

指导教师： 杨 涛

提交时间： 2016/3/19

CVPR2015 Paper

Translation

No: 01

姓名： 李刘洋

学号： 2013302575

班号： 10011305



有界相关滤波器

Hamed Kiani
Galoogahi
意大利理工学院

Terence Sim
新加坡国立大学
新加坡

Simon Lucey
卡耐基梅隆大学
美国匹兹堡

(译文: 李刘洋 西北工业大学 中国西安)

摘要

相关滤波器的优势在于傅里叶域有效性, 它可以进行预估: 频域中 $O(ND \log D)$, 对应空间中 $O(D^3 + ND^2)$, D 表示信号长度, N 表示信号数量。由于其鲁棒性和新引入的计算性能, 近期相关滤波器的发展, 例如 MOSSE, 重新燃起了他们在视觉社区领域的兴趣。然而, 我们在这篇论文中演示这种计算性能是有代价的。特别地, 我们论证只有 $\frac{1}{D}$ 比例的转换用例在边界效应上无影响, 这在追踪侦查性能上的表现是戏剧性的。在本文中, 我们计划一个新奇的相关滤波器来估计: (i) 利用固有的频域计算冗余性的优点, (ii) 有效地减少边界效应, (iii) 在学习过程中隐含地密集提取所有可能的训练用例补丁。用骄人的对象追踪和检测结果提出精确性和可计算性的功效。

1. 介绍

两信号的相关性是标记和追踪的一个标准方法。相关性与几乎所有计算机视觉方面的模式检测和对象追踪有关。相关性很少在空间域表现无效。相反地, 快速傅里叶变换负担了关联一个期望模板/滤波器和一个信号的有效应用。

相关滤波器最先由 Hester and Casasent 做了奠基工作, 作为一个频域学习滤波器在八九十年代得到重大发展。尽管很多变种被提及, 但是其方法核心仍然是学习一个滤波器, 当与一系列训练信号相关联时, 给出一个期望的响应。例如图 1 (b)。相关性的核心优势是这个方法尝试在由于频域相关效益选择在频域进行滤波学习。

相关滤波器已经在视觉领域重燃希望得益于 Bolme et al 在最小平方误差和 (MOSSE) 相关滤波器用于对象识别和追踪方面的近期工作。Bolme et al 的工作可以绕开相关滤波器方面的一些传统问题并且在转动, 刻度, 光线等条件变化下表现不错。核心的优势是相关滤波器在存储和计算方面极高的效率。

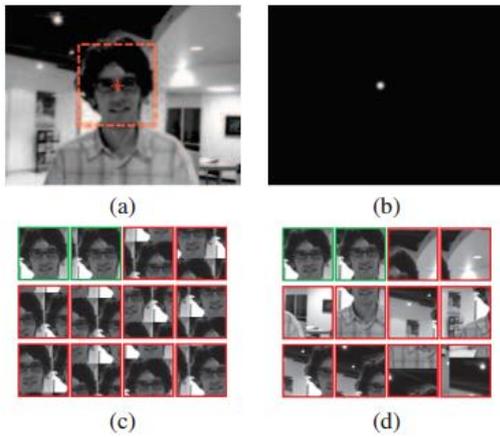


图 1

(a) 在图片相关性顶点应当出现的地方定义一个混合空间支持样例。(b) 相关滤波器应用与全图

(a) 图基础上的期望响应。(c) 典型相关滤波器的应用补丁例子集。与 (b) 图的一致性，绿色代表非零相关输出，红色代表零相关输出。

(d) 一个补丁子集用我们所期望的相关滤波器。注意我们所期望的方法使用所有可能的补丁填充塞图片的不同部分，这篇论文中心论点是如何在傅里叶域产出 (d) 图。(d) 图最后两张展示了 $\frac{D-1}{T}$ 部分接近图片边界被圆移所影响，在我们的方法中通过选择 $D \ll T$ 它可以被大大削减， D 、 T 指明在 (a) 图中向量化的脸部特征补丁长度和整个 (a) 图。

问题： 一个关于相关滤波器非常规的解释是区别模板已经从现实和合成实例的不平衡中预估出来。这些预估实例应用实际场景中的圆移形成，是不同传统转换方式的代表。我们使用的所有合成方式都被圆形边界影响所困扰，并不是具有代表性的移动样例（见图 1 (c)），结果就是学习这个模板的训练与实际场景在每一个 $D-1$ 合成样例中存在极大的不平衡（ D 代表样例维度）。这些边界效应可以戏剧性的影响预估模板表现。幸运的是，这些影响在相关滤波器组件轻微影响下大幅降低。但是现在需要解决空间域而非频域问题。不足的是，这个空间域的转移导致相关滤波器计算效率优势遭到破坏。这个挑战是本文集中考虑的一点。

贡献： 这篇文章我们做出了如下贡献：

- 我们提出了一个新的相关滤波器目标，是他能

够大幅度减少相关滤波器受边界效应影响的用例数量。

- 我们理论推导解决这个问题并减少其计算复杂度，对典型滤波器从 $O(D^3 + ND^2)$ 到 $O(ND \log D)$ ， D 代表矢量图长度， N 代表用例数量。
- 我们证明新目标如何有效最佳利用增广拉格朗日法 (ALM) 迭代计算，以致利用频域固有冗余优点。新方法的复杂度为 $O([N+K]T \log T)$ K 表示迭代次数， T 表示搜索窗大小。
- 我们展示我们的方法通过稠密抽样学习后，意味着它有能力合并所有可能起源于不同训练图部分的斑块，在固定大小存储下并且不靠路训练图和斑块的数量。（图 1 (d)）

相关工作： Bolme et al. 近期提出了一个传统相关滤波器的扩展提及到了最小平方误差总和 (MOSSE) 滤波器。这个方法被证明在对象追踪方面是难等可贵的，胜过了 2010 年提出的美术方法。引人注目的提高帧率从现在的 30fps 到 600fps 使得这个方法立刻引起视觉协会的兴趣。一个与 MOSSE 十分相关的算法也被 Bolme et al. 提出用于对象侦测和定位，提及到了平均合成精确滤波器 (ASEF) 它也报告了在图画展现方面的特殊表现。完整的关于其他相关滤波器的讨论，例如最优折中滤波器 (OTF) 无约束 MACE (UMACE) 滤波器，多通道相关滤波器 (MCCF)，内核 MOSSE 等，他们的应用不是本文讨论的范围，鼓励读者检查并完整课题讨论。

符号： 向量用小写黑体字母表示 (例如 \mathbf{a})，矩阵用大写黑体字母表示，(例如 \mathbf{A}) 常数标量用斜体表示 (例如 a 或 A)， $a(i)$ 表示向量 \mathbf{a} 的第 i 个元素。所有 M 型矩阵信号应当被表示成 \mathbf{a} 。卷积运算用 $*$ 表示。 M 维矩阵 \mathbf{a} 的 M 维 $\Delta\tau$ 离散循环移位表示成 $\mathbf{a}[\Delta\tau]$ 。矩阵 \mathbf{I} 表示 $D \times D$ 单位矩阵， $\mathbf{1}$ 表示 D 维单位向量， $\hat{\mathbf{a}}$ 表示一个向量经过 M 型离散傅里叶变换 (DFT) 可以表示为 $\hat{\mathbf{a}} \leftarrow \mathbf{F}(\mathbf{a}) = \sqrt{D} \mathbf{F} \mathbf{a}$ ， $\mathbf{F}()$ 是傅里叶变换运算符， \mathbf{F} 是 $D \times D$ 用于描绘傅里叶域的矢量化图片、信号的标准正交基向量。还有，我们用已有的表示 $\text{diag}(\hat{\mathbf{h}}) \hat{\mathbf{a}} = \hat{\mathbf{h}} \circ \hat{\mathbf{a}}$ ，其中 \circ 表示 Hadamard 卷积， $\text{diag}()$ 是 D 维向量转换

为 $D \times D$ 对角矩阵的运算符，滤波器 \hat{h} 和信号 \hat{a} 在这个性质下可以互换，复共轭的转置运算符 \dagger 在本文中还表示 Hermitian 伴随矩阵的相似性， $\text{conj}(\hat{a})$ 用于复共轭向量 \hat{a} 的共轭阵。

2. 相关滤波器

由于相关性在频域的效率，相关滤波器正式在频域被提出。没有什么可以再空间域抵抗相关滤波器（除了计算开销）。事实上，在空间域中观察相关滤波器可以给我们现有的相关滤波器理论基本问题的重要思想。

MOSSE 相关滤波器在空间域可以解决回归问题让人印象深刻：

$$E(\mathbf{h}) = \frac{1}{2} \sum_{i=1}^N \sum_{j=1}^D \|\mathbf{y}_i(j) - \mathbf{h}^\top \mathbf{x}_i[\Delta\tau_j]\|_2^2 + \frac{\lambda}{2} \|\mathbf{h}\|_2^2 \quad (1)$$

$\mathbf{y}_i \in \mathbb{R}^D$ 表示第 i 个观测样本的期望响应 λ 是规整算子。

$\mathbf{C} = [\Delta\tau_1, \dots, \Delta\tau_D]$ 代表长度为 D 的信号循环移位。Bolme et al. 提倡使用 2D 高斯小方差方法使得 \mathbf{y}_i 集中在对象的位置（通常是图像块的中心），这一目标的解决方案变成：

$$\mathbf{h} = \mathbf{H}^{-1} \sum_{i=1}^N \sum_{j=1}^D \mathbf{y}_i(j) \mathbf{x}_i[\Delta\tau_j] \quad (2)$$

当

$$\mathbf{H} = \lambda \mathbf{I} + \sum_{i=1}^N \sum_{j=1}^D \mathbf{x}_i[\Delta\tau_j] \mathbf{x}_i[\Delta\tau_j]^\top \quad (3)$$

在空间域解决相关滤波器问题很快变成了一个棘手的关于信号长度 D 的函数，方程 2 的复杂度变成 $O(D^3 + ND^2)$ 。

在此之外，就目前的计算成本问题而言，方程 1 描述的滤波器对转换错位非常敏感。当他试图检测或追踪一个变换域的对象时，其性能不受欢迎。这一敏感性被循环移位运算符 $\mathbf{x}[\Delta\tau]$ 求出，其中 $\Delta\tau = [\Delta x, \Delta y]^\top$ 代表二维循环移位 x, y ，它已经在相关文献 [19] 中指出，这种单循环移位产生的滤波器在

其他外观条件下没有一般性（如光照、角度、尺度、旋转等）。这种一般性问题可以通过选择非零标准化参数 λ ，或者使用 $N > 1$ 的集合训练代表着可能遇到的外观多样性的观测结果。

2.1 边界效应

然而目标方程 1 中更深层次的问题是偏移图像补丁 $\mathbf{x}[\Delta\tau]$ 在所有 $\Delta\tau \in \mathbf{C}$ 的取值中（除了 $\Delta\tau = 0$ 无平移的情况）不代表图像补丁会遇到一个正常的相关操作（如图 1 (c)）。在信号处理问题中，它通常指边界效应。绕过这个问题的简单办法是允许训练信号 $\mathbf{x} \in \mathbb{R}^T$ 的通过率大于滤波器 $\mathbf{h} \in \mathbb{R}^D$ ，就像 $T > D$ 。通过使用一个掩蔽矩阵 \mathbf{P} 可以变形方程 1 成为：

$$E(\mathbf{h}) = \frac{1}{2} \sum_{i=1}^N \sum_{j=1}^T \|\mathbf{y}_i(j) - \mathbf{h}^\top \mathbf{P} \mathbf{x}_i[\Delta\tau_j]\|_2^2 + \frac{\lambda}{2} \|\mathbf{h}\|_2^2 \quad (4)$$

0、1 组成的掩蔽矩阵 \mathbf{P} 将活跃/不活跃的信号封装起来。（图 2）增广方程 4 核心的有点在于增长比例不收边界效应影响。（典型滤波器中的 $\frac{1}{D}$ 被 $\frac{T-D+1}{T}$ 代替）在这一点上如果选择 $T \gg D$ 那么边界效应会被大幅度降低（图 1 (d)）。这个方法的计算复杂度为 $O(D^3 + NTD)$ 之比方程 1 的计算复杂度稍大，在实践中可以为 \mathbf{P} 建立一个查找表提高完成效率。

很明显在方程 4 中通过只对 T 可能循环位移的可能子集 $T-D+1$ 求和，边界效应可以被有效减少。然而在下面的部分我们将看到这样一个关于 \mathbf{P} 的变化是不可实现的，如果我们想在频域里有效解决这个问题的话。

2.2 频域中的效率

众所周知在信号处理中空间域的循环卷积可以被表示成频域 Hadamard 卷积。这使得我们可以将方程 1 表示的更简洁和平衡：

$$\begin{aligned} E(\hat{\mathbf{h}}) &= \frac{1}{2} \sum_{i=1}^N \|\hat{\mathbf{y}}_i - \hat{\mathbf{x}}_i \circ \text{conj}(\hat{\mathbf{h}})\|_2^2 + \frac{\lambda}{2} \|\hat{\mathbf{h}}\|_2^2 \quad (5) \\ &= \frac{1}{2} \sum_{i=1}^N \|\hat{\mathbf{y}}_i - \text{diag}(\hat{\mathbf{x}}_i)^\top \hat{\mathbf{h}}\|_2^2 + \frac{\lambda}{2} \|\hat{\mathbf{h}}\|_2^2 \end{aligned}$$

$\hat{h}, \hat{x}, \hat{y}$ 是 h, x, y 的傅里叶变换。 \hat{h} 的复杂共轭被用于确定这个方法是相关而不是卷积。方程 1 和 5 中的等值取决于另一个熟知的信号处理性质, 即 Parseval 原理:

$\mathbf{x}_i^T \mathbf{x}_j = \mathbf{D}^{-1} \hat{\mathbf{x}}_i^T \hat{\mathbf{x}}_j \forall i, j, \mathbf{x} \in \mathbb{R}^D$ 从而方程 5 的解决方法变为:

$$\begin{aligned} \hat{\mathbf{h}} &= [\text{diag}(\hat{s}_{xx}) + \lambda \mathbf{I}]^{-1} \sum_{i=1}^N \text{diag}(\hat{\mathbf{x}}_i) \hat{y}_i \quad (6) \\ &= \hat{s}_{xy} \circ^{-1} (\hat{s}_{xx} + \lambda \mathbf{1}) \end{aligned}$$

式中 \circ^{-1} 代表按元素除,

$$\hat{s}_{xx} = \sum_{i=1}^N \hat{\mathbf{x}}_i \circ \text{conj}(\hat{\mathbf{x}}_i) \quad \& \quad \hat{s}_{xy} = \sum_{i=1}^N \hat{y}_i \circ \text{conj}(\hat{\mathbf{x}}_i) \quad (7)$$

是平均训练观测值自然谱和正交谱。方程 1 和方程 5 中 \hat{h} 的解决方式是相同的 (其中一个表示在空间域, 另一个表示在频域) 这种理论的计算效率也很高。在频域可以找到一个 \hat{h} 的解决方案复杂度为 $O(N \log D)$ 。最小复杂度数在全体训练信号 $\{\mathbf{x}_i\}_{i=1}^N$ 和期望相应 $\{\mathbf{y}_i\}_{i=1}^N$ 中体现。

号 $\{\mathbf{x}_i\}_{i=1}^N$ 和期望相应 $\{\mathbf{y}_i\}_{i=1}^N$ 中体现。

3. 我们的解法

我们的解法出现于出现是当有人尝试应用相同傅里叶方法方程 4 洞察空间目标的复杂度时发现的, 方程 4 被表现在傅里叶域为:

$$E(\mathbf{h}) = \frac{1}{2} \sum_{i=1}^N \|\hat{y}_i - \text{diag}(\hat{\mathbf{x}}_i)^T \sqrt{D} \mathbf{F} \mathbf{P}^T \mathbf{h}\|_2^2 + \frac{\lambda}{2} \|\mathbf{h}\|_2^2 \quad (8)$$

不幸的是, 因为我们对 \mathbf{h} 强制加空间域约束 \mathbf{P}^T 使得这个方法复杂度膨胀到 $O(D^3 + ND^2)$ 使得 \mathbf{h} 必须在空间域被解决。

3.1 增广拉格朗日算法

我们提出解决方程 8 的算法涉及到了引入一个辅助变量 $\hat{\mathbf{g}}$, 这样, 方程 8 立刻变形为

$$\begin{aligned} E(\mathbf{h}, \hat{\mathbf{g}}) &= \frac{1}{2} \sum_{i=1}^N \|\hat{y}_i - \text{diag}(\hat{\mathbf{x}}_i)^T \hat{\mathbf{g}}\|_2^2 + \frac{\lambda}{2} \|\mathbf{h}\|_2^2 \\ \text{s.t. } &\hat{\mathbf{g}} = \sqrt{D} \mathbf{F} \mathbf{P}^T \mathbf{h} \quad (9) \end{aligned}$$

我们提及处理这个引入等值约束来自于一个

个增广拉格朗日算法 (ALM) 我们计划的增广拉格朗日算法目标可以被表示为:

$$\begin{aligned} \mathcal{L}(\hat{\mathbf{g}}, \mathbf{h}, \hat{\zeta}) &= \frac{1}{2} \sum_{i=1}^N \|\hat{y}_i - \text{diag}(\hat{\mathbf{x}}_i)^T \hat{\mathbf{g}}\|_2^2 + \frac{\lambda}{2} \|\mathbf{h}\|_2^2 \\ &\quad + \hat{\zeta}^T (\hat{\mathbf{g}} - \sqrt{D} \mathbf{F} \mathbf{P}^T \mathbf{h}) \\ &\quad + \frac{\mu}{2} \|\hat{\mathbf{g}} - \sqrt{D} \mathbf{F} \mathbf{P}^T \mathbf{h}\|_2^2 \quad (10) \end{aligned}$$

其中 μ 是补偿因素用于控制 ALM 算法聚合度, $\hat{\zeta}$ 是拉格朗日向量的傅里叶变换需要强制方程 9 的等值约束。ALM 算法在计算机视觉学习领域并不是新算法, 近期还有很多高效应用[7,8]。特别的, 交叉方向乘法器方法 (ADMMs) 已经证明了一个简单但是有用的算法非常适用于大量学习和视觉问题的分布式凸面最优法。对 ADMMs 的完整描述不是本文讨论的主要问题 (引文[7]完整讨论了该问题), 但是它可以被简单解释为为增广拉格朗日目标应用了一个高斯-赛德尔最优策略。这个策略将它分解成很好解决的子问题。可以在算法 1 中看到我们提出的算法的完整介绍, 我们详述每个子问题如下:

子问题 \mathbf{g} :

$$\begin{aligned} \hat{\mathbf{g}}^* &= \arg \min \mathcal{L}(\hat{\mathbf{g}}; \hat{\mathbf{h}}, \hat{\zeta}) \quad (11) \\ &= (\hat{s}_{xy} + \mu \hat{\mathbf{h}} - \hat{\zeta}) \circ^{-1} (\hat{s}_{xx} + \mu \mathbf{1}) \end{aligned}$$

其中 $\hat{\mathbf{h}} = \sqrt{D} \mathbf{F} \mathbf{P}^T \mathbf{h}$, 在实践中, 通过对零增补过的 \mathbf{h} (对 \mathbf{P}^T 隐含矩阵隐匿) 使用 $\text{FFT} \hat{\mathbf{h}}$ 可以被有效预估。

子问题 \mathbf{h} :

$$\begin{aligned} \mathbf{h}^* &= \arg \min \mathcal{L}(\mathbf{h}; \mathbf{g}, l) \quad (12) \\ &= \left(\mu + \frac{\lambda}{\sqrt{D}}\right)^{-1} (\mu \mathbf{g} + l) \end{aligned}$$

其中, $\mathbf{g} = \frac{1}{\sqrt{D}} \mathbf{P} \mathbf{F}^T \hat{\mathbf{g}}$, $l = \frac{1}{\sqrt{D}} \mathbf{P} \mathbf{F}^T \hat{\zeta}$.

在实践中, 通过应用反向 FFT 和隐匿矩阵 \mathbf{P} 的查找表可以快速有效预估 \mathbf{g} 和 l 。

更新拉格朗日乘子:

$$\hat{\zeta}^{(i+1)} \leftarrow \hat{\zeta}^{(i)} + \mu (\hat{\mathbf{g}}^{(i+1)} - \hat{\mathbf{h}}^{(i+1)}) \quad (13)$$

其中 $\hat{\mathbf{g}}^{(i+1)}$ 和 $\hat{\mathbf{h}}^{(i+1)}$ 是当前子问题迭代

$i+1$ 次计算结果 (应用迭代 ADMM)

μ 的选择: 一个简单公认的选择 μ 的方法

如下:

$$\mu^{(i+1)} = \min(\mu_{\max}, \beta\mu^{(i)}) \quad (14)$$

我们试验中发现 $\mu^{(0)} = 10^{-2}$, $\beta = 1.1$, $\mu_{\max} = 20$ 的计算结果不错。

3.2 密集取样学习

近期密集取样学习的好处已经被几个最近的方法完全探索[25,13,12]。这种方法计划训出一个分类器/探测器通过开拓所有可能的从训练样本中提取的负面补丁(例如, 100×100 的训练图中 10^4 补丁), 作为 HNM 的一个可行选择。应用 LDA 框架的密集采样策略, Hariharan et al.[12]演示了比 HNM 高级用法和计算更有效的探测表现。

检查方程 4 可以发现循环移位算子 $\Delta\tau$ 还原了所有训练图像 x_i 的移位版本(矢

量化的) $\{x_i[\Delta\tau_j]\}_{j=1}^T$, T 表示 x_i 的长度。注意我们方法的位移图像通过傅里叶域循环位移的卷积/相关性质隐藏的算出了, 在实践中, 我们不需要直接使用位移算子 $\Delta\tau$ 去计算位移图像(见方程 8)。通过在每个位移图像上使用隐含矩阵 P , 如方程 4 中

$Px_i[\Delta\tau_j]$ 我们需要从 $x_i[\Delta\tau_j]$ 选择(获取)一个补丁(子图像), 他的范围比 x_i 的范围小, 并且它是矢量化图像的第 j 个定位中心。这个方法算出所有 T 可能的训练图 x_i 的补丁/样本, 图 2. 由于方程 4 自带稠密取样, 我们通过优化这个方法学习相关滤波器, 这个计划的方法建立在稠密取样学习技术上。

3.3 计算复杂度

检查算法 1 会发现在 FFT 下主要影响每次 ADMM 最优方法迭代支出的是 $O(T\log T)$ 。在该算法中(在迭代步骤 4、5 之前), 有预估标准和交叉谱能量的参数向量 \hat{s}_{xx} 和 \hat{s}_{xy} 他的时间复杂度是 $O(NT\log T)$, N 表示训练信号的数量。我们已经给出了 K 代表 ADMM 迭代次数, 因

此算法总的复杂度为 $O([N+K]T\log T)$

3.4 存储效率

我们已经给出了 N 维向量化 T 长度的训练图, 我们的方法的训练一个相关滤波器的存储开销是 $O(T)$ 。这是计算标准和交叉谱能量向量 \hat{s}_{xx} 和 \hat{s}_{xy} 所需要的开销(在步骤 4、5, 方程 7 中)。这意味着我们方法的存储开销是固定的不受图片数量和 ADMM 迭代影响。还有下面要提到的, 密集抽样已经包含在我们的算法之中, 不需要额外的存储空间去加载所有可能的训练图片块。

我们通过给出一个例子强调这点好处。假设用 100000 个 100×100 的训练图(双精度浮点)训练处一个 50×50 的对象检测器(模板)。在这种情况下, 我们的方法使用了 $1000 \times 100 \times 100 = 10^7$ 块 50×50 的图像块去训练检测器(稠密抽样)这只需要 0.02MB 存储空间去计算 \hat{s}_{xx} 和 \hat{s}_{xy} 。另外, 学习一个广泛使用在识别任务的 SVM 分类器[26]引来了一个与图像块数量线性相关的存储支出, 所以, 使用 10^7 个 50×50 的图像块, 存储需求根本不需要 200GB。

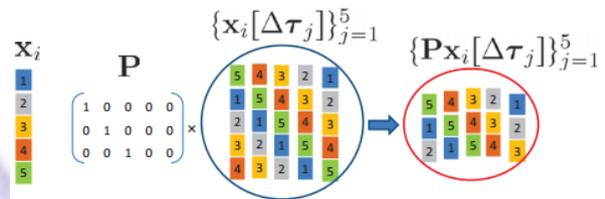


图 2, 稠密取样: 一个向量化的信号 x_i (左) 长度为 5, 所有可能的子信号(右) 长度 $D=3$ 通过乘一个 x_i 的位移矩阵(隐

匿) P 得出。 $\{x_i[\Delta\tau_j]\}_{j=1}^T$,

$\{Px_i[\Delta\tau_j]\}_{j=1}^T$, P 是一个 $D \times T$ 矩阵。

4. 实验

4.1 定位表现

在第一次实验中, 我们在人眼定位问题中测试了我们的算法, 与文献中的学习相

关滤波器进行了比较，例如 OTF [21], MACE [20], UMACE [24], ASEF [6], 还有 MOSSE [5]。CMU Multi-PIE 脸型数据库被用于这个实验。包含 900 张人脸正常光照正面照片。我们随机选择了 400 张图片用于训练这个测试。所有图片裁剪成 128×128 的相同大小，混合了双眼坐标。裁剪过的图片被标准化成 0 均值, 1.0 的标准偏差。

我们为其他方法使用裁剪过的 64×64 图片块，训练出了一个 64×64 的滤波器(右眼居中定位)。为我们的方法使用全脸图片 ($T=128 \times 128, D=64 \times 64$)。与 ASEF 和 MOSSE 相似，我们用 AD 高斯函数定义期望相应，空间方差 $s=2$ 。通过相关测试图片与滤波器，人眼定位追踪上了通过选择输出预测人眼定位波峰的结果。定位结果评估出了预估和期望的标准化的瞳距，

$$d = \frac{\|p_r - m_r\|_2}{\|m_l - m_r\|_2}$$

m_r, m_l 分别表示实际的左右眼坐标， p_r 是右眼的预测定位。标准化距离

$d < th$ 的定位被认为是成功的定位，门限值 th 是通过瞳距设定的。

十次运行的平均结果展示在图 3，如图我们的方法优于其他所有门限值和训练设置大小的方法。优与过渡拟合，训练图数量增长，OYF 和 MAC E 精度下降。实验中我们观察到 UMACE, ASEF 和 MOSSE 表现较差主要源于错误的那左眼和鼻子定位。这个问题很少出现在我们的方法中，因为我们的滤波器经历过稠密抽样图片训练。

一个视觉描述滤波器和它的输出可以再图 4 中看到。PSR 峰值展示了我们的方法得出了比其他滤波器更可靠的相关相应。

还有，我们检查了 T (训练图片大小) 对人眼定位的影响。在这里，我们使用裁剪过的不同大小的右眼图片，大小分别有 $T = \{D, 1.5D, 2D, 2.5D, 3D, 3.5D, 4D\}$ 训练滤波器大小为 32×32 。在 $t=D$ 的情况下，训练图只有右眼块，当 $T=4D$ 时，训练图成为全脸图，

结果展示在图 5 (a)，表明了最差的识别结果出现在 $T=D$ 时，定位精准度随着训练图相关滤波器的大小增长而上升。最好的定位率出现在 $T=4D$ 时，这是因为当 $T=4D$ 时，

(i) 图片块显著不受边界效应 $\left(\frac{T-D+1}{T}\right)$ 影响，(ii) 一系列的负面影响图片(如鼻子，嘴巴，左眼等)用来训练了强健的错误定位和背景部分的容错率。

算法 1: 使用 ADMM 的算法
1: 初始化 $h^{(0)}, l^{(0)}$ 。
2: 用 0 填补向量并且使用 FFT。
$\sqrt{D}FP^T h^{(0)} \rightarrow \hat{h}^{(0)}$
3: 应用 FFT: $\sqrt{D}Fl^{(0)} \rightarrow \hat{\zeta}^{(0)}$
4: 使用方程 7 估算自动谱能量 \hat{s}_{xx}
5: 使用方程 7 估算交叉谱能量 \hat{s}_{xy}
6: $i=0$
7: 重复
8: 使用方程 11 算出 $\hat{g}^{(i+1)}, \hat{h}^{(i)}$ 和 $\hat{\zeta}^{(i)}$
9: 结果逆 FFT:
$\frac{1}{\sqrt{D}}PF^T \hat{g}^{(i+1)} \rightarrow g^{(i+1)}$
10: 结果逆 FFT:
$\frac{1}{\sqrt{D}}PF^T \hat{\zeta}^{(i+1)} \rightarrow l^{(i+1)}$
11: 使用方程 12 算出 $\hat{h}^{(i+1)}, \hat{g}^{(i+1)}$ 和 $\hat{\zeta}^{(i)}$
12: 填充，应用 FFT
$\sqrt{D}FP^T h^{(i+1)} \rightarrow \hat{h}^{(i+1)}$
13: 更新拉格朗日乘子向量，方程 13
14: 更新补偿向量，方程 14
15: $i=i+1$
16: 直到 $\hat{g}, \hat{h}, \hat{\zeta}$ 收敛

4.2 运行时间表现

这个实验论证了我们的方法优与其他方法的优点。尤其是我们对比了我们方法比文献中提及的其他方法更高效的使用了迭代。我们与一个收敛最快的方法比较了收敛性从而优化我们的方法,图6中的结果表示,

(a) 收敛时间是滤波器大小和训练图数量的函数,(b) 迭代次数要求我们优化方程8中的对象。在(a)中我们深刻注意到在训练过程中收敛性如何与滤波器大小和训练图数量无关。这可以主要归因于自动谱和交叉谱能量的计算效率。结果是ADMM迭代不需要重新更改训练设置,使得我们的方法戏剧性的比其他稚嫩的迭代算法表现出色,因为他们在每一步迭代训练过程中重新计算空间域卷积。相似的,在(b)中没我们可以发现很少的迭代就可以得到较好的收敛性。

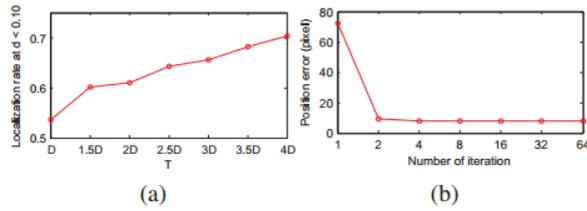


图 5: (a) 定位率是训练图大小 (T) 的函数, 滤波器大小为 $D=32 \times 32$ 。(b) ADMM 迭代的错误位置追踪。我们在训练表现和计算复杂度方面权衡后选择了 2 次迭代。

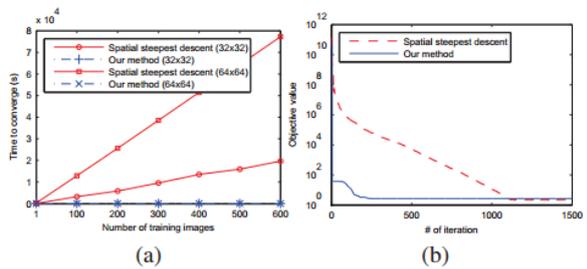


图 6: 我们的算法和收敛最快的算法[27]相比较的运行时间表现。我们的方法表现更好由于:(a)训练两个不同大小的滤波器使得收敛加速,(b)迭代收敛所需要的次数。

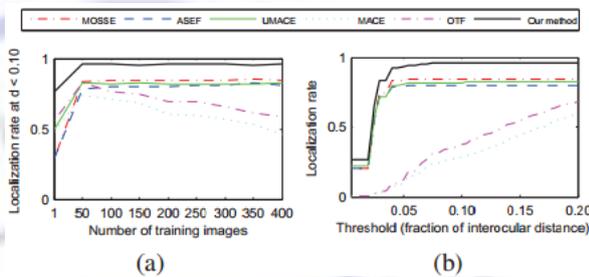


图 3: 人眼定位表现是与 (a) 训练图片数量、(b) 定位阈值相关的函数。

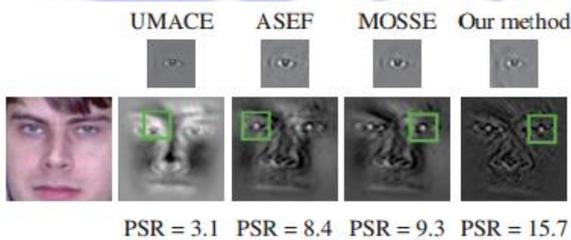


图 4: 一个人眼定位的例子。输出(下)是由使用 64×64 相关滤波器产生(上)的。绿色方块代表估计的右眼定位, PSNR 代表输出峰值的锐度。

4.3 追踪表现

最后,我们评估了推荐方法在实时追踪正常流畅视频中的表现,与文献中领先的追踪器进行了比较[3,11,9,10,23,5,13]。所有这些方法根据他们的参考文献调整了参数。 $m \times n$ 大小目标的期望响应被定义为有 $s = \sqrt{mn}/16$ 方差的 2D 高斯法[13]。规整参数 λ 设置为 10^{-2} 。我们评估了不同 ADMM 迭代次数 {1,2,4,8,16,32,64} 下方法的表现,像图 5 (b) 中展示的那样,最终我们选择了两次迭代。(在精度和追踪速度中做了一次权衡)。MOSSE 和我们的方法都使用了追踪初始化步骤,我们使用了八个随机仿射扰动初始化我们的首个滤波器。我们借用

了网上的 MOSSE 改编版[5]来改编我们滤波器第 i 帧，使用了平均自动谱和交叉谱：

$$\begin{aligned}(\hat{s}_{xx})^i &= \eta(\hat{x}_i \circ \text{conj}(\hat{x}_i)) + (1 - \eta)(\hat{s}_{xx})^{i-1} \\(\hat{s}_{xy})^i &= \eta(\hat{y}_i \circ \text{conj}(\hat{x}_i)) + (1 - \eta)(\hat{s}_{xy})^{i-1}\end{aligned}\quad (15)$$

η 为适应率，我们惊奇的发现 $\eta=0.025$ 适用于我们的方法快速适应对象姿势、大小、光照变化等。

表 1 中评估了追踪结果，建立在 (i) 精度：预计位置和实际位置差距在 20 个像素点的帧数百分比，(ii) 平均错误定位像素，还有 (iii) 追踪速度 (fps) 这是追踪论文[3][11][9]中使用的标准方法。我们的方法基本实现了最高精度和最少定位错误，追上了 STRUCK。原因是我们的方法结合了所有可能的前景（目标）和背景（非目标）视觉信息块去训练追踪器（稠密取样，例如一个 100×100 取 10^4 块）。由于计算性能限制，非滤波器方法如 STYUCK 和 MILTrack 只是用少数随机收集在目标定位周围的目标图像和非目标图像块（约 10-20 个）。

更重要的是，MOSSE 和 kernel-MOSSE 由于边界效应从而精度差于我们的方法，图 1 (c)。还有原因是他们没有使用背景图去训练他们的滤波器。在追踪速度方面 MOSSE 比我们的方法快 600fps，我们的方法得出的帧率低于 MOSSE，因为它使用了迭代的方法。然而我们追踪器的速度是 100fps 已经足够实时追踪的需求。一些视频的追踪结果展示在图 8 中，展示了我们方法在所有阈值下的高精度，以及更少的漂移帧。图 8、9 以外的更多展示可以在补充材料中找到。

请注意我们的实验的目的不是展现我们的方法比其他测试视频好，我们在 “Tiger1”

视频中求得较低的精度，只有 79%，与其他方法比较 MILTrack(94%)还有 STRUCK (95%)。这意味着与想起他相关滤波器相同基础的追踪器，我们的方法在 “Tiger1” 这样的全闭塞和严苛的场景转换视频中表现并不好，如图 7。实际上，我们意在展示我们结果的竞争性可以被简单实用的快速追踪器实现。与其他复杂缓慢的技术相比，我们的方法在对象追踪方面有较强的适应性。

5. 总结

这是一个在限制圆形边界效应方面表现引人注目的预估相关滤波器，他保留了典型频域相关滤波器计算效率高的优势。更重要的是，我们展示了这个方法隐含着嵌入的相关滤波器稠密抽样策略，它继承了频域卷积运算的循环移位性质。这使得人们可以通过利用一大堆负面图片训练出一个高效的侦测器/滤波器，并且保证高效使用存储空间，还受训练图和抽样块数量的影响。我们的方法演示了一个经过验证的高级结果，在对象检测和实时追踪方面都比当前发展水平高。

	MOSSE [5]	KMOSSE [13]	MILTrack [3]	STRUCK [11]	OAB(1) [9]	SemiBoost [10]	FragTrack [1]	Our method
FaceOcc1	{ 1.00 , 7}	{ 1.00 , 5}	{0.75, 17}	{0.97, 8}	{0.22, 43}	{0.97, 7}	{0.94, 7}	{ 1.00 , 8}
FaceOcc2	{0.74, 13}	{0.95, 8}	{0.42, 31}	{0.93, 7 }	{0.61, 21}	{0.60, 23}	{0.59, 27}	{ 0.97 , 7}
Girl	{0.82, 14}	{0.44, 35}	{0.37, 29}	{ 0.94 , 10 }	-	-	{0.53, 27}	{0.90, 12}
Sylv	{0.87, 7}	{ 1.00 , 6}	{0.96, 8}	{0.95, 9}	{0.64, 25}	{0.69, 16}	{0.74, 25}	{ 1.00 , 4}
Tiger1	{0.61, 25}	{0.62, 25}	{0.94, 9}	{ 0.95 , 9 }	{0.48, 35}	{0.44, 42}	{0.36, 39}	{0.79, 18}
David	{0.56, 14}	{0.50, 16}	{0.54, 18}	{0.93, 9}	{0.16, 49}	{0.46, 39}	{0.28, 72}	{ 1.00 , 7}
Cliffbar	{0.88, 8}	{0.97, 6}	{0.85, 12}	{0.44, 46}	{0.76, -}	-	{0.22, 39}	{ 1.00 , 5}
Coke Can	{0.96, 7}	{ 1.00 , 7}	{0.58, 17}	{0.97, 7}	{0.45, 25}	{0.78, 13}	{0.15, 66}	{0.97, 7}
Dollar	{ 1.00 , 4}	{ 1.00 , 4}	{ 1.00 , 7}	{ 1.00 , 13}	{0.67, 25}	{0.37, 67}	{0.40, 55}	{ 1.00 , 6}
Twinnings	{0.48, 16}	{0.89, 11}	{0.76, 15}	{ 0.99 , 7 }	{0.74, -}	-	{0.82, 14}	{ 0.99 , 9}
<i>mean</i>	{0.80, 11}	{0.84, 12}	{0.72, 16}	{0.91, 12}	{0.53, 31}	{0.62, 29}	{0.51, 37}	{ 0.97 , 8}
<i>fps</i>	600	100	25	11	25	25	2	100

表 1: 追踪效果展示成一个二元组{20 像素的精度, 平均像素精度错误}, 在这里我们的方法实现了 10 个例子中的 8 个最优。最好的帧率被 MOSSE 获得, 我们的方法通过两次 ADMM 迭代实现了 100fps 实时追踪, 每次试验的最佳表现用黑色加粗标注。

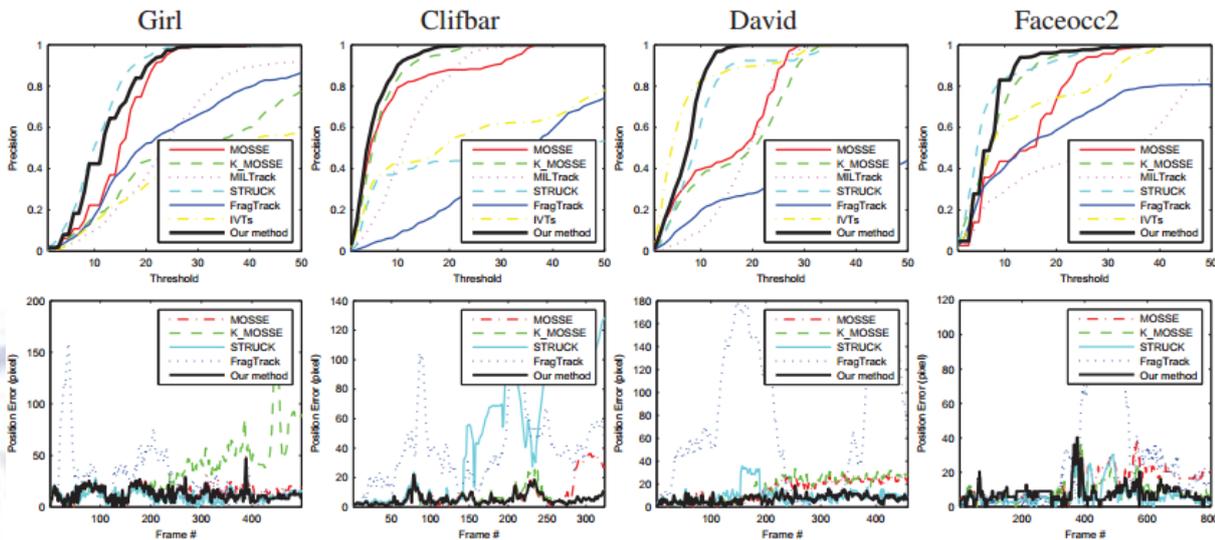


图 8: 所选视频的追踪结果, (a) 阈值与精度的对比 (b) 每帧的定位错误



图 9: 我们方法在极限复杂姿势、大小、光照和局部条件下对两个视频对象追踪的结果。蓝色(虚线)框和红框分别代表实际位置和我们方法的预测位置。每一帧我们都样例输出了目标和训练滤波的相关性。

参考文献:

- [1] A. Adam, E. Rivlin, and I. Shimshoni. Robust fragments based tracking using the integral histogram. In CVPR, 2006.
- [2] B. Babenko, M. H. Yang, and S. Belongie. Visual tracking with online multiple instance learning. In CVPR, 2009.
- [3] B. Babenko, M.-H. Yang, and S. Belongie. Robust object tracking with online multiple instance learning. PAMI,33(8):1619–1632, 2011.
- [4] V. N. Boddeti, T. Kanade, and B. Kumar. Correlation filters for object alignment. In CVPR, 2013, pages 2291–2298. IEEE, 2013.
- [5] D. S. Bolme, J. R. Beveridge, B. A. Draper, and Y. M. Lui. Visual object tracking using adaptive correlation filters. In CVPR, 2010.
- [6] D. S. Bolme, B. A. Draper, and J. R. Beveridge. Average of synthetic exact filters. In CVPR, 2009.
- [7] S. Boyd. Distributed Optimization and Statistical Learning via the Alternating Direction Method of Multipliers. Foundations and Trends in Machine Learning, 3:1–122, 2010.
- [8] A. Del Bue, J. Xavier, L. Agapito, and M. Paladini. Bilinear modelling via augmented lagrange multipliers (BALM). PAMI, 34(8):1–14, Dec. 2011.
- [9] H. Grabner, M. Grabner, and H. Bischof. Real-time tracking via on-line boosting. In BMVC, 2006.
- [10] H. Grabner, C. Leistner, and H. Bischof. Semi-supervised on-line boosting for robust tracking. In ECCV. Springer, 2008.
- [11] S. Hare, A. Saffari, and P. H. Torr. Struck: Structured output tracking with kernels. In ICCV, 2011.
- [12] B. Hariharan, J. Malik, and D. Ramanan. Discriminative decorrelation for clustering and classification. In ECCV 2012, pages 459–472. Springer, 2012.
- [13] J. F. Henriques, R. Caseiro, P. Martinez, and J. Batista. Exploiting the circulant structure of tracking-by-detection with kernels. In ECCV, 2012.
- [14] J. F. Henriques, R. Caseiro, P. Martins, and J. Batista. Highspeed tracking with kernelized correlation filters. PAMI, 2014.
- [15] C. F. Hester and D. Casasent. Multivariate technique for multiclass pattern recognition. Appl. Opt., 19(11):1758–1761, 1980.
- [16] H. Kiani, T. Sim, and S. Lucey. Multi-channel correlation filters. In ICCV, 2013.
- [17] H. Kiani, T. Sim, and S. Lucey. Multi-channel correlation filters for human action recognition. In ICIP, 2014.
- [18] B. V. K. V. Kumar. Minimum-variance synthetic discriminant functions. J. Opt. Soc. Am. A, 3(10):1579–1584, 1986.
- [19] B. V. K. V. Kumar, A. Mahalanobis, and R. D. Juday. Correlation Pattern Recognition. Cambridge University Press, 2005.
- [20] A. Mahalanobis, B. V. K. V. Kumar, and D. Casasent. Minimum average correlation energy filters. Appl. Opt., 26(17):3633–3640, 1987.
- [21] P. Refregier. Optimal trade-off filters for noise robustness, sharpness of the correlation peak, and hornor efficiency. Optics Letters, 16:829–832, 1991.
- [22] A. Rodriguez, V. N. Boddeti, B. V. Kumar, and A. Mahalanobis. Maximum margin correlation filter: A new approach for localization and classification. TIP, 22(2):631–643, 2013.
- [23] D. Ross, J. Lim, R. Lin, and M. Yang. Incremental learning for robust visual tracking. IJCV, 77(1):125–141, 2008.
- [24] M. Savvides and B. V. K. V. Kumar. Efficient design of advanced correlation filters for robust distortion-tolerant face recognition. In AVSS, pages 45–52, 2003.
- [25] J. Valmadre, S. Sridharan, and S. Lucey. Learning detectors quickly with stationary statistics. In ACCV, 2014.
- [26] P. Viola and M. Jones. Rapid object detection using a boosted cascade of simple features. In Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on, volume 1, pages I–511. IEEE, 2001.
- [27] M. Zeiler, D. Krishnan, and G. Taylor. Deconvolutional networks. CVPR, 2010.