花了四五天的时间终于把Moose和CSK的的文章看了一遍，并且把CSK的代码精读了一遍并且添加了详细的中文注释，改的地方极少，回过头来看核心代码其实比较简单（但是背后的原理其实理解的不是很透，这也是想要总结一下的原因），主要是这几天杂事比较多浪费了许多时间，只有每天下午效率还算高一点，添加了注释的代码已上传至[github](https://github.com/zhxing001/coder_of_paper_about_tracking)（刚开始用，代码较少，欢迎交流）。

本文的创新点主要再下面几个部分：（这是作者自己写的）

1. Dense sampling，作者提出了一种理论框架，可以学习一个一般的分类器来遍历所有子窗口。
2. 证明在论文所述情况下可以利用核矩阵具有循环结构。
3. 对下面三个问题具有快速，精确的闭式解：
4. KRLS with dense ampling.
5. Detection at all subwindows with generic Kernel classifiers.
6. 在所有子窗口计算内核，包括高斯核核多项式核（matlab代码给出了高斯核的计算）

作者铺垫了很多知识，这些知识可以对照这论文来看相关资料，我一开始也准备在这里写，发现光写公式就很麻烦，还是把主要代码试下过程的流程在这里总结出来比较好。

**1.首先是参数准备：(y,yf)**

除了一些乱七八糟的参数之外，这里要准备的主要是训练标签y，应该是一个理想的高斯型，方差作者有建议的选择方式，还有其频域yf。因为每一帧都会把目标的中心放在当前截取的sub\_window的中心，所以这个y和yf的参数可以循环使用，还有一个原因就是target\_size的大小是不变的，这就直接导致了csk这个算法的初始版本对于尺度变化是无法掌控的（天啦我竟然理解到这一点，如果在这个基础上做个多通道和尺度变换，后main的论文还没看，还不知道尺度变换的细节怎么做，不过做了这个就又是一篇CVPR了，可惜这个已经有人做过了）

**2.然后是训练：**

对于第一帧图像来说，从ground\_truth中读取目标的位置和大小信息（当然这是测试视频有gt信息，如果没有则需指定，对第二帧及以后的图像就直接用检测到的position了）做下面几个处理。

1. 获得子窗口x，这个并不是跟踪框的大小，而是对跟踪框进行padding之后的带下，一般1-1.5倍，论文里给的是1倍，也就是扩大成跟踪框的两倍，在这里面进行跟踪，寻找新目标的位置。
2. 对子窗口进行加窗处理，主要是为了处理边缘，减少其变换到频域时的噪声。因为fft认为所处理的信号是周期信号，所以边缘在计算fft是是连起来的（左右-上下），加窗可以使边缘平滑，减少变换到复频域的噪声（我大概是这样理解的）。
3. 计算k和alpha：

其中k是径向基函数，也就是核函数，这也是一个矩阵，和x,y大小都是相同的。

Alpha是组成w的组合系数（看原文Eq 1下的解）。

通过k和Alpha可以计算出相应图。

K和Alpha是由x和y得到的，其实k和a就代表了这个相关滤波的模型，而且已经是包含了X的，也就是为了计算响应最后的准备工作了。

**3.识别（detection）**

计算响应并，获得新的目标位置，更新Alpha，z（线性插值），这个都是线性插值更新。然后就是重复2，3步骤了，

写完还是觉得画一个流程图可能会比较好。

这个看的不是特别清楚，如果装了visio的话可以直接点进去看。