 S = |γt∩γa | / |γt∪γa |，其中∩和∪分别表示两个区域的交集和并集，| · |指其区域内的像素点个数。为了估量算法在一系列帧中的性能，我们计算重叠率S大于给定的阈值to的成功帧的数量。成功率图给出了此阈值从0到1变化时成功帧所占的比例。使用某一特定阈值（比如to=0.5）下的一个成功率来评估跟踪器可能并不公平或具有代表性。我们使用每一个成功率图的曲线下面积（AUC）作为替代，用于给跟踪算法进行排序。

**鲁棒性评估** 评估跟踪器的传统方式是，根据第一帧中的准确位置进行初始化，然后在一个测试序列中运行算法，最后得出平均精确度或成功率的结果报告。我们把这种方法称为一次通过的评估（OPE）。然而跟踪器可能对初始化非常敏感，并且在不同的初试帧给予不同的初始化会使其性能变得更差或更好。因此，我们提出两种方式来评估跟踪器对初始化的鲁棒性，即在时间上（即在不同帧开始跟踪）和空间上（即以不同的边界框开始跟踪）扰乱初始化。这两个测试分别称为时间鲁棒性评估（TRE）和空间鲁棒性评估（SRE）。

所提出的测试场景大部分就存在于现实世界中实际应用的场合，而跟踪器通常通过目标检测器来初始化，检测器在位置或尺寸方面可能会给跟踪器引入初始化误差。另外，在不同时刻的实例中，检测器可能被用于重新初始化跟踪器。通过研究跟踪器在不同鲁棒性评估中的特点，我们可以对跟踪算法进行更为深入的理解和分析。

**时间鲁棒性评估** 给定一个标记了目标准确边界框的初试帧，跟踪器被初始化并运行直到序列结束，即整个序列的一部分。跟踪器会在每一个序列的片段上进行评估且整体的统计数据也会被记录下来。

**空间鲁棒性评估** 我们在第一帧中通过移动或缩放准确的ground truth来抽取初始化的边界框。在这里，我们使用8种空间位置上的偏移，包括4种中心偏移和4种角偏移，以及4种尺度变化（见补充材料）。偏移量为目标尺寸的10%，尺度比例变化可取准确值的0.8、0.9、1.1和1.2。因此，针对SRE我们对每个跟踪器评估12次。

1. **评估结果**

对于每一个跟踪器，所有的评估都使用了源码的默认参数。表1列出了每一个跟踪器在Intel i7 3770 CPU (3.4GHz)的PC机上OPE运行时的FPS。更详细的速度统计数据，如最大值和最小值，可以在补充材料中获得。

对于OPE，每一个跟踪器进行了超过29000帧的测试。对于SRE，每个跟踪器在每个序列中被评估了12次，共生成了超过350000个边界框结果。对于TRE，每一个序列被分成20个片段，所以每个跟踪器测试了大约310000帧。据我们所知，这是对视觉跟踪最大规模的一次性能评估。在本文中，我们给出了最重要的发现，而更多的细节和图表可以在补充材料中找到。

**5.1．整体性能**

每一个跟踪器的整体性能以成功率图和精确度图的形式展示在图3中，其中只列出了前10名的算法，至于清晰和完整的图，则已在补充材料中给出。对于成功率图，我们使用AUC分数值来对跟踪器进行总结和排名，而对于精确度图，我们使用了在阈值为20时的结果来进行排名。在精确度图中，某些跟踪器的排名与成功率图中的排名略有不同，因为他们是基于测量跟踪器不同特性的不同标准。由于成功率的AUC评分估量了整体性能，它比单一阈值下成功率图的评分更加准确，下面我们将主要分析基于成功率图的排名，但是使用精确度图作为辅助。

TRE的平均性能比OPE的性能更高，因为OPE所测试的帧数少于TRE的从第一个片段到最后一个片段的帧数之加。由于跟踪器在较短的的序列中倾向于表现更好，TRE中的所有结果的平均值可能更高。另一方面，SRE的平均性能比OPE更低。初始化误差可能会导致跟踪器使用了不精确的表观信息来更新，从而导致跟踪框逐渐漂移。

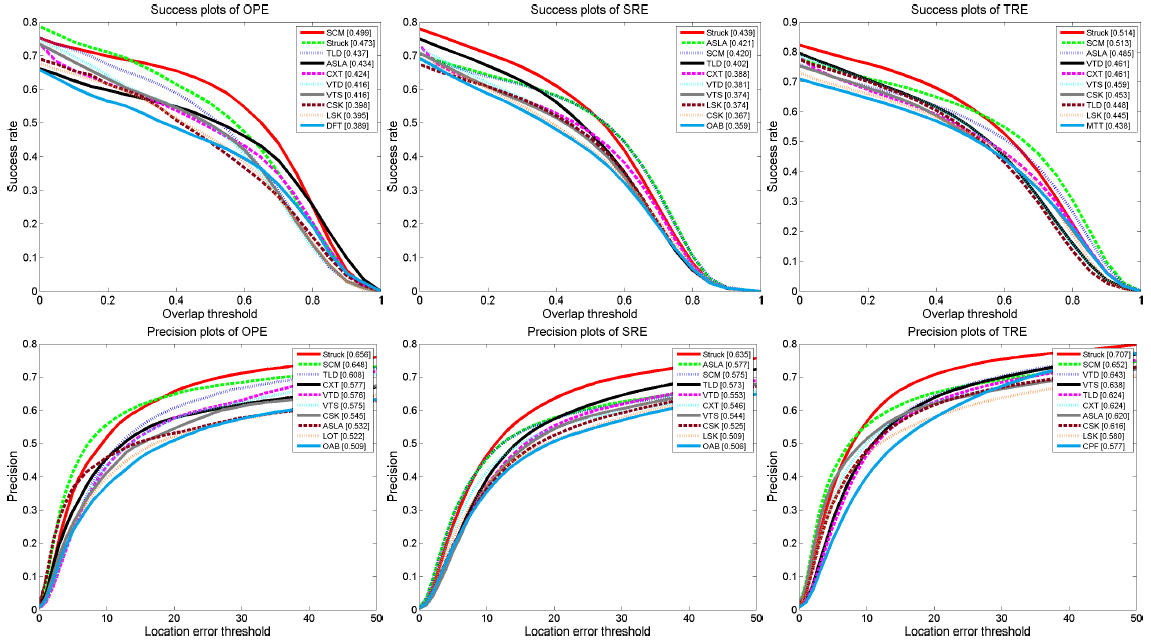


图3. OPE、SRE和TRE图。图中给出了每个跟踪器的性能评分。对于每幅图，排名前10的跟踪器的清晰而完整的图在补充材料中给出（可以查看到高分辨率图）。

在成功率图中，排名最高的跟踪器SCM在OPE中超过Struck2.6%，但是在SRE中低于Struck1.9%。结果也表明OPE不是最佳的性能指标，因为OPE只是SRE或TRE的一次实验而已。TLD在TRE中的排名比在OPE和SRE中的排名更低。这是因为TLD包含重新检测的模块而能在长序列中表现更好，但在TRE中则是大量的短序列片段。Struck在TRE和SRE中的成功率图表明，当重叠率阈值较小时，Struck的成功率比SCM和ALSA更高，但重叠率阈值较大时，Struck的成功率比SCM和ALSA要低。这是因为Struck仅仅估计目标的位置而并未处理尺度变化。

SCM，ASLA，LSK，MTT和L1APG都使用了稀疏表示。这些跟踪器在SRE和TRE中表现良好，这表明稀疏表示是反映表观变化（例如遮挡）的有效模型。我们注意到SCM，ASLA和LSK胜过MTT和L1APG。结果表明局部稀疏表示比全局稀疏模板更加有效。ASLA的AUC分数从OPE到SRE都低于其他前5名跟踪器而且ASLA的排名提升了。这表明ASLA所采用的配准池技术比未配准和背景杂乱更加鲁棒。

在前10名跟踪器中， CSK的速度最高，其所提出的循环行列式结构起到了很重要的作用。VTD和VTS方法采用了混合模型来提高跟踪性能。对比其他排名更高的跟踪器，他们的性能瓶颈归因于所采用的基于稀疏主成分分析表示方法中采用了全局模板。由于篇幅限制，下面章节的分析中只给出SRE图，补充材料中包含了更多的结果。

**5.2．基于特性的性能分析**

通过标记每一个序列的特性，我们用不同的显著特性构建序列的子集，以便于在每一种挑战性因素下分析跟踪器的性能。由于篇幅限制，我们仅仅阐述和分析了在OCC、SV和FM特性下SRE的成功率图和精确度图，如图4所示。补充材料提供了更多的结果。

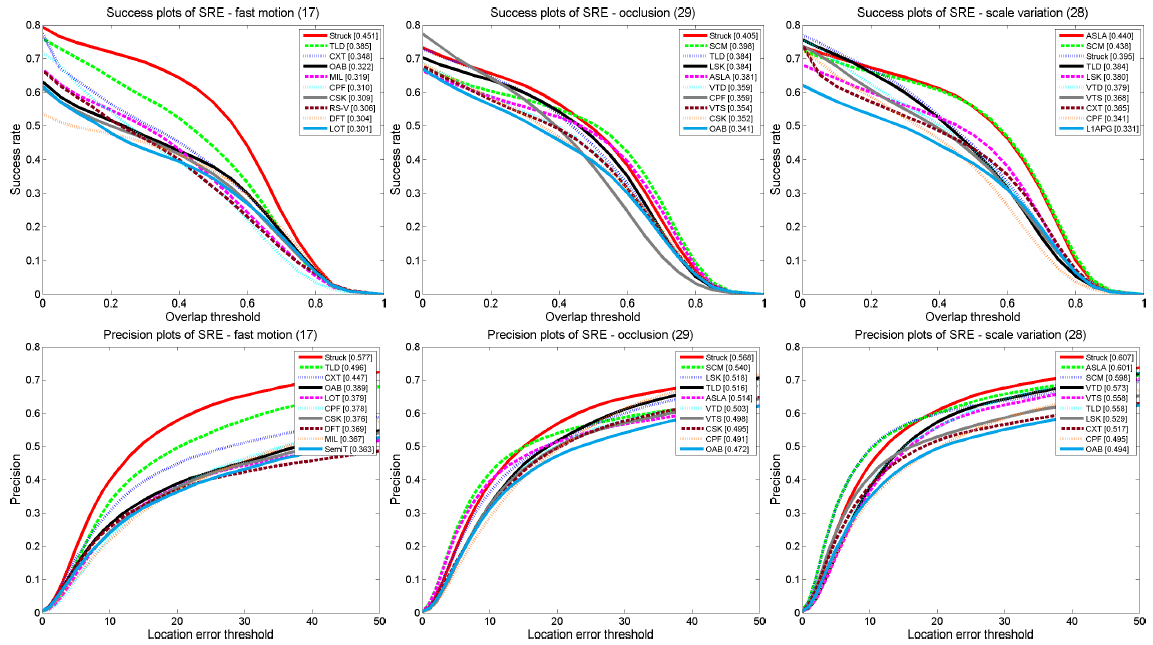


图4. OCC，SV和FM子集的图。每幅图的标题中出现的数值是子集序列的序号。补充材料中仅仅提供了排名前10跟踪器的清晰且完整的图（可以查看到高分辨率图）。

当目标快速运动时，基于密集抽样的跟踪器（如Struck，TLD和CXT）比其它跟踪器表现得更好。原因之一就是搜索范围足够大并且判别模型能够在杂乱的背景中将目标辨别出来。然而，由于不好的动态模型，基于随机搜索的高整体性能跟踪器（如SCM和ASLA）在子集序列中表现并不佳。如果参数设置成较大的值，跟踪器需要采样更多的粒子才能使其性能保持稳定。这些跟踪器可以在动态模型方面设计更为有效的粒子滤波器来进一步提升性能。

在OCC子集中，Struck， SCM，TLD，LSK和ASLA方法胜过其他跟踪器。结果表明结构化学习和局部稀疏表示对解决遮挡问题是有效的。在SV子集中，ASLA，SCM和Struck性能最好。结果表明，具有仿射运动模型的跟踪器（如ASLA和SCM）通常比像Struck那种利用少量期望值只描述平移运动的跟踪器能更好地处理尺度变化。

**5.3. 不同尺度的初始化**

我们已知跟踪器对初始化的变化通常是比较敏感的。图5和图6展示了在不同尺度的初始化条件下，跟踪器的整体性能表现情况。当计算重叠率时，我们重新缩放跟踪结果框，以便于整体性能可以和原始尺度时的性能作比较，即图3中的OPE图。图6说明了所有跟踪器在每一种尺度下的平均性能，当尺度因子较大（比如1.2倍）时跟踪器的性能通常会极大地下降，因为目标的初始表示不可避免地会包含许多背景像素点。随着初始化尺度的增大，TLD，CXT，DFT和LOT的性能会下降。这表明，这些跟踪器对杂乱的背景更加敏感。当尺度因子较小时，有些跟踪器性能会更好，如L1APG，MTT，LOT和CPF。L1APG和MTT出现这种情况的一个原因是目标模板必须被调整到适应通常更小的标准模板的尺寸，以便于如果初始模板较小时，更多的表观细节能够在模型中得以保留。另一方面，当初始边界框增大时，有些跟踪器表现良好甚至优异，比如Struck，OAB，SemiT和BSBT。这表明类哈尔特征由于在计算特征时使用了求和操作而对复杂背景具有一定的鲁棒性。总的来说，相对于其他性能良好的方法，Struck对尺度变化不太敏感。

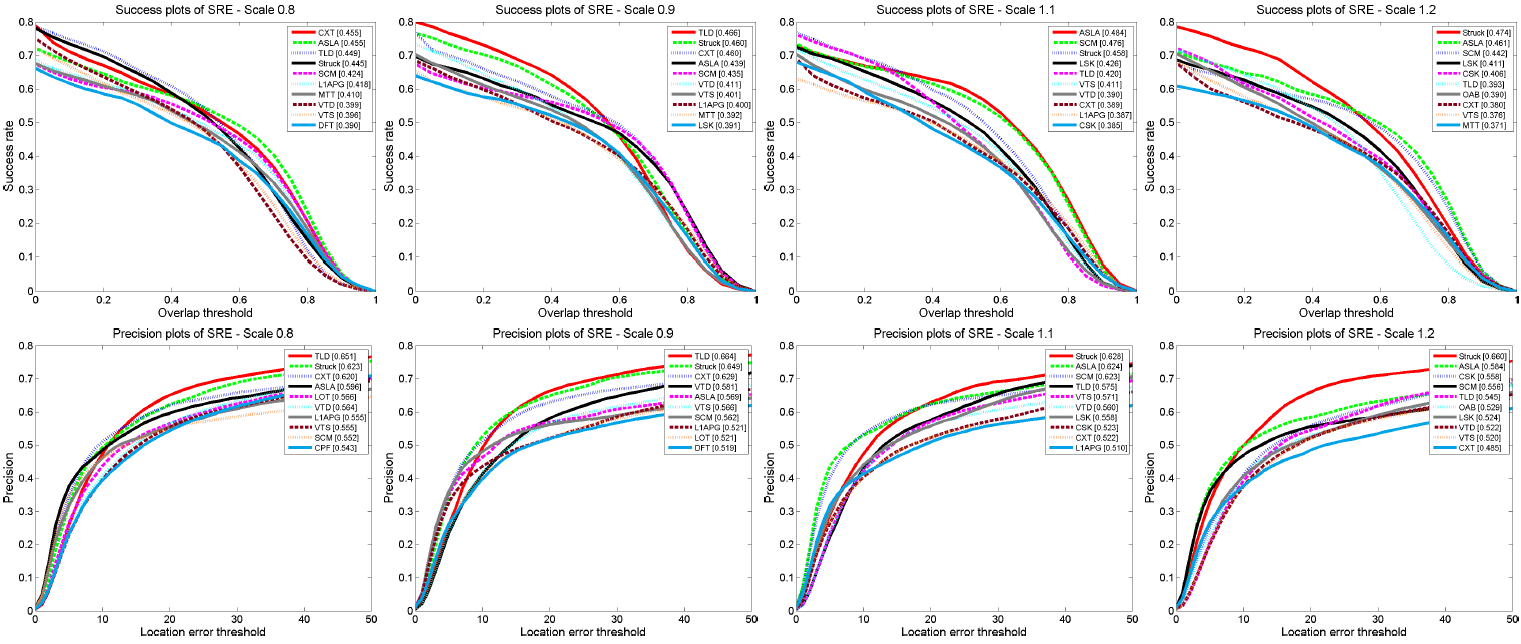


图5. 以不同尺寸边界框初始化的跟踪器的SRE。每幅图上面的数值为尺度因子。对于每幅图，排名前10的跟踪器的清晰且完整的图已在补充材料中给出。

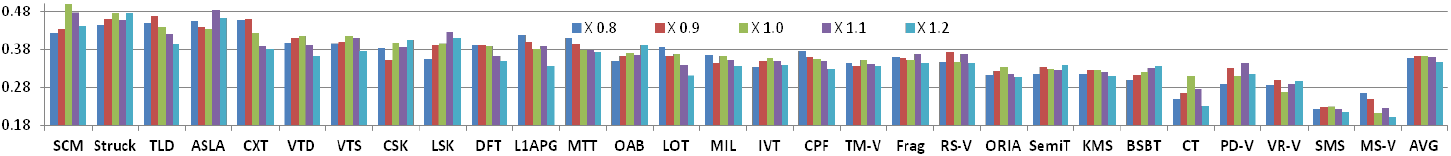


图6. 以不同尺寸边界框初始化的跟踪器的性能总结。AVG（最后一个）阐明了所有跟踪器在每一个尺度下的平均性能。

1. **结论**

在本文中，我们通过大规模的实验对近年来在线跟踪算法的性能进行了评估。基于我们的评估结果和观察，我们强调了对提高跟踪性能必需的一些模块。首先，背景信息对有效的跟踪是至关重要的，我们可以通过在判别模型中隐含地使用先进的学习技术对背景信息进行编码（如Struck），或显式地用作跟踪上下文信息（如CXT）来利用背景信息。其次，局部模型对跟踪是非常重要的，正如局部稀疏表示（如ASLA和SCM）比全局稀疏表示（如MTT和L1APG）更能提升性能。它在目标表观发生部分变化时尤其有用，比如部分遮挡或形变。再者，运动模型或者说动态模型在目标跟踪中也是至关重要的，尤其是当目标的运动幅度较大或突然运动时。然而，我们所评估的大部分跟踪器并未注意到这个部分。基于动态模型的位置预测能够减少搜索范围，因此可以提高跟踪的有效性和鲁棒性。改进这些部分将能进一步提高在线目标跟踪算法现有的技术。

评估结果展示了近十年来目标跟踪领域所取得的巨大进步。我们提出并演示了从多个方面深入分析跟踪算法的评估标准。这些大量的性能评估有助于更好地理解当前先进的在线目标跟踪方法，也给新算法的测评提供了一个平台。我们正在进行的工作着重于拓展数据集和代码库，旨在收录更多完整标注好的序列和跟踪器。

**鸣谢** 略

**参考文献**

[1] A. Adam, E. Rivlin, and I. Shimshoni. Robust Fragments-based Tracking using the Integral Histogram. In *CVPR*, 2006.

[2] N. Alt, S. Hinterstoisser, and N. Navab. Rapid Selection of Reliable Templates for Visual Tracking. In *CVPR*, 2010.

[3] S. Avidan. Support Vector Tracking. *PAMI*, 26(8):1064–1072, 2004.

[4] S. Avidan. Ensemble Tracking. *PAMI*, 29(2):261–271, 2008.

[5] B. Babenko, M.-H. Yang, and S. Belongie. Visual Tracking with Online Multiple Instance Learning. In CVPR, 2009.

[6] B. Babenko, M.-H. Yang, and S. Belongie. Robust Object Tracking with Online Multiple Instance Learning. PAMI, 33(7):1619–1632, 2011.

[7] Y. Bai and M. Tang. Robust Tracking via Weakly Supervised Ranking SVM. In CVPR, 2012.

[8] S. Baker and I. Matthews. Lucas-Kanade 20 Years On: A Unifying Framework. IJCV, 56(3):221–255, 2004.

[9] S. Baker, S. Roth, D. Scharstein, M. J. Black, J. Lewis, and R. Szeliski. A Database and Evaluation Methodology for Optical Flow. In ICCV, 2007.

[10] C. Bao, Y. Wu, H. Ling, and H. Ji. Real Time Robust L1 Tracker Using Accelerated Proximal Gradient Approach. In CVPR, 2012.

[11] M. J. Black. EigenTracking: Robust Matching and Tracking of Articulated Objects Using a View-Based Representation. IJCV, 26(1):63– 84, 1998.

[12] K. Cannons. A Review of Visual Tracking. Technical Report CSE2008-07, York University, Canada, 2008.

[13] R. Collins. Mean-shift Blob Tracking through Scale Space. In CVPR,2003.

[14] R. Collins, X. Zhou, and S. K. Teh. An Open Source Tracking Testbed and Evaluation Web Site. In PETS, 2005.

[15] R. T. Collins, Y. Liu, and M. Leordeanu. Online Selection of Discriminative Tracking Features. PAMI, 27(10):1631–1643, 2005.

[16] D. Comaniciu, V. Ramesh, and P. Meer. Kernel-Based Object Tracking. PAMI, 25(5):564–577, 2003.

[17] N. Dalal and B. Triggs. Histograms of Oriented Gradients for Human Detection. In CVPR, 2005.

[18] T. B. Dinh, N. Vo, and G. Medioni. Context Tracker: Exploring Supporters and Distracters in Unconstrained Environments. In CVPR, 2011.

[19] J. Fan, X. Shen, and Y. Wu. Scribble Tracker: A Matting-based Approach for Robust Tracking. PAMI, 34(8):1633–1644, 2012.

[20] J. Fan, Y. Wu, and S. Dai. Discriminative Spatial Attention for Robust Tracking. In ECCV, 2010.

[21] R. B. Fisher. The PETS04 Surveillance Ground-Truth Data Sets. In PETS, 2004.

[22] H. Grabner, M. Grabner, and H. Bischof. Real-Time Tracking via On-line Boosting. In BMVC, 2006.

[23] H. Grabner, C. Leistner, and H. Bischof. Semi-supervised On-Line Boosting for Robust Tracking. In ECCV, 2008.

[24] H. Grabner, J. Matas, L. V. Gool, and P. Cattin. Tracking the Invisible: Learning Where the Object Might be. In CVPR, 2010.

[25] G. D. Hager and P. N. Belhumeur. Efficient Region Tracking With Parametric Models of Geometry and Illumination. PAMI, 20(10):1025–1039, 1998.

[26] S. Hare, A. Saffari, and P. H. S. Torr. Struck: Structured Output Tracking with Kernels. In ICCV, 2011.

[27] F. Henriques, R. Caseiro, P. Martins, and J. Batista. Exploiting the Circulant Structure of Tracking-by-Detection with Kernels. In EC- CV, 2012.

[28] M. Isard and A. Blake. CONDENSATION–Conditional Density Propagation for Visual Tracking. IJCV, 29(1):5–28, 1998.

[29] A. D. Jepson, D. J. Fleet, and T. F. El-Maraghi. Robust Online Appearance Models for Visual Tracking. PAMI, 25(10):1296–1311, 2003.

[30] X. Jia, H. Lu, and M.-H. Yang. Visual Tracking via Adaptive Structural Local Sparse Appearance Model. In CVPR, 2012.

[31] Z. Kalal, J. Matas, and K. Mikolajczyk. P-N Learning: Bootstrapping Binary Classifiers by Structural Constraints. In CVPR, 2010.

[32] J. Kwon and K. M. Lee. Tracking of a Non-Rigid Object via Patchbased Dynamic Appearance Modeling and Adaptive Basin Hopping Monte Carlo Sampling. In CVPR, 2009.

[33] J. Kwon and K. M. Lee. Visual Tracking Decomposition. In CVPR, 2010.

[34] J. Kwon and K. M. Lee. Tracking by Sampling Trackers. In ICCV, 2011.

[35] X. Li, W. Hu, C. Shen, Z. Zhang, A. Dick, and A. Hengel. A Survey of Appearance Models in Visual Object Tracking. TIST, 2013, in press.

[36] B. Liu, J. Huang, L. Yang, and C. Kulikowsk. Robust Tracking using Local Sparse Appearance Model and K-Selection. In CVPR, 2011.

[37] B. D. Lucas and T. Kanade. An Iterative Image Registration Technique with An Application to Stereo Vision. In IJCAI, 1981.

[38] D. R. Martin, C. C. Fowlkes, and J. Malik. Learning to Detect Natural Image Boundaries Using Local Brightness, Color, and Texture Cues. PAMI, 26(5):530–49, 2004.

[39] I. Matthews, T. Ishikawa, and S. Baker. The Template Update Problem. PAMI, 26(6):810–815, 2004.

[40] X. Mei and H. Ling. Robust Visual Tracking using L1 Minimization. In ICCV, 2009.

[41] X. Mei, H. Ling, Y. Wu, E. Blasch, and L. Bai. Minimum Error Bounded Efficient L1 Tracker with Occlusion Detection. In CVPR, 2011.

[42] X. Mei, H. Ling, Y. Wu, E. Blasch, and L. Bai. Efficient Minimum Error Bounded Particle Resampling L1 Tracker with Occlusion De- tection. TIP, 2013, in press.

[43] S. Oron, A. Bar-Hillel, D. Levi, and S. Avidan. Locally Orderless Tracking. In CVPR, 2012.

[44] P. Pe´rez, C. Hue, J. Vermaak, and M. Gangnet. Color-Based Probabilistic Tracking. In ECCV, 2002.

[45] P. Phillips, H. Moon, S. Rizvi, and P. Rauss. The FERET Evaluation Methodology for Face-Recognition Algorithms. PAMI, 22(10):1090–1104, 2000.

[46] F. Porikli, O. Tuzel, and P. Meer. Covariance Tracking using Model Update based on Lie Algebra. In CVPR, 2006.

[47] D. Ross, J. Lim, R.-S. Lin, and M.-H. Yang. Incremental Learning for Robust Visual Tracking. IJCV, 77(1):125–141, 2008.

[48] J. Santner, C. Leistner, A. Saffari, T. Pock, and H. Bischof. PROST: Parallel Robust Online Simple Tracking. In CVPR, 2010.

[49] L. Sevilla-Lara and E. Learned-Miller. Distribution Fields for Tracking. In CVPR, 2012.

[50] S. Stalder, H. Grabner, and L. van Gool. Beyond Semi-Supervised Tracking: Tracking Should Be as Simple as Detection, but not Sim- pler than Recognition. In ICCV Workshop, 2009.

[51] B. Stenger, T. Woodley, and R. Cipolla. Learning to Track with Multiple Observers. In CVPR, 2009.

[52] F. Tang, S. Brennan, Q. Zhao, and H. Tao. Co-Tracking Using SemiSupervised Support Vector Machines. CVPR, 2007.

[53] O. Tuzel, F. Porikli, and P. Meer. Region Covariance: A Fast Descriptor for Detection and Classification. In ECCV, 2006.

[54] P. Viola and M. J. Jones. Robust Real-Time Face Detection. IJCV, 57(2):137–154, 2004.

[55] D. Wang, H. Lu, and M.-H. Yang. Online Object Tracking with Sparse Prototypes. TIP, 22(1):314–325, 2013.

[56] Y. Wu, J. Cheng, J. Wang, H. Lu, J. Wang, H. Ling, E. Blasch, and L. Bai. Real-time Probabilistic Covariance Tracking with Efficient Model Update. TIP, 21(5):2824–2837, 2012.

[57] Y. Wu, H. Ling, J. Yu, F. Li, X. Mei, and E. Cheng. Blurred Target Tracking by Blur-driven Tracker. In ICCV, 2011.

[58] Y. Wu, B. Shen, and H. Ling. Online Robust Image Alignment via Iterative Convex Optimization. In CVPR, 2012.

[59] M. Yang, Y. Wu, and G. Hua. Context-Aware Visual Tracking. PAMI, 31(7):1195–1209, 2008.

[60] A. Yilmaz, O. Javed, and M. Shah. Object Tracking: A Survey. ACM Computing Surveys, 38(4):1–45, 2006.

[61] J. H. Yoon, D. Y. Kim, and K.-J. Yoon. Visual Tracking via Adaptive Tracker Selection with Multiple Features. In ECCV, 2012.

[62] L. Yuan, A. Haizhou, T. Yamashita, L. Shihong, and M. Kawade. Tracking in Low Frame Rate Video: A Cascade Particle Filter with Discriminative Observers of Different Life Spans. PAMI, 30(10):1728–1740, 2008.

[63] K. Zhang, L. Zhang, and M.-H. Yang. Real-time Compressive Tracking. In ECCV, 2012.

[64] T. Zhang, B. Ghanem, S. Liu, and N. Ahuja. Robust Visual Tracking via Multi-task Sparse Learning. In CVPR, 2012.

[65] W. Zhong, H. Lu, and M.-H. Yang. Robust Object Tracking via Sparsity-based Collaborative Model. In CVPR, 2012.

***Translated by Roamer at NUPT***

***May 17th, 2015***

1. 此处的“在线”是指在跟踪过程中，在任何时刻的实例中仅有之前少数几帧的信息被利用。 [↑](#footnote-ref-1)
2. 一些代码[36，58]是直接联系作者得到的，也有一些方法是我们自己完成的[44，16]。 [↑](#footnote-ref-2)