**抓取程序调整需求**

一、对抓取目标进行分类和优先级

目前现状：

1.按时间先后抓取识别的所有目标，抓手客观上存在来不及抓的情况；

2.客户的需求，对清除异纤的优先级要高于黄毛；

3.目前异纤的设定抓取阈值为0.2，但根据图片分析，真正的异纤90%以上阈值大于0.3。而在0.2-0.3之间的异纤中，大部分识别的目标都是草杆或树枝（后续产品和算法进行优化以降低草杆树枝被识别的概率），少量是真正异纤。深黄毛目前没有实际阈值，统一为1.0。

建议方案：

1. 根据不同分类实施抓取优先级，并且在同分类中区分优先和疑似的2个等级，每个分类的优先和疑似的区分阈值可在配置文件设置；

2. 分类的优先级为：异纤>深黄毛>中黄毛（如需抓取）>浅黄毛（如需抓取）；

3. 抓取等级的顺序为：异纤优先等级>深黄毛优先等级>中黄毛优先等级（如需抓取）>浅黄毛优先等级（如需抓取）>异纤疑似等级>深黄毛疑似等级>中黄毛疑似等级（如需抓取）>浅黄毛疑似等级（如需抓取）。一个等级内已无目标点后再读取下一个等级点位。

4. 每个等级的最大容量可在配置文件进行设置，目前建议统一设置为5个；

5. 同分类同等级内目前仍以时间先后为序排序并抓取。

1. 数据库和统计报表

目前点位表

CREATE TABLE `t\_point\_record` (

`id` bigint(11) NOT NULL AUTO\_INCREMENT COMMENT '自增主键',

`point\_xmax` int(10) unsigned DEFAULT NULL COMMENT '识别目标的x坐标最大值',

`point\_ymax` int(10) unsigned DEFAULT NULL COMMENT '识别目标的y坐标最大值',

`point\_xmin` int(10) unsigned DEFAULT NULL COMMENT '识别目标的x坐标最小值',

`point\_ymin` int(10) unsigned DEFAULT NULL COMMENT '识别目标的y坐标最小值',

`point\_x` int(10) unsigned DEFAULT NULL COMMENT '识别目标的x坐标平均值（中心点）',

`point\_y` int(10) unsigned DEFAULT NULL COMMENT '识别目标的y坐标平均值（中心点）',

`state` tinyint(4) NOT NULL DEFAULT '1' COMMENT '状态[1:新增;2;超出边缘;3:重复;4:成功抓取;5:来不及抓取]',

`is\_del` tinyint(4) NOT NULL DEFAULT '1' COMMENT '是否删除[0:是;1:否]',

`create\_time` timestamp NOT NULL DEFAULT '2000-01-01 00:00:00' COMMENT '添加时间',

`update\_time` timestamp NOT NULL DEFAULT '2000-01-01 00:00:00' COMMENT '更新时间',

PRIMARY KEY (`id`)

FOREIGN KEY（照片表的主键）

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COMMENT='点位表';

根据上述需求，要增加3个字段，分类、阈值和等级

**报表页面呈现**

搜索和筛选：

在页面的搜索区域可以：选择照片时间的起始日期和时间，结束日期和时间；目标点位的状态（1:新增;2;超出边缘;3:重复;4:成功抓取;5:来不及抓取）进行搜索；也可输入照片ID直接搜索。筛选项增加可选择点位的分类和等级。

结果列表：

搜索出来的目标点位列表，呈现信息包括：识别目标的x和y坐标平均值（中心点）、照片ID（可点击打开照片，且照片上有识别出的目标点位的框）、照片的拍照开始时间和结束时间。增加显示点位的分类、等级和阈值。

1. 可能的问题

目前在仅抓取异纤的情况下，在现场看，抓手大部分时间是停止状态，但统计数据表明，抓手仍然放弃了超过50%的异纤。可以检查一下造成这个情况的原因，如果确有问题，是否在程序重构后可以避免这个问题。