

指导教师： 杨涛

提交时间： 2015/3/29

The task of
Digital Image Processing

数字图像处理

School of Computer Science

No: 1

姓名： 丛垚

学号： 2012202529

班号： 10011206



实时视觉跟踪的自适应颜色属性

马丁 Danelljan¹, 法赫德 沙赫巴

兹 卡恩¹,

迈克尔 菲尔斯贝格¹, 约斯特 van de

维吉尔²

¹ 计算机视觉实验室, 林雪平大学, 瑞士

² 计算机视觉中心, 计算机科学部门, 巴塞

罗那自治大学, 西班牙

{martin.danelljan, fahad.khan, michael.felsberg}@liu.se, joost@cvc.uab.es

摘要

在计算机视觉中, 视觉跟踪是一项具有挑战性的问题。大多数顶尖的视觉跟踪器是根据亮度信息或者用简单的颜色标志来描述图像。与目前的视觉跟踪相反的是, 在目标识别和侦察中, 复杂而且有特色的颜色结合亮度已经显现出出色的性能。由于跟踪问题的复杂性, 所需的颜色特性应该是计算效率高, 而且具备一定程度上的光照不变性, 同时要保持较高的辨识度。

这篇论文研究的是在跟踪监测框架中颜色的贡献。我们的研究表明: 在视觉跟踪中颜色属性表现地很优秀。而且我们进一步提出一种自适应低维颜色属性的变体。量化和属评估是执行 41 项具有挑战性的基准颜色序列。本文提出的方法使基线基于强度的跟踪器在中距离测量精度提高了 24%。而且, 我们发现我们的方法在运行上比最先进的跟踪方法快超过 100 帧每秒。

1. 简介

视觉对象跟踪是在计算机视觉中最具挑战性的问题之一, 其目的是在一个图像序列中估计一个目标的所在位置。它在众多应用程序扮演了一个重要的角色, 尤其是在人机交互, 监视和机器人学领域。像光照变化, 局

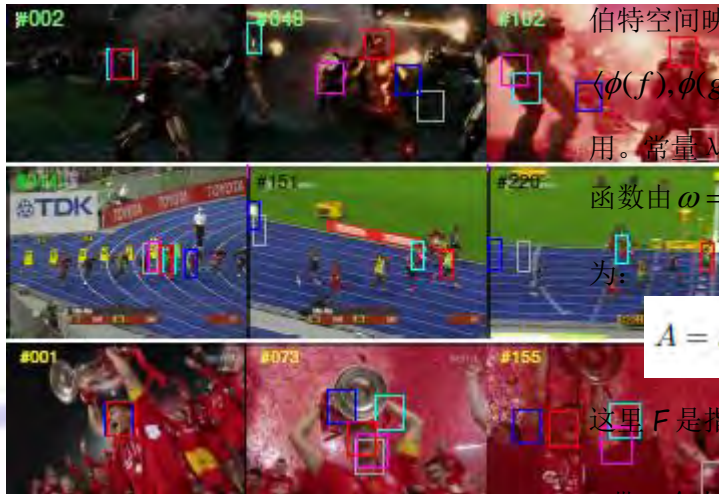
部遮挡, 背景杂斑和外形变形等许多因素使问题更加复杂。在这篇论文中, 我们调查了颜色可以把困难减轻到什么程度。

许多顶级的跟踪方法依赖于亮度或纹理信息 [7, 27, 11, 5, 20]。视觉跟踪进步显著, 颜色信息的应用却被局限于简单的色彩空间的转换。与视觉跟踪相比, 复杂的颜色特征在物体识别和监测方面已经表现出卓越的效果。对于视觉跟踪来说, 开发色彩信息是一项艰难的挑战。由于光线的变化, 阴影, 遮挡, 高光, 相机和物体的相对位置等原因, 量色方法会导致一个图像序列产生很大的波动。在彩色图像领域, 关于这些因素的鲁棒性的研究已经开始进行, 并且成功应用于图像分类和行为识别等方面。因此, 为了视觉对象跟踪这个任务, 我们评估了现有的颜色转换。

现在有两种主要方法来处理视觉跟踪问题, 就是衍生法和判别分类法。衍生法通过寻找与目标模型做相近的区域来处理问题。在这些方法中, 模型是根据模板或者子空间模型建立起来的。而判别分类法 [7, 27, 5, 9] 是将目标和背景区分开来, 把跟踪转换为一个二分类问题。不像衍生法, 判别分类法同时运用目标和背景信息来寻找判定边界, 从而在背景中区分出目标物体。这是一种被用于跟踪检测的方法 [7, 9], 我们用目标和周围背景的样本块来在线地训练一个判别式分类器。最近, 由 Wu 等人提出了一个在线跟踪算法的综合评价法。在这个评估方法中, 一个被叫做码移键控法的跟踪检测方法在顶级的十种视觉跟踪方法中表现出最快的速度。这个方法不是在一个帧中展示获取子窗口的过程导致循环结构, 而是探测一个稠密取样策略。由于它的竞争力绩效和最快实现速度, 我们的方法是以码移键控跟踪方法为基础的。

贡献: 在这篇论文中, 我们用色彩属性扩展了码移键控跟踪方法。这个方法在目标识别上已经表现出极为出色的结果。这要归功于它们在光照不变性和辨别力上的良好平衡。对于多通道(颜色)信号, 我们发现码移键控跟踪方法的更新计划并不是最优的。为了解决这一问题, 我们适应了更新计划, 并且用

实验验证它对于多通道跟踪的重要性。颜色属性的高维度结果导致增加计算开销。这可能限制其在实时监测和机器人等领域的应用。为了解决这一问题，我们计划用一种适合的降维技术。这个技术可以把原本的 11 维降至二维。我们发现这样可以使跟踪运行速度超过每秒 100 帧都没有损失精度。对比于在目标识别中流行的其他颜色表现，一个广泛的评估显示颜色属性性能优越。最后，我们表明利用综合评价法，在 41 个图像序列中我们的跟踪方法中表现出先进性能。图形 1 显示出在具有挑战性的环境中的跟踪结果。对比于其他几种先进算法，我们的方法表现优良。



图形 1: 在具有挑战性的环境中，例如：亮度变化，闭塞，变形和平面旋转等，顶级的跟踪方法与我们的方法的对比。这些图例分别来自于《钢铁侠》，博尔特和足球赛。

Struck[7], EDFT[6], CSK[9], LSHT[8]和我们的方法的结果分别由蓝色，灰色，青色，品红和红色方框框出。

2. 码移键控跟踪方法

我们的方法是基于码移键控方法。近期的评估表明，这种方法在在顶尖的 10 种跟踪方法中速度最快。这种方法利用核的最小二乘分类器通过一个单一图像点阵来了解一个目标。这种方法之所以有这样快的速度，是因为它扩展了循环行列式结构。这个循环行列式从局部图像点阵的周期性假设中得出。这里我们为这个方法提供了一个简单的概述。

我们用一个单一的灰度图像点阵 x 训练这个分类器，这个点阵取自目标周围 $M \times N$ 大小。这个分类方法把所有的环形移动 $x_{m,n}$ ($m, n \in \{0, \dots, M-1\} \times \{0, \dots, N-1\}$) 当做分类器的训练例子。这些点是通过高斯函数变换成 y 的，所以 $y(m, n)$ 是 $x_{m,n}$ 的高斯函数。分类器被训练通过改变 ω 使代价减到最小 (1)。

$$\epsilon = \sum_{m,n} |(\phi(x_{m,n}), \omega) - y(m, n)|^2 + \lambda \langle \omega, \omega \rangle \quad (1)$$

在这个函数中， ϕ 是核心 κ 诱导的希尔伯特空间映像，其 y 中点积

$\langle \phi(f), \phi(g) \rangle = \kappa(f, g)$ 起到了决定性作用。常量 $\lambda \geq 0$ 是正则化参数。(1) 中的代价函数由 $\omega = \sum_{m,n} a(m, n) \phi(x_{m,n})$ ，其中系数 a 为：

$$A = \mathcal{F}\{a\} = \frac{Y}{U_x + \lambda} \quad (2)$$

这里 F 是指 DFT (离散傅里叶变换)。我们用 s 带一个大写字母表示 DFT，例如 $Y = F\{y\}$,

$U_x = F\{u_x\}$ 。在这里，

$u_x(m, n) = \kappa(x_{m,n}, x)$ 是核函数 κ 的输出。公式 2 中保持如果 κ 是移位不变量，也就是说 $\kappa(f_{m,n}, g_{m,n}) = \kappa(f, g)$ 对于任何 m, n, f, g 都成立。这适用于高斯径向基函数内核利用码移键控跟踪方法的服务。

探测步骤是由在新的画面中第一次出现的一个灰度图像点阵 z 执行，该点阵大小为 $M \times N$ 。探测结果由 $\hat{y} = \mathcal{F}^{-1}\{AU_z\}$,

在这个公式中， $U_z = \mathcal{F}\{u_z\}$ 是傅里叶变换内核输出 $u_z(m, n) = \kappa(z_{m,n}, \hat{x})$ 对于

点阵 z 的实际对应。在这里 \hat{x} 表示目标的灰度块的外表，它向多帧方向近似。在新的画面中，目标位置通过寻找 \hat{y} 的最大值点来估计。[9]的工作显示，核心输出 u_x 和 u_z 可以用快速傅里叶变换高效的计算。更多细节，请参考文献[9]。

3. 颜色视觉跟踪

为了体现颜色信息，我们将码移键控跟踪算法通过定义适当的内核 κ 扩展到多维颜色特性。这通过扩展径向基函数内核的多维特性的 L^2 范数实现的。这个特征从图像点阵中提取出来，这个图像点阵通过函数 $x : \{0, \dots, M-1\} \times \{0, \dots, N-1\} \rightarrow \mathbb{R}^D$ 再现出来。在这个函数中 $x(m,n)$ 是一个包含在 (m,n) 区域内所有特征值的 D 维向量。在传统的码移键控跟踪算法中一个灰度图像点阵预先要与一个汉宁窗相乘。我们对每个特性通道应用相同的过程。最后的表现是通过亮度叠加和颜色通道获得的。

3.1. 视觉跟踪与颜色属性

总的来说，颜色特征的选择是一个视觉跟踪算法成功与否的关键。最近，颜色属性 [23] 对于目标识别，目标检测和行为识别取得了优秀的结果 [14, 13, 12]。在此，我们针对视觉跟踪问题研究它们。颜色属性，或者说色名，是人类为了分别世界上的色彩而为颜色贴上的文字标签。在柏林和凯 [3] 的一项语言学研究中表明，英语包含了 11 种基本颜色词：黑色，蓝色，棕色，灰色，绿色，橙色，粉色，紫色，红色，白色和黄色。在计算机视觉领域，颜色命名是一项把 RGB 观测和语言颜色标签联系起来的操作。我们使用 [23] 提供的映射，这个映射是自动从图片检索与谷歌图片搜索中获得的。这将 RGB 值映射到一个概率性的 11 维颜色表示减小到 1。

传统的码移键控跟踪算法把灰度值规范到 $[-0.5, 0.5]$ 。由于窗口操作造成计数器的失真，影响了内核的 L^2 距离。我们研究了两种不同的色名标准化技术。在第一种情况中，色名以简单地从色坐标中减去 $1/11$ 为

中心。这种方法把色名投射到一个十维的子空间，最后色坐标总结到 0。在第二种情况下，正常化是由把色名投影到一个十维的子空间的标准正交基执行的。这种投射使色名集中，同时把维度从 11 减小到 10。这个标准正交基的选择对于码移键控跟踪算法并不重要，正如在 3.3 中讨论的那样。我们发现第二种技术能获得更好的性能，因此用它来规范化色名。

3.2. 对于颜色特征的分类器的鲁棒控制

为了完成视觉跟踪，即对于外表变化是鲁棒的，更新目标模型是必要的。在码移键控跟踪算法中，模型包含了学习目标外观 \hat{x} 和转换分类器系数 A 。这些计算都仅仅考虑的当前外观状况。因而，跟踪器通过简单的线性插值使用了一个特定的更新分类器系

的数方法： $A^p = (1 - \gamma)A^{p-1} + \gamma A$ ，在这个公式中 p 当前画面的指标， γ 是一个学习率参数。由于不是所有的前一帧被同时用作更新当前模型，所以这是次佳的操作。与码移键控跟踪算法相反，MOSSE 跟踪算法 [4] 采用一种鲁棒的更新计划，它在计算当前模型时考虑到以前所有帧的模型。然而，该方案只适用于线性内核和一维特性。这里，我们用内核分类器和多位的颜色特性推广了 [4] 的更新计划。

为了更新分类器，我们考虑所有从第一帧到当前帧 p 从目标上提取的表像 $\{x^j : j = 1, \dots, p\}$ 。代价函数由这些帧的加权平均二次误差构成。为了保持训练和监测任务的简单，解决方法局限于只包含一个分类器系数 a 。每一帧 j 用一个常量 $\beta_j \geq 0$ 加权。因而总代价的表达式是：

$$\epsilon = \sum_{j=1}^p \beta_j \left(\sum_{m,n} |(\phi(x_{m,n}^j), w^j) - y^j(m,n)|^2 + \lambda(w^j, w^j) \right), \text{ where } w^j = \sum_{k,l} a(k,l) \phi(x_{k,l}^j)$$

这个价值函数被缩减到如下：

$$A^p = \frac{\sum_{j=1}^p \beta_j Y^j U_x^j}{\sum_{j=1}^p \beta_j U_x^j (U_x^j + \lambda)}$$

像在(2)，我们规定傅里叶变换核心输出

$$U_x^j = \mathcal{F}\{u_x^j\}, \text{ 在这个公式中}$$

$u_x^j(m, n) = \kappa(x_{m,n}^j, x^j)$ 权重 β_j 被搁置，转而用学习率参数 γ 。整个模型使用(5)被更新。在(4)中的分式 $A^p = A_N^p / A_D^p$ 的分子

A_N^p 和分母 A_D^p 是分开更新的。对象外观 \hat{x}^p 同传统的码移键控跟踪算法一样更新。

$$\begin{aligned} A_N^p &= (1 - \gamma)A_N^{p-1} + \gamma Y^p U_x^p \\ A_D^p &= (1 - \gamma)A_D^{p-1} + \gamma U_x^p (U_x^p + \lambda) \\ \hat{x}^p &= (1 - \gamma)\hat{x}^{p-1} + \gamma x^p \end{aligned}$$

注意，这个方案语序模型在没有存储所有的从前的表象的情况下更新。但是当前模型 $\{A_N^p, A_D^p, \hat{x}^p\}$ 必须被保存下来。然后利用(5)，这个模型在每一新帧中更新。这也确保了计算量的增长对于跟踪算法的速度的影响可以忽略不计。正如在传统码移键控跟踪算法，已知外观 \hat{x}^p 被用来估计下一帧 $p+1$ 的监测结果 \hat{y} 。

3.3. 低维自适应颜色属性

码移键控跟踪算法的计算时间与特征降维成线性关系。这对于想颜色属性这样的高维的颜色特征是一个问题。我们建议用一个自适应的降维技术，在保存有用信息的同时彻底地减少色维度的数量，从而显著地提升速度。

我们用公式表示为当前帧 p 寻找一个合适的维度减少映射方法这个问题，并通过使代价函数最小的方式解决。

$$\eta_{\text{tot}}^p = \alpha_p \eta_{\text{data}}^p + \sum_{j=1}^{p-1} \alpha_j \eta_{\text{smooth}}^j$$

(4) η_{data}^p 是一个只取决于当前帧的数据项，

η_{smooth}^j 是一个关于帧数 j 的平滑度系数。这个系数影响由权重 $\alpha_1, \dots, \alpha_p$ 控制。

令 \hat{x}^p 成为 D_1 - 维的已知外观。降维技术会通过正交列向量找到一个 $D_1 \times D_2$ 映射矩阵 B_p 。通过线性映射

$$\bar{x}^p(m, n) = B_p^T \hat{x}^p(m, n), \forall m, n, B_p \text{ 可以}$$

计算外表的新的 D_2 - 维的特征图谱 \bar{x}^p 。该数据项包含当前外观的重构误差。

$$(5c) \quad \eta_{\text{data}}^p = \frac{1}{MN} \sum_{m,n} \|\hat{x}^p(m, n) - B_p B_p^T \hat{x}^p(m, n)\|^2 \quad (7)$$

数据项(7)的最小值符合关于当前外观 \hat{x}^p 的主成分分析的表现。然而，由于从前的已知分类器系数 A^p 已经过时了，只用(7)更新投影矩阵会恶化目标模型的质量。

为了获得一个鲁棒的投影矩阵学习算法，我们在(6)中加入了平滑度系数。让 B_j 作为投影矩阵已经在更早的帧 ($j < p$) 中被计算。如果新的投影矩阵 B_p 和更早的投影矩阵 B_j 中的列向量没有保持同样的特征子空间，平滑度系数只是添加了一个代价。这是由于点积和径向基函数核心程序在一元运算中是不变的这一事实产生的。因此，假如它持续同样的特殊选择的基础是不重要的。平滑度系数是：

$$(6) \quad \varepsilon_{\text{smooth}}^j = \sum_{k=1}^{D_2} \lambda_j^{(k)} \|b_j^{(k)} - B_p B_p^T b_j^{(k)}\|^2 \quad (8)$$

算法 1 自适应投影矩阵计算

输入:

帧数 p ; 已知对象的外观 \hat{x}^p

从前协方差矩阵 Q_{p-1} ; 参数 μ , D_2

输出:

投影矩阵 B_p , 当前协方差矩阵 Q_p

- 1: Set $\bar{x}^p = \frac{1}{MN} \sum_{m,n} \hat{x}^p(m, n)$
- 2: Set $C_p = \frac{1}{MN} \sum_{m,n} (\hat{x}^p(m, n) - \bar{x}^p)(\hat{x}^p(m, n) - \bar{x}^p)^T$
- 3: **if** $p = 1$ **then**
- 4: Set $R_1 = C_1$
- 5: **else**
- 6: Set $R_p = (1 - \mu)Q_{p-1} + \mu C_p$
- 7: **end if**
- 8: Do EVD $R_p = E_p S_p E_p^T$, with sorted eigenvalues in S_p
- 9: Set B_p to the first D_2 columns in E_p
- 10: Set $[\Lambda_p]_{i,j} = [S_p]_{i,j}, 1 \leq i, j \leq D_2$
- 11: **if** $p = 1$ **then**
- 12: Set $Q_1 = B_1 \Lambda_1 B_1^T$
- 13: **else**
- 14: Set $Q_p = (1 - \mu)Q_{p-1} + \mu B_p \Lambda_p B_p^T$
- 15: **end if**

公式 8 是更早的基础向量 B_j 在新的基础 B_p 中的重构误差。每个基础向量 $b_j^{(k)}$ 在 B_j 中的重要性由权重 $\lambda_j^{(k)} \geq 0$ 决定。

用数据项(7)和平滑度系数(8), 在 $B_p^T B_p = I$ 的约束下, 总代价(6)减少了。这是通过对矩阵 $R_p = \alpha_p C_p + \sum_{j=1}^{p-1} \alpha_j B_j \Lambda_j B_j^T$ 进行特征

值分解而减少的。矩阵中, C_p 是当前外观的协方差矩阵, Λ_j 是权重为 $\lambda_j^{(k)}$ 的 $D_2 \times D_2$ 对角矩阵。投影矩阵 B_p 被选为 D_2 下的 R_p 的标准化特征向量, 其中 R_p 相当

于特征值。我们把(8)中的权值 $\lambda_j^{(k)}$ 放入 R_j 的特征值中, 这相当于基础向量 $b_j^{(k)}$ 。在(6)中的权重 α_j 被用作学习率参数 μ 。这确保了矩阵 R_p 的有效计算, 却不需要存储所有从前的矩阵 B_j 和 Λ_j 。过程在算法 1 中总结出来。

4. 实验

这里我们给出了我们的实验结果。首先, 计划为视觉跟踪提出了颜色特征的综合评价法(适用于目标识别)。然后, 我们评估了所推荐的颜色特征学习计划。再次, 我们评估了我们的自适应低维颜色属性。最后, 我们提供了在定量和属与比目前顶尖的跟踪算法更出色的算法。

4.1. 实验装置

我们的方法在本机 Matlab 中实现¹。实验执行环境是因特尔至强 2 内核 2.66GHz 的 CPU 和 16GB 的 RAM。在我们的方法中，我们用与在[9]中为传统码移键控跟踪算法中建议的相同的参数值。我们的自适应颜色属性的学习率参数 μ 对于所有的图像序列都被固定在 0.15。

数据集：我们令所有 35 种颜色序列² 被用在跟踪算法[25]的最近评价中。此外，我们又采用了 6 中其他的颜色序列³，即：风筝冲浪，衬衫，冲浪运动员，宝石和熊猫。用于我们实验的图像序列具有挑战性的情况，比如：动态模糊，光照变化，尺度变换，遮挡，平面旋转，变形，消失，背景杂斑和低分辨率。

评价方法：为了证实我们提出的方法的性能，我们效仿在[25]中采用的方案。结果给出了三个评价指标：中心位置误差 (CLE)，距离精度 (DP) 和重复精度 (OP)。中心位置错误被计算为目标的预估中心位置 and 实际位置平均欧式距离。距离精度是序列中帧的相对数量，在这个序列中中心未知错误小于某一临界值。我们将 DP 临界值定在 20 个像素 [9, 25]。结果用 41 种图像序列的 CLE 和 DP 值的中值来概括。我们也展示了在中间帧频的跟踪算法速度。中值结果表现出对整体性能的鲁棒估计。

我们也提出可精度和成功图[25]。在精密策划中，距离精度被控制在一个阈值范围内。跟踪算法用 20 像素的距离精度结果来排名。一个成功图例要容纳重叠精度的一系列阈值。重叠精度被规定为帧的百分比，这些帧的边界框重叠超过一个阈值 $t \in [0, 1]$ 。跟踪算法根据区仙侠面具进行排名。精度和成功点都显示了在所有序列的平均精度成绩。

4.2. 颜色特征

除了根据颜色属性评估跟踪算法外，我们还用一种关于其他颜色表示的广泛评价。

这些颜色特征的动机不同于光照不变性和生物启发颜色表示的辨别力。

RGB: 作为一个基线算法，我们用标准的三原色色彩空间。

LAB: LAB 彩色空间视觉均匀，意味着在相等距离下，在感知上颜色也可以被认为是同样距离。

YCbCr: YCbCr 近似于视觉感知，通常用于图像压缩算法。

rg: rg 色彩通道是第一个我们认为的光度不变性颜色表现。他们用

$$(r, g) = \left(\frac{R}{R+G+B}, \frac{G}{R+G+B} \right)$$

来计算，并且对于阴影和遮蔽效应不变。

HSV: 在 HSV 色彩空间中，H 和 S 对于阴影和遮蔽是不变的，而且 H 对镜面反射也是不变的。

Opponent: 图像根据如下公式转换：

$$\begin{pmatrix} O1 \\ O2 \\ O3 \end{pmatrix} = \begin{pmatrix} \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} & 0 \\ \frac{1}{\sqrt{6}} & \frac{1}{\sqrt{6}} & \frac{-2}{\sqrt{6}} \\ \frac{1}{\sqrt{3}} & \frac{1}{\sqrt{3}} & \frac{1}{\sqrt{3}} \end{pmatrix} \begin{pmatrix} R \\ G \\ B \end{pmatrix}. \quad (9)$$

这个表示法对于镜面反射不变的。

C: C 颜色表示通过规范强度用阴影遮蔽增加光度不变量。这是根据以下公式完成的。

$$C = \begin{pmatrix} O1 & O2 & O3 \end{pmatrix}^T [21]$$

HUE: hue 是一个 36 维直方图表示法，其中

$$H = \arctan \left(\frac{O1}{O2} \right).$$

这个直方图的更新是通过饱和度 $S = \sqrt{O1^2 + O2^2}$ 来应对 hue 表示法的不稳定性。这个表示法对于阴影遮挡和镜面反射是不变的。

Opp-Angle: Opp-Angle 是一个基于

$$ang_x^O = \arctan \left(\frac{O1_x}{O2_x} \right)$$

36 维直方图表示法[22]，其中下标 x 表示空间导数。它对于镜面反射，阴影遮蔽和动态模糊不变。

S0: 最后，我们考虑到张等人[26]仿生描述。这个色彩表示法是对比于颜色通道基于中心环绕滤波器完成的。

4.3. 实验 1: 颜色特征评估

表 1 展示了在 4.2 节中讨论的关于颜色特征的结果。所有的颜色表示都适当地归一化。我们在颜色表示中加入了一个没有亮度分量的亮度频道。这个亮度频道可以通过 Matlab 的函数“rgb2gray”来计算。传统的码移键控跟踪算法只是带有亮度。它给出了 54.5% 这样一个一般的距离精度。36 维表示法 HUE 和 Opp-Angle 都表现极差。最好的结果是通过 10 维色名获得的，同时比传统算法增加了 19.5%。同样，基于亮度的码移键控算法给出了一个 50.3 个像素的中值中心位置误差，而用携带色名算法给出的最好结果是 16.9 个像素的中值中心位置误差。

总之，颜色联合亮度确实可以改善性能。然而，仔细选择颜色属性对于获得显著地性能改进是至关重要的。最好的结果是用色名得到的。

	Int	Int + RGB	LAB	YCbCr	Int + rg	Opponent
median DP	54.5	49.1	65.9	48.6	50.6	57.6
median CLE	50.3	39.3	19.4	46.3	38.5	25.5

表 1: 比较不同的跟踪算法的颜色处理方法。最好的两个结果分别用红色和蓝色标记出来。传统的亮度频道加在没有固有亮度分量的颜色表示法上。结果用 41 个图像阵列的中值距离精度和中心位置误差表示。两种指标的最好情况都是通过用色名得到的。

4.4.实验 2: 鲁棒更新方案

这个实验显示多通道颜色特性对于已改进的更新方案的影响。我们参考颜色特征作为颜色和亮度的联系。图 2 显示了用被改进的更新方案⁵在中等距离精度获得的性能增长。从 9 到 11 评估的颜色特性，已被改进的更新方案提高了跟踪算法的性能。这种增长在高位的颜色属性像 HUE 和 Opp-Angle 尤其显著。因此，最佳性能用色名再次获得。其结果从 74% 增长至 81.4%。



图 2: 比较原始的更新方式和改进的方法在中等距离精度的差距。我们的方法提高了大多数颜色处理方式的性能。最好的结果是用色名改进的学习方法。

4.5.实验 3: 低维自适应颜色属性

正如前面提到的，一个跟踪方法的计算费用是对大多数实际应用来说是决定性因素。然而，一个低计算花费在不影响精确度的情况下是合适的。在这篇文章中，我们也提出了低维自适应颜色属性。维度减少技术已经在 3.3 节中介绍了。它被用于把 10 维色名减小至 2 维⁶。表 2 显示出用低维自适应颜色属性改进过的方法获得的结果和它与色名的比较。结果清晰的显示出改进后的方法在速度上有显著提高，并且保持了原有的竞争力绩效。

Method	Dimensions	median DP	median CLE	median FP
CN	10	81.4	13.8	78.9
CN ₂	2	79.3	14.3	105

表 2: 改进后的方法 CN₂ 和色名 CN 的比较结果。我们提供了中等距离精度和中心位置误差两个指标的结果。注意到 CN₂ 速度更快，同时在精度上稍有减弱。

4.6.与顶尖技术的比较

我们把我们的方法同 15 种不同的顶尖的跟踪算法相比较，结果在书面上取得了极好的效果。被用来比较的方法有:CT[27],TL D[11],DFT[20],EDFT[6],ASLA[10],L1APG [2],CSK[9],SCM[28],LOT[18],CPF[19],CXT [5],Frag[1],Struck[7],LSHT[8],LSST[24]。除了 LSST,LSHT 和 EDFT 其他所有的编码和二进制都由基准评价提供²。

表 3 显示了上述算法在 41 个具有挑战性的图像序列在中值中心位置误差和距离精度方面的比较。我们也报告了每秒中间帧数的速度。最好的两组结果用红色和蓝色标出。我们的方法显著提高了基于亮度的码移键控跟踪算法的性能，表现在中值位置误差相对减少到 72%。而且，我们的 CN 跟踪算法也将基线法的中等距离精度从 54.5% 增加到 8

1.4%。Struck 算法在近期评估中表现出最好的性能，在我们的评估中同样比其他现存的方法做的好。尽管如此，我们的 CN 跟踪算法的简单性，它同样在中等距离精度上比 Struck 算法高出 10%，同时，在操作速度上也达到了 7 倍甚至更高的帧速率。最后，结果还是表现出我们的算法在没有明显损失精度的同时，显著提升了速度。

	CT [2]	LSS [3]	Frag [4]	LIAPG [5]	LOI [6]	ASLA [7]	TLD [8]	SCM [9]	EDFT [10]	CSK [11]	LSHT [12]	ASLA [13]	TLD [14]	GKT [15]	LOT [16]	CN [17]	CN ₂ [18]	
Median CLE	78.4	78.4	70.8	62.9	60.9	56.8	54.4	54.3	34.1	33.7	34.9	34.5	31.4	39.5	37.1	55.9	71.3	78.9
Median DP	20.8	23.4	38.7	28.9	37.1	42.2	45.4	45.4	89.0	89.0	89.0	89.0	89.0	89.0	89.0	89.0	89.0	81.4
Median FPS	68.9	3.57	3.34	1.03	0.467	0.946	20.7	20.7	20.7	20.7	20.7	20.7	20.7	20.7	20.7	20.7	20.7	78.9

表 3: 我们的跟踪算法和 15 种顶尖算法在这 41 张图像阵列中的定量比较。结果报告了中等距离精度和中心位置误差两个参数⁴。我们还提供了每秒中间帧数的速度。最好的两个结果分别用红色和蓝色标记。这两种被改进的方法 CN 和 CN₂ 显示出最优性能。我们注意到 CN₂ 处理方法是在速度和精度方面位居第二。

图 3 显示所有 41 个序列的精度和成功图，其中包含平均距离和重叠精度。传说中的值是距离精度在 20 像素的平均值和曲线下面积。只有前 10 种跟踪算法可以清晰地显示出结果。在精度图方面，两种最好的方法是在本文中改进的 CN 和 CN₂。在阈值为 20 像素的平均距离精度上，我们的 CN 方法比 Struck 高出 3.5%，比基线 CSK 跟踪算法高 14.8%。值得提到的是，基线码移键控算法不能判断形状变化。尽管存在固有限制，比较其他顶尖的方法，在，平均重叠精度上，我们的两种方法提供了最有可能的结果。图 4 显示了我们的 CN₂ 的方法和现有的跟踪算法在逐帧动画方面的比较。比较对象是 5 个示例序列的中心像素误差。我们的方法比其他跟踪算法表现更优。

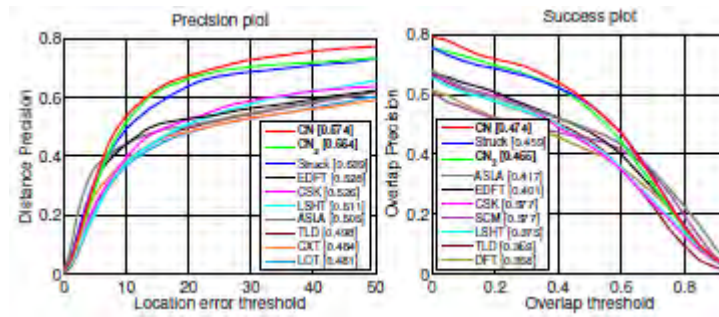


图 3: 41 个序列精度和成功图。每个跟踪算法的精度平均分数在图中显示出来。我们的两种方法用黑色显示出来。注意到我们的 CN 跟踪算法在平均距离精度上比码移键控跟踪算法提高了 14.8%。在两种情形下，我们的方法都比其他的顶尖跟踪算法表现更好。

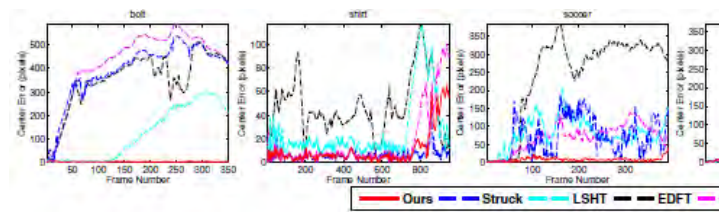


图 4: 我们的 CN₂ 方法同现存的方法在逐帧动画方面关 5 个实力序列的比较。图像显示的是像素的中心位置错误。对比于其他的顶尖的方法，我们的方法提供了更令人满意的后果。

鲁棒性的初始化: 众所周知，视觉跟踪算法对于初始化很敏感。为了评估初始化的鲁棒性，我们采用了在基准评价[25]中的方案。追踪算法通过在不同的帧（参考时间鲁棒性）和不同的位置（参考空间鲁棒性）进行初始化来判断。对于空间鲁棒性，对于每个序列需要评估 12 个不同的初值，而关于时间鲁棒性每个序列被分成 20 个部分。

我们选择了现存的跟踪算法中的在距离和重叠精度排前 5 个的算法来进行时间和空间鲁棒性的实验。我们的方法和选出来的方法的比较结果在图 5 中显示出来。在两种评估中，我们的 CN 和 CN₂ 方法都取得了最好的结果。

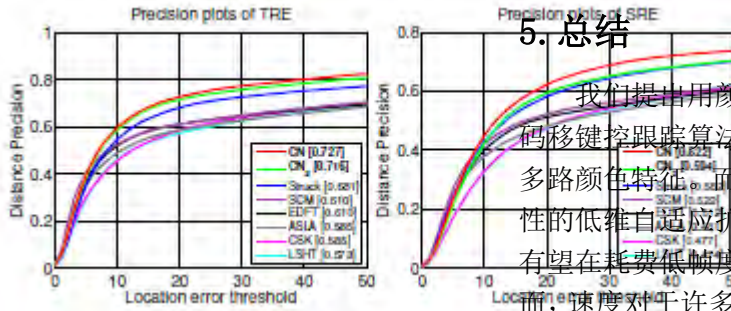


图 5: 时间鲁棒性和空间鲁棒性的精确图。我们的方法在两个评估中显示出最好的表现。

我们也通过 VOT 挑战评估方法论来评估视觉算法。这种方法与时间鲁棒性标准相似。关于这 41 个序列, 电痕破坏的平均数 (1.05), 我们的方法比 Struck 更低 (2.64)。

基于属性的评估: 很多因素都可能影响一个视觉跟踪算法的表现。在最近的基准评价中, 图像序列被 11 种不同的属性注释, 即: 光照变化, 尺度变换, 闭塞, 形变动态模糊快动作, 平面内旋转, 平面外旋转, 超出视野, 背景杂斑和低分辨率。对于 35 个注释[25]有上述属性的序列, 我们同其他方法进行了一个比较。我们的方法从 7 到 11 这些属性上表现出色: 背景杂斑, 动态变换, 形变, 光照变化, 平面内旋转, 平面外旋转和闭塞。

图 6 显示示例不同属性的精度图, 只有前 10 种能显示清晰的结果。对于光照变化序列, CN 和 CN₂ 方法, 对比于其他方法都显示出优秀的结果。这是由于颜色属性有一定程度上的测光不变性, 同时保留了判别力。一般的, 我们的方法不会处理超出视野范围的情况。这种情况, LOT 算法表现最好。

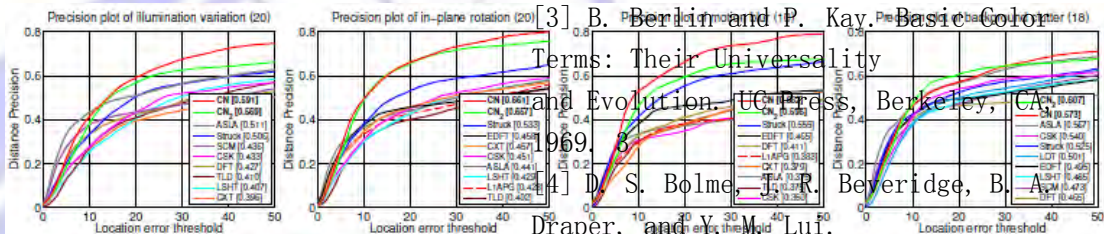


图 6: 不同属性的精度图。即: 光照变化, 平面内旋转, 背景杂斑。出现在标题的值表示视频的数量与各自相关的属性。本文提到的两种方法对比于顶尖算法表现出色。

5. 总结

我们提出用颜色属性进行跟踪。我们为码移键控跟踪算法扩展了学习机制来适应多路颜色特征。而且, 我们提出一个颜色属性的低维自适应扩展。一些现存的跟踪算法有望在耗费低帧度的同时显著提高精度。然而, 速度对于许多实际应用程序例如机器人学和实时监督来说, 也是重要指标。我们的方法保持了顶尖算法的精度, 同时速度是 FPS 算法的 100 倍以上。这使它尤其适用于现实应用程序。

尽管颜色被频繁的用于早期跟踪作品, 但是大多数现代工作主要应用简单的颜色变换。这篇论文论证了认真选择色彩转换的重要性。我们希望这篇文章可以促进研究人员认识到颜色的重要性, 并把它作为跟踪设计的不可分割的部分。

鸣谢: 这篇文章由 SSF 通过对工程 CUAS 的拨款进行支持, 由 VR 通过对工程 ETT 的拨款进行支持, 通过对 ICT 研究 ELLIT 和 CAD IS 战略支持。

参考:

[1] A. Adam, E. Rivlin, and Shimshoni. Robust fragments-based tracking using the integral histogram. In CVPR, 2006. 6

[2] C. Bao, Y. Wu, H. Ling, and H. Ji. Real time robust l1 tracker using accelerated proximal gradient approach. In CVPR, 2012. 1, 6

[3] B. Berlin and P. Kay. Basic Color Terms: Their Universality and Evolution. UC Press, Berkeley, CA, 1969. 8

[4] D. S. Bolme, J. R. Beveridge, B. A. Draper, and Y. M. Lui. Visual object tracking using adaptive correlation filters. In CVPR, 2010. 3

[5] T. B. Dinh, N. Vo, and G. Medioni. Context tracker: Exploring

- supporters and distracters in unconstrained environments. In CVPR, 2011. 1, 6
- [6] M. Felsberg. Enhanced distribution field tracking using channel representations. In ICCV Workshop, 2013. 1, 6
- [7] S. Hare, A. Saffari, and P. Torr. Struck: Structured output tracking with kernels. In ICCV, 2011. 1, 2, 6
- [8] S. He, Q. Yang, R. Lau, J. Wang, and M.-H. Yang. Visual tracking via locality sensitive histograms. In CVPR, 2013. 1, 6
- [9] J. Henriques, R. Caseiro, P. Martins, and J. Batista. Exploiting the circulant structure of tracking-by-detection with kernels. In ECCV, 2012. 1, 2, 4, 5, 6
- [10] X. Jia, H. Lu, and M.-H. Yang. Visual tracking via adaptive structural local sparse appearance model. In CVPR, 2012. 6
- [11] Z. Kalal, J. Matas, and K. Mikolajczyk. P-n learning: Bootstrapping binary classifiers by structural constraints. In CVPR, 2010. 1, 6
- [12] F. S. Khan, R. M. Anwer, J. van de Weijer, A. Bagdanov, A. Lopez, and M. Felsberg. Coloring action recognition in still images. IJCV, 105(3):205 - 221, 2013. 1, 3
- [13] F. S. Khan, R. M. Anwer, J. van de Weijer, A. Bagdanov, M. Vanrell, and A. Lopez. Color attributes for object detection. In CVPR, 2012. 1, 3
- [14] F. S. Khan, J. van de Weijer, and M. Vanrell. Modulating shape features by color attention for object recognition. IJCV, 98(1):49 - 64, 2012. 1, 2, 3
- [15] J. Kwon and K. M. Lee. Tracking by sampling trackers. In ICCV, 2011. 1
- [16] B. Liu, J. Huang, L. Yang, and C. Kulikowski. Robust tracking using local sparse appearance model and k-selection. In CVPR, 2011. 1
- [17] K. Nummiaro, E. Koller-Meier, and L. J. V. Gool. An adaptive PAMI, 32(9):1582 - 1596, 2010. 1, 5
- [18] S. Oron, A. Bar-Hillel, D. Levi, and S. Avidan. Locally orderless tracking. In CVPR, 2012. 1, 6
- [19] P. Perez, C. Hue, J. Vermaak, and M. Gangnet. Color-based probabilistic tracking. In ECCV, 2002. 1, 6
- [20] L. Sevilla-Lara and E. G. Learned-Miller. Distribution fields for tracking. In CVPR, 2012. 1, 6
- [21] K. van de Sande, T. Gevers, and C. G. M. Snoek. Evaluating color descriptors for object and scene recognition. [22] J. van de Weijer and C. Schmid. Coloring local feature extraction. In ECCV, 2006. 1, 5
- [23] J. van de Weijer, C. Schmid, J. J. Verbeek, and D. Larlus. Learning color names for real-world applications. TIP, 18(7):1512 - 1524, 2009. 2, 3
- [24] D. Wang, H. Lu, and M.-H. Yang. Least soft-threshold squares tracking. In CVPR, 2013. 6
- [25] Y. Wu, J. Lim, and M.-H. Yang. Online object tracking: A benchmark. In CVPR, 2013. 2, 4, 5, 7
- [26] J. Zhang, Y. Barhomi, and T. Serre. A new biologically inspired

- color image descriptor. In ECCV, 2012. 1, 5
- [27] K. Zhang, L. Zhang, and M. Yang. Real-time compressive tracking. In ECCV, 2012. 1, 6
- [28] W. Zhong, H. Lu, and M.-H. Yang. Robust object tracking via sparsity-based collaborative model. In CVPR, 2012. 6
- color-based particle filter. IVC, 21(1):99 - 110, 2003. 1

