

# rcpr算法原理简介

我们的方法灵感源于最近在级联人脸标点上的研究【19, 4, 28, 29, 21】（文献全部相对于<http://note.youdao.com/share/?id=1123faf2e11bf83da06e3b3848a32567&type=note>），这些研究中，人脸形状通过提升的回归器逐渐更新。每一级上的回归器学习不仅取决于图像信息，也取决于从前一级回归器估计到的形状。这种方式的特征学习称之为形状索引特征。这种特征在人脸形状发生几何变化上呈现出更高的不变性。这对获得高的标点准确率和速度而言是很重要的。

我们使用了一种结合姿态索引特征和提升回归的姿态回归框架。这种框架在【4, 28, 29, 21】中都证明了其人脸标点的高效性。我们定义形状S为一个2L维的向量，L是标点总数。逐步回归的过程是：

$$S^t = S^{t-1} + \mathcal{R}^t(x, S^{t-1}), t = 1, \dots, T. \quad (2)$$

每一个  $\mathcal{R}^t$  是一个回归函数，它会在前一级形状的基础上增加一个形状增量。学习时使当前形状  $S^t$  与真实形状  $\hat{S}$  之间的误差最小。——在所有的样本上考虑这个误差。如下式所示：

$$\mathcal{R}^t = \arg \min_{\mathcal{R}} \sum_i \|\hat{S}_i - (S_i^{t-1} + \mathcal{R}(x_i, S_i^{t-1}))\|^2, \quad (3)$$

where index  $i$  iterates over all the training samples.

一个关键的创新点在于，在标点框架中，每一个回归器  $\mathcal{R}^t$  依赖于前一个形状  $S^{t-1}$ ，在训练学习的过程中，特征定义为与  $S^{t-1}$  相关，所以称之为姿态/形状索引特征【19, 4】，这种特征对面部形状变化时，呈现出很好的几何不变性。

回归函数  $\mathcal{R}^t$  是K个基于树的回归量的和

$$\mathcal{R}^t(x, S^{t-1}) = \sum_{k=1}^K \mathcal{R}_k^t(x, S^{t-1}). \quad (5)$$

每一个回归量  $\mathcal{R}_k^t$  是一个决策树（蕨）——它的每个叶子上存储了一个形状增量。图像窗口X落入哪一个叶子上，则决策树（回归量）即输出该叶子上的形状增量。（实现中  $T=100, K=50$ ）

怎么学习所有的决策树（蕨） $\mathcal{R}_k^t$  呢？？？

## 树的学习：

从根节点递归地构造树——构建过程中，每个节点存储了一个训练图块集，如果该节点深度达到最大深度（ $D_{max}=??$ ）或者节点存储的图块集数量很小（ $M_{min}=20$ ），则停止递归，令该节点为叶子节点，并且将叶子的形变增量存储起来。

否则，该节点被创建为非叶子，并从一个大的随机生成的二值测试池选择一个最优的二值测试。根据所选的二值测试，将到达该节点的训练图块集分为2个子集（输出0的为一个子集，输出1的为另外一个子集），分别成为该节点的2个儿子节点。

$$t_{a,p,q,r,s,\tau}(\mathcal{I}) = \begin{cases} 0, & \text{if } I^a(p,q) < I^a(r,s) + \tau \\ 1, & \text{otherwise.} \end{cases}$$

二值测试函数训练完成后即得到坐标点  $(p,q)$ ， $(r,s)$  和相应阈值  $\tau$  的信息，（ $a$ 表示通道数，此处忽略，实现时采用单通道）实际训练得到的形变增量存储于 `regModel-regs-regInfo-[fids, thrs]` 数据结构之中。  
`fids` 存储二值测试选用的坐标点对，`thrs` 存储相应的阈值。

全局的树输出的训练学习：

每个叶子上存储一个形状增量（多个标定点的增量），并且所有的形状增量使用方程（3）进行优化。注意到这是一个简单的全局线性回归问题。实际训练得到的形变增量存储于 `regModel-regs-regInfo-ysFern` 数据结构之中