Kernelized Correlation Filters Notes

FanmingL

2018 年 11 月 28 日

图 1: 左图为检测到物体的图像，需要寻找到右图中的物体

1 核相关滤波器

1.1 问题提出

若我们有两张图片，在两张图片中均有某物体，在第一张图片中检测到该物体，那么如何在第二张图片中去寻找该物体呢？

一个很直观的思想是：罗列出第一张图所有可能的平移变换，比如，这是一张 75×75 的图片，该图片的左上角在 $(0, 0)$ 处，移动图片，使得图片左上角移动至其余的 $75 \times 75 - 1$ 像素处，超出边界的像素补充至上边界和左边界。这样我们可以得到 75×75（包括原图）张图像，将图1中右图与这 75×75 张图像对比，计算图像的相关性，找到最大相关性的图像，该图像对应的平移便是图1左图中物体到图1右图中物体的平移。

这篇文章的思想也大抵如此，但是它用非常精妙的技巧将计算复杂度降低到了 $O(n \log n)$。

1.2 问题的数学表示

1.2.1 循环矩阵

如第1.1节所示，我们需要得到所有能够移动得到的图片，那么应该如何表示呢？

只需要将图像（为了方便起见，这里暂时用灰度图像）化为列向量 $\mathbf{x}$，然后对 $\mathbf{x}$ 做循环位移再转变回矩阵即可。循环位移即左乘 $\mathbf{P}$ 矩阵。

$$\mathbf{P} = \begin{bmatrix} 0 & 0 & 0 & \cdots & 1 \\ 1 & 0 & 0 & \cdots & 0 \\ 0 & 1 & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ 0 & 0 & \cdots & 1 & 0 \end{bmatrix} \tag{1}$$

可以看出 $\mathbf{Px} = [x_n, x_1, x_2, \cdots, x_{n-1}]^T$ 即为 $\mathbf{x}$ 向下移动一次得到的新向量。由此我们可以得到 n 个平移得到的向量。

$$\{\mathbf{P}^u \mathbf{x} | u = 0, \cdots, n-1\} \tag{2}$$

我们将所有这些向量拼成一个矩阵，即

$$\mathbf{X} = \begin{bmatrix} x_1 & x_2 & x_3 & \cdots & x_n \\ x_n & x_1 & x_2 & \cdots & x_{n-1} \\ x_{n-1} & x_n & x_1 & \cdots & x_{n-2} \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ x_2 & x_3 & x_4 & \cdots & x_1 \end{bmatrix} \tag{3}$$

我们定义操作 $C(\mathbf{x}) === X$，即以 $\mathbf{x}^T$ 为第一行构造循环矩阵。

1.2.2 用岭回归求解问题

我们尝试用岭回归求解第1.1节描述的问题，首先设 $\mathbf{x}$ 为原始的图像，$\mathbf{x}_i$ 定义为

$$\mathbf{x}_i = \mathbf{P}^{i-1}\mathbf{x} \quad i = 1, \cdots, n \tag{4}$$

下一步我们需要得到一个线性回归模型

$$f(\mathbf{z}) = \mathbf{z}^T\mathbf{w} \tag{5}$$

我们希望用它来判断 $\mathbf{z}$ 与原始图像的相关性大小。

为了构造训练集我们为每一个图像 $\mathbf{x}_i$ 给定一个与原始图像的相关系数，用 y_i 表示。给 y_i 赋值的原则是 $\mathbf{x}_i$ 偏移原始图像越远，相关性越小。y_i 可以用高斯函数去给定，即 $i = 0$ 时 $y_i = 1$ 其余 y_i 按照其图像左上角与原始图像左上角的距离按照高斯函数衰减（即偏的越远，y_i 越小，这是很好理解的）。

于是我们有了我们的训练样本 $\mathbf{X}$ 和标签 $\mathbf{y}$，接下来我们用岭回归去得到线性模型的系数 $\mathbf{w}$。

岭回归的优化目标是最小化经验风险和结构风险之和，即

$$\min \sum_i^n ((f(\mathbf{x}_i) - y_i)^2 + \lambda \|\mathbf{w}\|^2 \tag{6}$$

写成矩阵形式

$$\min J(\mathbf{w}) = (\mathbf{X}\mathbf{w} - \mathbf{y})^T(\mathbf{X}\mathbf{w} - \mathbf{y}) + \lambda \mathbf{w}^T\mathbf{w} \tag{7}$$

对 $J(\mathbf{w})$ 求偏导，并使偏导为零得到

$$\mathbf{w} = (\mathbf{X}^T\mathbf{X} + \lambda I)^{-1}\mathbf{X}^T\mathbf{y} \tag{8}$$

更一般的，对于 $\mathbf{X}$ 为复矩阵的情况

$$\mathbf{w} = (\mathbf{X}^H\mathbf{X} + \lambda I)^{-1}\mathbf{X}^H\mathbf{y} \tag{9}$$

至此我们用循环矩阵构造的训练集得到了我们的回归模型，对图1右图的检测也只需要将该图化为列向量 $\mathbf{z}$，然后计算 $\mathbf{f}(\mathbf{z}) = C(\mathbf{z})\mathbf{w}$，$\mathbf{f}(\mathbf{z})$ 最大值对应的平移就是右图中物体相对于左图中物体的估计位置。

但是很显然，这样一种运算代价实在是太高了，所以本文根据循环矩阵的特性，对上述计算大大化简。

1.3 用循环矩阵的特征值分解简化结果

1.3.1 离散傅立叶变换

若有一 N 维离散序列 $\mathbf{x}$ 其离散傅立叶变换可以写为

$$\hat{x}_k = \sum_{n=0}^{N} \exp(-j\frac{2\pi}{N}nk)x_n \tag{10}$$

其逆变换可写为

$$x_n = \frac{1}{N}\sum_{k=0}^{N} \exp(j\frac{2\pi}{N}nk)x_k \tag{11}$$

根据式10可以将式10写成矩阵形式即

$$\hat{\mathbf{x}} = \mathbf{G}\mathbf{x} \tag{12}$$

其中 $G_{kn} = \exp(-j\frac{2\pi}{N}nk)$。可以看出 $G_{kn} = G_{nk} = \exp(-j\frac{2\pi}{N}nk)$，所以

$$\mathbf{G} = \mathbf{G}^T \tag{13}$$

逆变换可以写为

$$\mathbf{x} = \frac{1}{N}\mathbf{G}^*\hat{\mathbf{x}} \tag{14}$$

我们可以来看看 $\mathbf{G}$ 是否是酉矩阵（即是否满足 $\mathbf{G}^H\mathbf{G} = I$）

$$\begin{aligned}\mathbf{x} &= \frac{1}{N}\mathbf{G}^*\hat{\mathbf{x}} \\ \mathbf{x} &= \frac{1}{N}\mathbf{G}^*\mathbf{G}\mathbf{x} \\ \mathbf{x} &= \frac{1}{N}\mathbf{G}^H\mathbf{G}\mathbf{x}\end{aligned} \tag{15}$$

即满足 $\frac{1}{N}\mathbf{G}^H\mathbf{G} = I$，为了之后推导方便，我们令

$$\mathbf{F} = \frac{1}{\sqrt{N}}\mathbf{G} \tag{16}$$

$\mathbf{F}$ 显然满足对称性

$$\mathbf{F}^T = \mathbf{F} \tag{17}$$

且 $\mathbf{F}$ 是酉矩阵

$$\mathbf{F}^H\mathbf{F} = \mathbf{F}\mathbf{F}^H = I \tag{18}$$

且基于 $\mathbf{F}$ 的离散傅立叶变换可写为

$$\hat{\mathbf{x}} = \mathcal{F}(\mathbf{x}) = \sqrt{N}\mathbf{F}\mathbf{x} \tag{19}$$

逆离散傅立叶变换为

$$\mathbf{x} = \mathcal{F}^{-1}(\hat{\mathbf{x}}) = \frac{1}{\sqrt{N}}\mathbf{F}^H\hat{\mathbf{x}} \tag{20}$$

式19、式20、式17、式18在之后的推导中将会多次使用!

1.3.2 循环矩阵的特征值分解

若某一方阵 $\mathbf{X}$ 的元素 X_{ij} 的值只与 $(i-j)$ 的大小有关，那么矩阵 $\mathbf{X}$ 为循环矩阵且，矩阵 $\mathbf{X}$ 可以进行特征值分解

$$\mathbf{X} = \mathbf{F}diag(\hat{\mathbf{x}})\mathbf{F}^H \tag{21}$$

其中 $\hat{\mathbf{x}} = \mathcal{F}(\mathbf{x})$，即 $\mathbf{x}$ 的离散傅立叶变换，$\mathbf{F}$ 即式16所定义的。

1.3.3 训练过程的简化

我们回到式9，用式21对其进行化简。我们有

$$\begin{aligned}\mathbf{X}^H\mathbf{X} &= \mathbf{F}diag(\hat{\mathbf{x}}^*)\mathbf{F}^H\mathbf{F}diag(\hat{\mathbf{x}})\mathbf{F}^H \\ &= \mathbf{F}diag(\hat{\mathbf{x}}^*)diag(\hat{\mathbf{x}})\mathbf{F}^H \\ &= \mathbf{F}diag(\hat{\mathbf{x}}^* \odot \hat{\mathbf{x}})\mathbf{F}^H\end{aligned} \tag{22}$$

所以

$$\begin{aligned}\mathbf{X}^H\mathbf{X} + \lambda I &= \mathbf{F}diag(\hat{\mathbf{x}}^* \odot \hat{\mathbf{x}})\mathbf{F}^H + \lambda\mathbf{F}\mathbf{F}^H \\ &= \mathbf{F}diag(\hat{\mathbf{x}}^* \odot \hat{\mathbf{x}} + \lambda)\mathbf{F}^H\end{aligned} \tag{23}$$

所以

$$(\mathbf{X}^H\mathbf{X} + \lambda I)^{-1} = \mathbf{F}diag(\frac{1}{\hat{\mathbf{x}}^* \odot \hat{\mathbf{x}} + \lambda})\mathbf{F}^H \tag{24}$$

$$\begin{aligned}(\mathbf{X}^T\mathbf{X} + \lambda I)^{-1}\mathbf{X}^H &= \mathbf{F}diag(\frac{1}{\hat{\mathbf{x}}^* \odot \hat{\mathbf{x}} + \lambda})\mathbf{F}^H\mathbf{F}diag(\hat{\mathbf{x}}^*)\mathbf{F}^H \\ &= \mathbf{F}diag(\frac{1}{\hat{\mathbf{x}}^* \odot \hat{\mathbf{x}} + \lambda})diag(\hat{\mathbf{x}}^*)\mathbf{F}^H \\ &= \mathbf{F}diag(\frac{\hat{\mathbf{x}}^*}{\hat{\mathbf{x}}^* \odot \hat{\mathbf{x}} + \lambda})\mathbf{F}^H\end{aligned} \tag{25}$$

所以

$$\begin{aligned}\mathbf{w} &= \mathbf{F}diag(\frac{\hat{\mathbf{x}}^*}{\hat{\mathbf{x}}^* \odot \hat{\mathbf{x}} + \lambda})\mathbf{F}^H\mathbf{y} \\ \mathbf{F}^H\mathbf{w} &= diag(\frac{\hat{\mathbf{x}}^*}{\hat{\mathbf{x}}^* \odot \hat{\mathbf{x}} + \lambda})\mathbf{F}^H\mathbf{y} \\ \mathbf{F}^T\mathbf{w}^* &= diag(\frac{\hat{\mathbf{x}}}{\hat{\mathbf{x}}^* \odot \hat{\mathbf{x}} + \lambda})\mathbf{F}^T\mathbf{y}^* \\ \mathbf{F}^T\mathbf{w} &= diag(\frac{\hat{\mathbf{x}}}{\hat{\mathbf{x}}^* \odot \hat{\mathbf{x}} + \lambda})\mathbf{F}^T\mathbf{y} \\ \mathbf{F}\mathbf{w} &= diag(\frac{\hat{\mathbf{x}}}{\hat{\mathbf{x}}^* \odot \hat{\mathbf{x}} + \lambda})\mathbf{F}\mathbf{y} \\ \mathcal{F}(\mathbf{w}) &= diag(\frac{\hat{\mathbf{x}}}{\hat{\mathbf{x}}^* \odot \hat{\mathbf{x}} + \lambda})\mathcal{F}(\mathbf{y})\end{aligned} \tag{26}$$

最后得到

$$\hat{\mathbf{w}} = diag(\frac{\hat{\mathbf{x}}}{\hat{\mathbf{x}}^* \odot \hat{\mathbf{x}} + \lambda})\hat{\mathbf{y}} \tag{27}$$

左乘对角阵是对行加权，即

$$\hat{\mathbf{w}} = \frac{\hat{\mathbf{x}} \odot \hat{\mathbf{y}}}{\hat{\mathbf{x}}^* \odot \hat{\mathbf{x}} + \lambda} \tag{28}$$

至此，我们将 $\mathbf{w}$ 的表达式化简至此，大部分的运算量都花在离散傅立叶变换上，而得益于快速傅立叶变换，时间复杂度为 $O(n\log n)$ 的算法，这个算法的时间复杂度也变成了 $O(n\log n)$。

1.3.4 检测过程的简化

于是训练过程可以用 $O(n\log n)$ 的算法代替，那么检测过程呢?

$$\begin{aligned}
\mathbf{f}(\mathbf{z}) &= C(\hat{\mathbf{z}})\mathbf{w} \\
&= \mathbf{F}diag(\hat{\mathbf{z}})\mathbf{F}^H\mathbf{w} \\
\mathbf{F}^H\mathbf{f}(\mathbf{z}) &= diag(\hat{\mathbf{z}})\mathbf{F}^H\mathbf{w} \\
\mathbf{F}^T\mathbf{f}(\mathbf{z}) &= diag(\hat{\mathbf{z}}^*)\mathbf{F}^T\mathbf{w} \\
\mathbf{F}\mathbf{f}(\mathbf{z}) &= diag(\hat{\mathbf{z}}^*)\mathbf{F}\mathbf{w}
\end{aligned} \tag{29}$$

最后得到

$$\hat{\mathbf{f}}(\mathbf{z}) = \hat{\mathbf{z}}^* \odot \hat{\mathbf{w}} \tag{30}$$

实际上，式27与论文中的式子不一样（我觉得论文里写错了），我们对我们的公式可以做个简单的验证，即令 $\mathbf{z}$ 为原图，即 $\mathbf{z} = \mathbf{x}$，我们做个简单的检测

$$\begin{aligned}
\hat{\mathbf{f}}(\mathbf{z}) &= \hat{\mathbf{z}}^* \odot \hat{\mathbf{w}} \\
&= \hat{\mathbf{x}}^* \odot \hat{\mathbf{w}} \\
&= \frac{\hat{\mathbf{x}}^* \odot \hat{\mathbf{x}}}{\hat{\mathbf{x}}^* \odot \hat{\mathbf{x}} + \lambda} \odot \hat{\mathbf{y}}
\end{aligned} \tag{31}$$

令 $\lambda = 0$ 上式化为

$$\hat{\mathbf{f}}(\mathbf{z}) = \hat{\mathbf{y}} \tag{32}$$

即

$$\mathbf{f}(\mathbf{z}) = \mathbf{y} \tag{33}$$

这个结果是合理的。

1.4 核方法

这里我们利用核方法，利用映射 $\phi(\cdot)$ 将 $\mathbf{x}$ 映射到非线性空间中。

1.4.1 岭回归

我们的优化对象仍然为

$$\min \sum_{i=1}^{m}(f(\mathbf{x}_i) - y_i)^2 + \lambda\|\mathbf{w}\|^2 \tag{34}$$

根据表示定理（见西瓜书 137 页），这个问题的解，总可写为

$$f^*(\mathbf{x}) = \sum_{i=1}^{m}\alpha_i\kappa(\mathbf{x}, \mathbf{x}_i) \tag{35}$$

这里的 $*$ 不表示共轭，表示最优解。所以线性回归方程可写为

$$f(\mathbf{z}) = \mathbf{w}^T \phi(\mathbf{z}) \tag{36}$$

其中 $\mathbf{w}$ 可写为

$$\mathbf{w} = \sum_{i=1}^{m} \alpha_i \phi(\mathbf{x}_i) \tag{37}$$

令核矩阵为 $\mathbf{K}$，且 $K_{ij} = \kappa(\mathbf{x}_i, \mathbf{x}_j)$。

1.4.2 矩阵表示

把 $\mathbf{w}$ 代入至线性回归方程得到

$$f(\mathbf{z}) = \sum_{i=1}^{m} \alpha_i \kappa(\mathbf{z}, \mathbf{x}_i) \tag{38}$$

优化对象为

$$\begin{aligned}
&\min \sum_{i=1}^{m} (\sum_{j=1}^{m} \alpha_j \kappa(\mathbf{x}_i, \mathbf{x}_j) - y_i)^2 + \lambda \|\mathbf{w}\|^2 \\
&\min \sum_{i=1}^{m} (\sum_{j=1}^{m} \alpha_j K_{ij} - y_i)^2 + \lambda \|\mathbf{w}\|^2 \\
&\min \sum_{i=1}^{m} (\mathbf{K}_i^T \boldsymbol{\alpha} - y_i)^2 + \lambda (\sum_{i=1}^{m} \alpha_i \phi(\mathbf{x}_i))^T (\sum_{i=1}^{m} \alpha_i \phi(\mathbf{x}_i)) \\
&\min (\mathbf{K}\boldsymbol{\alpha} - \mathbf{y})^T (\mathbf{K}\boldsymbol{\alpha} - \mathbf{y}) + \lambda \boldsymbol{\alpha}^T \mathbf{K} \boldsymbol{\alpha}
\end{aligned} \tag{39}$$

可以看出我们的优化对象从 $\mathbf{w}$ 变成了 $\boldsymbol{\alpha}$，我们称 $\boldsymbol{\alpha}$ 为对偶空间，而 $\mathbf{w}$ 称为主空间。

1.4.3 求解

令 $J(\boldsymbol{\alpha}) = (\mathbf{K}\boldsymbol{\alpha} - \mathbf{y})^T (\mathbf{K}\boldsymbol{\alpha} - \mathbf{y}) + \lambda \boldsymbol{\alpha}^T \mathbf{K} \boldsymbol{\alpha}$ 对 $\boldsymbol{\alpha}$ 求偏导得到。

$$\begin{aligned}
\frac{\partial J}{\partial \boldsymbol{\alpha}} &= 2\mathbf{K}^T \mathbf{K}\boldsymbol{\alpha} - 2\mathbf{K}\mathbf{y} + 2\lambda \mathbf{K}\boldsymbol{\alpha} \\
&= 2\mathbf{K}(\mathbf{K}\boldsymbol{\alpha} - \mathbf{y} + \lambda \boldsymbol{\alpha})
\end{aligned} \tag{40}$$

令偏导为零得

$$\boldsymbol{\alpha} = (\mathbf{K} + \lambda I)^{-1} \mathbf{y} \tag{41}$$

1.4.4 训练过程

若 $\mathbf{K}$ 为循环矩阵，那么，式41也可类似之前一样简化。可以证明，当 $\kappa(\mathbf{x}, \mathbf{x}') = \kappa(\mathbf{x}, \mathbf{x}')$ 时，$\mathbf{K}$ 为循环矩阵，即 K_{ij} 的值只与 $(i-j)$ 的值有关。证明如下。以下的 $\mathbf{P}$ 在式1中定义。

$$\begin{aligned}
K_{ij} &= \kappa(\mathbf{P}^i \mathbf{x}, \mathbf{P}^j \mathbf{x}) \\
&= \kappa(\mathbf{P}^{-i} \mathbf{P}^i \mathbf{x}, \mathbf{P}^{-i} \mathbf{P}^j \mathbf{x}) \\
&= \kappa(\mathbf{x}, \mathbf{P}^{j-i} \mathbf{x})
\end{aligned} \tag{42}$$

所以 K_{ij} 的值只与只与 $(i-j)$ 有关所以 $\mathbf{K}$ 为循环矩阵。实际上线性核、RBF 核等都满足该性质。

所以可以对 $\mathbf{K}$ 进行特征值分解。令 $\mathbf{k}^{\mathbf{xx}}$ 为矩阵 $\mathbf{K}$ 的第一行的转置，即 $k_i^{\mathbf{xx}} = \kappa(\mathbf{x}, \mathbf{x}_i)$。

$$\begin{aligned}
\boldsymbol{\alpha} &= (\mathbf{F}diag(\hat{\mathbf{k}}^{\mathbf{xx}})\mathbf{F}^H + \lambda\mathbf{F}\mathbf{F}^H)\mathbf{y} \\
&= \mathbf{F}diag(\frac{1}{\hat{\mathbf{k}}^{\mathbf{xx}} + \lambda})\mathbf{F}^H\mathbf{y} \\
\mathbf{F}^H\boldsymbol{\alpha} &= diag(\frac{1}{\hat{\mathbf{k}}^{\mathbf{xx}} + \lambda})\mathbf{F}^H\mathbf{y} \\
\hat{\boldsymbol{\alpha}} &= diag(\frac{1}{\hat{\mathbf{k}}^{\mathbf{xx}} + \lambda})\hat{\mathbf{y}}
\end{aligned} \tag{43}$$

所以我们得到。

$$\hat{\boldsymbol{\alpha}} = \frac{\hat{\mathbf{y}}}{\hat{\mathbf{k}}^{\mathbf{xx}} + \lambda} \tag{44}$$

1.4.5 检测过程

为方便，首先对 $\mathbf{k}^{\mathbf{xx}'}$（即，核相关向量）定义

$$k_i^{\mathbf{xx}'} = \kappa(\mathbf{x}', \mathbf{P}^{i-1}\mathbf{x}) = \kappa(\mathbf{x}', \mathbf{x}_i) \tag{45}$$

观察式45和式38可以将式38如下化简

$$f(\mathbf{z}) = \sum_{i=1}^{m} \alpha_i k_i^{\mathbf{xz}} = (\mathbf{k}^{\mathbf{xz}})^T\boldsymbol{\alpha} \tag{46}$$

所以

$$\begin{aligned}
\mathbf{f}(\mathbf{z}) &= C(\mathbf{k}^{\mathbf{xz}})\boldsymbol{\alpha} \\
&= \mathbf{F}diag(\hat{\mathbf{k}}^{\mathbf{xz}})\mathbf{F}^H\boldsymbol{\alpha} \\
\mathbf{F}^H\mathbf{f} &= diag(\hat{\mathbf{k}}^{\mathbf{xz}})\mathbf{F}^H\boldsymbol{\alpha} \\
\mathbf{F}\mathbf{f} &= diag(\hat{\mathbf{k}}^{*\mathbf{xz}})\mathbf{F}\boldsymbol{\alpha} \\
\hat{\mathbf{f}} &= diag(\hat{\mathbf{k}}^{*\mathbf{xz}})\hat{\boldsymbol{\alpha}}
\end{aligned} \tag{47}$$

最后可得

$$\hat{\mathbf{f}} = \hat{\mathbf{k}}^{*\mathbf{xz}} \odot \hat{\boldsymbol{\alpha}} \tag{48}$$

1.5 核相关向量的计算

在训练过程，即式44和检测过程，即式48中，往往都是先将原始数据 $\mathbf{x}$ 转化为核相关向量 $\mathbf{k}^{\mathbf{xx}'}$ 再进行计算，这里给出内积核和径向基函数（RBF）核的简化计算方法。但并不是所有核都能有简化的方法。

1.5.1 内积核

内积核的形式为 $\kappa(\mathbf{x}, \mathbf{x}') = g(\mathbf{x}^T\mathbf{x}')$，结合式45可得

$$k_i^{\mathbf{xx}'} = \kappa(\mathbf{x}', \mathbf{P}^{i-1}\mathbf{x}) = g((\mathbf{P}^{i-1}\mathbf{x})^T\mathbf{x}') \tag{49}$$

所以

$$\mathbf{k}^{\mathbf{x}\mathbf{x}'} = g(C(\mathbf{x})\mathbf{x}') \tag{50}$$

对 $C(\mathbf{x})$ 施行特征值分解可得，

$$\begin{aligned}
\mathbf{k}^{\mathbf{x}\mathbf{x}'} &= g\left(\mathbf{F}diag(\hat{\mathbf{x}})\mathbf{F}^H\mathbf{x}'\right) \\
&= g\left(\mathbf{F}diag(\hat{\mathbf{x}})(\mathbf{F}^T\mathbf{x}')^*\right) \\
&= g\left(\mathbf{F}diag(\hat{\mathbf{x}})(\mathbf{F}\mathbf{x}')^*\right) \\
&= g\left(\frac{1}{\sqrt{N}}\mathbf{F}diag(\hat{\mathbf{x}})\hat{\mathbf{x}'}^*\right) \\
&= g\left(\frac{1}{\sqrt{N}}(\mathbf{F}^H diag(\hat{\mathbf{x}}^*)\hat{\mathbf{x}'})^*\right) \\
&= g\left(\frac{1}{\sqrt{N}}(\mathbf{F}^H \hat{\mathbf{x}}^* \odot \hat{\mathbf{x}'})^*\right) \\
&= g\left((\mathcal{F}^{-1}(\hat{\mathbf{x}}^* \odot \hat{\mathbf{x}'}))^*\right)
\end{aligned} \tag{51}$$

由于 $\mathbf{x}$ 和 $\mathbf{x}'$ 均为实序列，所以 $\hat{\mathbf{x}}$ 和 $\hat{\mathbf{x}}$ 必然都是共轭对称的序列，所以 $\hat{\mathbf{x}}^* \odot \hat{\mathbf{x}'}$ 也是共轭对称的序列，所以 $\hat{\mathbf{x}}^* \odot \hat{\mathbf{x}'}$ 的逆傅立叶变换必然是实序列，即 $(\mathcal{F}^{-1}(\hat{\mathbf{x}}^* \odot \hat{\mathbf{x}'}))^* = \mathcal{F}^{-1}(\hat{\mathbf{x}}^* \odot \hat{\mathbf{x}'})$，所以得到

$$\mathbf{k}^{\mathbf{x}\mathbf{x}'} = g\left(\mathcal{F}^{-1}(\hat{\mathbf{x}}^* \odot \hat{\mathbf{x}'})\right) \tag{52}$$

实际上，式53也是在推导中常用的。

$$C(\mathbf{x})\mathbf{x}' = \mathcal{F}^{-1}(\hat{\mathbf{x}}^* \odot \hat{\mathbf{x}'}) \tag{53}$$

特别的，我们可以写出多项式核 $\kappa(\mathbf{x}, \mathbf{x}') = (\mathbf{x}^T\mathbf{x}' + a)^b$ 的核向量为

$$\mathbf{k}^{\mathbf{x}\mathbf{x}'} = \left(\mathcal{F}^{-1}(\hat{\mathbf{x}}^* \odot \hat{\mathbf{x}'}) + a\right)^b \tag{54}$$

线性核 $\kappa(\mathbf{x}, \mathbf{x}') = \mathbf{x}^T\mathbf{x}'$ 的核向量为

$$\mathbf{k}^{\mathbf{x}\mathbf{x}'} = \mathcal{F}^{-1}(\hat{\mathbf{x}}^* \odot \hat{\mathbf{x}'}) \tag{55}$$

可以看出，核向量的计算也被简化为 $O(n \log n)$ 的时间复杂度了

1.5.2 径向基函数核

内积核的形式为 $\kappa(\mathbf{x}, \mathbf{x}') = h(\|\mathbf{x} - \mathbf{x}'\|^2)$，结合式45可得

$$k_i^{\mathbf{x}\mathbf{x}'} = \kappa(\mathbf{x}', \mathbf{P}^{i-1}\mathbf{x}) = h\left(\|\mathbf{x}' - \mathbf{P}^{i-1}\mathbf{x}\|^2\right) \tag{56}$$

将上式展开得到

$$k_i^{\mathbf{x}\mathbf{x}'} = h\left(\|\mathbf{x}'\|^2 + \|\mathbf{x}\|^2 - 2(\mathbf{P}^{i-1}\mathbf{x})^T\mathbf{x}'\right) \tag{57}$$

写成向量形式

$$\mathbf{k}^{\mathbf{x}\mathbf{x}'} = h\left(\|\mathbf{x}'\|^2 + \|\mathbf{x}\|^2 - 2C(\mathbf{x})\mathbf{x}'\right) \tag{58}$$

将式53代入上式得到

$$\mathbf{k}^{\mathbf{x}\mathbf{x}'} = h\left(\|\mathbf{x}'\|^2 + \|\mathbf{x}\|^2 - 2\mathcal{F}^{-1}(\hat{\mathbf{x}}^* \odot \hat{\mathbf{x}'})\right) \tag{59}$$

特殊的，考虑高斯核 $\kappa(\mathbf{x}, \mathbf{x}') = \exp(-\frac{1}{\sigma^2}\|\mathbf{x} - \mathbf{x}'\|^2$

$$\mathbf{k}^{\mathbf{x}\mathbf{x}'} = \exp\left(-\frac{1}{\sigma^2}\left(\|\mathbf{x}'\|^2 + \|\mathbf{x}\|^2 - 2\mathcal{F}^{-1}(\hat{\mathbf{x}}^* \odot \hat{\mathbf{x}'})\right)\right) \tag{60}$$

该式同样是一个 $O(n \log n)$ 时间复杂度的式子。

1.6 拓展至多通道问题

若原始图像是彩色图像，或者我们不用原始图像去做特征而是用提取的特征（如 HOG 特征），这样我们就会面临多通道问题，之前的所有推导都是建立在多通道问题上的，这样我们就需要对我们的结果进行修改使其能够适应多通道问题。

假设特征有 C 个通道，$\mathbf{x}_1, \cdots, \mathbf{x}_C$，为了能够将其能够适用式44和式48，需用核函数将其转为式45的核相关向量形式，在第1.5部分我们对内积核和 RBF 核做了讨论，其中只牵扯到对原始向量的求范数和内积的操作，这两种操作可以很容易拓展至多通道。

$$\|\mathbf{x}\|^2 = \sum_{i=1}^{C} \|\mathbf{x}_i\|^2 \tag{61}$$

$$\mathbf{x}^T\mathbf{x}' = \sum_{i=1}^{C} \mathbf{x}_i^T\mathbf{x}' \tag{62}$$

所以式55可写为

$$\mathbf{k}^{\mathbf{x}\mathbf{x}'} = \mathcal{F}^{-1}(\sum_{c=1}^{C} \hat{\mathbf{x}}_c^* \odot \hat{\mathbf{x}'}_c) \tag{63}$$

式60可写为

$$\mathbf{k}^{\mathbf{x}\mathbf{x}'} = \exp\left(-\frac{1}{\sigma^2}\left(\|\mathbf{x}'\|^2 + \|\mathbf{x}\|^2 - 2\mathcal{F}^{-1}(\sum_{c=1}^{C} \hat{\mathbf{x}}_c^* \odot \hat{\mathbf{x}'}_c)\right)\right) \tag{64}$$

1.7 对偶相关滤波器

我们使用线性核 $\kappa(\mathbf{x}, \mathbf{x}') = \mathbf{x}^T\mathbf{x}'$ 时，核相关向量的表示为式63，训练过程为式44，检测过程为式48，我们称这套 tracking 方法为对偶相关滤波器，因为它是线性的，并且是在对偶空间 $\boldsymbol{\alpha}$ 训练的。

1.8 总结

核相关滤波器 (KCF) 和对偶相关滤波器 (DCF) 与其他滤波器相比，最大的特点是速度快，同时也保证了准确性，从图2可以看出。

总之这篇论文行文非常顺畅，数学推导非常优美（但是有些地方似乎推导的不对? 可能是我错了?)，效果也非常好，可以说是 tracking 类论文的经典了！

	Algorithm	Feature	Mean precision (20 px)	Mean FPS
Proposed	KCF	HOG	**73.2%**	172
	DCF		**72.8%**	**292**
	KCF	Raw pixels	56.0%	154
	DCF		45.1%	278
Other algorithms	Struck [7]		65.6%	20
	TLD [4]		60.8%	28
	MOSSE [9]		43.1%	**615**
	MIL [5]		47.5%	38
	ORIA [14]		45.7%	9
	CT [3]		40.6%	64

Table 1: Summary of experimental results on the 50 videos dataset. The reported quantities are averaged over all videos. Reported speeds include feature computation (e.g. HOG).

图 2: 算法对比