

# QLCP v2024.6 操作手册

## 0. 概述

QLCP (Quick Light Curve Pipeline) 为一个地基望远镜时域观测数据处理管线，输入观测原始数据所在目录，最终输出目标的光变曲线以及相关数据。该软件早期版本已经发表在《天文研究与技术》期刊2023年1期。当前版本为重构后版本。

程序发布在：

- <https://github.com/DrJieZheng/qlcp>
- <https://gitee.com/drjiezhen/qlcp>
- PyPI: `qlcp`

## 1. 安装

```
1 | pip3 install qlcp
```

要求：Python版本至少为3.10。需要安装的其他软件包为：`numpy`，`scipy`，`matplotlib`，`astropy`，`PyAstronomy`，`qastutil`，`qmatch`，正常情况下，pip3（或pip）会识别到这些依赖项目并进行自动安装。

## 2. 运行

以下为最简调用方式：

```
1 | import qlcp
2 |
3 | qlcp.run(
4 |     "path/to/raw/data",
5 |     "path/to/output",
6 | )
```

具体参数见下文。

## 3. 参数

参数汇总

```

1 raw_dir:str,
2 red_dir:str,
3 steps:str="lbfiopwkcdg",
4 obj:str=None,
5 band:str=None,
6 use_bias:str=None,
7 use_flat:str|dict=None,
8 alt_bias:str=None,
9 alt_flat:str|dict=None,
10 alt_coord:dict|tuple[str]=None,
11 base_img:str=None,
12 se_cmd:str="source-extractor",
13 aper:float|list[float]=None,
14 starxy:list[list[float]]|dict=None,
15 ind_tgt:int|list[int]=None,
16 ind_ref:int|list[int]=None,
17 ind_chk:int|list[int]=None,
18 mode:workmode=workmode(workmode.MISS_SKIP+workmode.EXIST_APPEND),
19 ini_file:str|tuple[str]|list[str]=None,
20 **kwargs

```

## 原始数据路径 **raw\_dir** （必选）

通常为整晚的观测数据的保存位置。程序将自动从其中识别出BIAS、FLAT、科学目标等。

```
"raw/20240606_85/",
```

## 输出路径 **red\_dir** （必选）

如果不存在，程序将自动创建目录。本程序所有输出都在该路径内。

```
"red/20240606_85_red/",
```

注意：请合理安排数据保存位置，尽量让原始数据和处理结果分离，避免误操作。

前两个参数是必选参数，不需要加上参数名。后续参数建议加上参数名。

## 处理目标 **obj**

如果指定，只处理指定目标，可以为一个或多个。否则处理所有目标。

```
obj="UYUMa",
```

```
obj=["UYUMa", "ACAnd"],
```

## 处理波段 **band**

如果指定，只处理指定波段，否则处理所有波段。

```
band="BVR",
```

## 替代本底 `use_bias` `alt_bias`

如果指定 `use_bias`，使用该本底，否则使用当天数据生成的本底。如果当天本底缺失，使用 `alt_bias`。

```
use_bias="usethis/bias_85.fits/",
```

```
alt_bias="alt/bias_85.fits/",
```

优先级： `usebias` --> `today_bias` --> `altbias`。

## 替代平场 `use_flat` `alt_flat`

基本同bias，但平场必须和波段一一对应。

```
1 use_flat={
2     "B": "usethis/flat_B_85.fits/",
3     "V": "usethis/flat_V_85.fits/",
4     "R": "usethis/flat_R_85.fits/",
5 },
```

```
1 alt_flat={
2     "B": "alt/flat_B_85.fits/",
3     "V": "alt/flat_V_85.fits/",
4     "R": "alt/flat_R_85.fits/",
5 },
```

优先级： `useflat` --> `today flat` --> `altflat`。

## 坐标 `alt_coord`

如果指定，使用该坐标，否则从fits头中读取 `RA` 和 `DEC` 两个字段。如果只给一对，则给所有目标。不同目标用字典形式，键为目标名，值是坐标。坐标用元组/列表形式，第一个为赤经，格式为hh:mm:ss.s，第二个为赤纬，格式为±dd:mm:ss.s，秒的小数位不限。

可以多给，每个目标只根据名字选择自己的。

```
alt_coord= ("12:34:56", "+23:45:67.89"),
```

```
1 alt_coord={
2     "AAA": ("12:34:56", "+23:45:67.89"),
3     "BBB": ("23:56:34", "+56:12:34"),
4 },
```

## 基础图像 `base_img`

用于指定对齐时的图像，如果指定为整数，则从0开始计数，否则为文件名，如果非当日图像，必须是同一台望远镜的数据。默认为0，即当日该目标该波段第1幅图像。

```
base_img=10
```

```
base_img="base/AAA.fits"
```

关联性：后续的 `starxy` 参数是目标在本图像中坐标。

## Source Extractor命令 `se_cmd`

考虑到不同系统、不同版本Source Extractor的命令可能不同，这里提供命令名称，默认为 `source-extractor`。

- `se_cmd="sex"`, macOS和早期Linux
- `se_cmd="sextracfor"`, 中期Linux
- `se_cmd="source-extractor"`, 目前Linux

## 测光孔径 `aper`

本程序输出星等，除AUTO星等外，还根据指定的孔径输出不同星等。本参数可以是单个浮点数，或者浮点数组成的列表，最多可以支持9个孔径。如果不提供本参数，会默认以5个像素为孔径进行孔径测光。

```
aper=8.0,
```

```
aper=(3, 6.0, 9),
```

## 星坐标 `starxy`

指定一个列表，每个元素是坐标，顺序为x, y。

如果缺失，程序将自动从图像中寻找，由于各波段是独立处理，因此自动找到的目标星、比较星相互可能不同。

```
1 starxy=[
2     ( 927, 1018),
3     ( 855,  920),
4     (1107, 1349),
5     (1161, 1434),
6     (1220, 1289),
7     ( 688, 1050),
8     (1255,  861),
9 ],
```

本参数也可以指定字典，关键字为目标名，值是坐标。

```

1  starxy={
2      "UYUMa": [
3          ( 927, 1018),
4          ( 855,  920),
5          (1107, 1349),
6          (1161, 1434),
7          (1220, 1289),
8      ],
9      "ACAnd": [
10         (1027,  718),
11         (1220, 1289),
12         ( 688, 1050),
13         (1255,  861),
14     ],
15 },

```

## 目标星、比较星、检验星 `ind_tgt` `ind_ref` `ind_chk`

给出目标、参考、检验星在列表中的索引，默认目标星是0号，比较星和检验星是1~n-1。

注意：和之前程序不同，本程序中目标星可以是多个。

```

1  ind_tgt=0,
2  ind_ref=[1,3,4],
3  ind_chk=2,

```

下标也可以才用字典方式提供，与 `starxy` 参数相同。

```

1  ind_tgt={"UYUMa": 0, "ACAnd": 3},
2  ind_ref={"UYUMa": [1,3,4], "ACAnd": [2,4]},
3  ind_chk={"UYUMa": 2},

```

## 文件存在模式 `mode`

如果输出文件存在，或者输入文件缺失，程序将根据此参数决定如何处理。

文件存在，处理模式有：

- `workmode.EXIST_SKIP` 跳过已经处理过的文件，记录一个警告，但是不报错。
- `workmode.EXIST_OVER` 覆盖已存在的文件。
- `workmode.EXIST_APPEND` 追加模式。针对列表处理时，处理过的文件跳过，未处理的文件正常处理，和SKIP相同；对于生成聚集型文件，如总星表、光变曲线时，和OVER相同，覆盖。（默认模式）
- `workmode.EXIST_ERROR` 报错，触发异常。

文件缺失，处理模式有：

- `workmode.MISS_SKIP` 跳过缺失的文件，记录一个警告，但是不报错，对于聚集文件，直接跳过整个步骤。（默认模式）

- `workmode.MISS_ERROR` 报错，触发异常。

```
mode=workmode(workmode.EXIST_APPEND+workmode.MISS_SKIP),
```

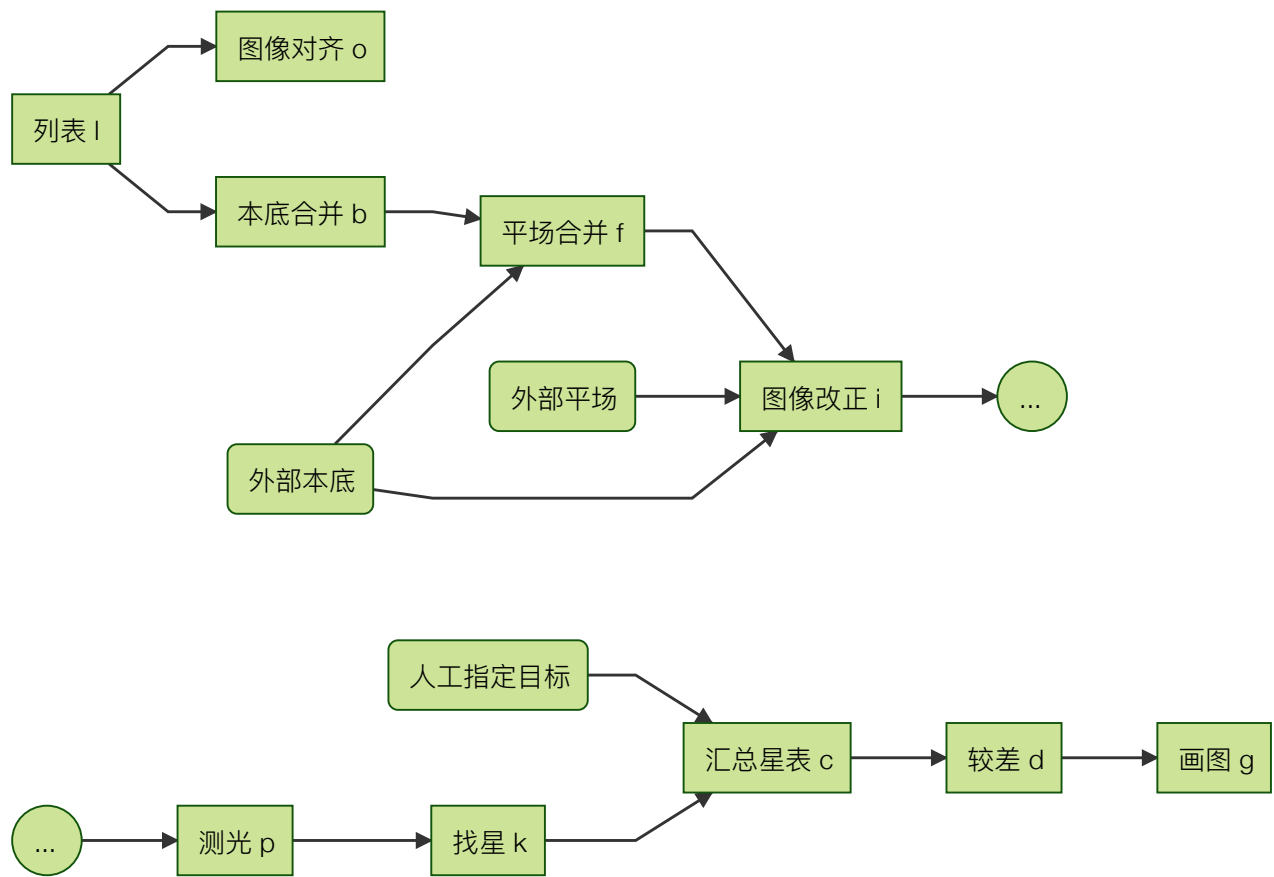
## 步骤 `steps`

本程序分为多个步骤，可以选择性执行，部分步骤顺序可以调整，部分步骤之间有依赖关系。

```
steps="lbf",
```

## 步骤依赖性

步骤依赖性表示做某个步骤之前应该先做哪一个。如果依赖的前置步骤更换参数重做了，后随步骤也必须相应重做。



## 列表生成 `1`

根据原始数据生成列表。配置文件中指定了文件名识别的模板，可以识别文件名中的目标类型（本底、平场、科学目标），目标名称，波段，序列号或其他观测编号。并生成列表。

列表保存在 `{red}/lst/{obj}_{band}.lst` 文件中。

相关配置参数：

- `patterns` 文件名模板，列表形式，每个元素是正则表达式，匹配文件名，默认为以下几种

- 减号分割模式 UYUMa-0003l.fit bias-0001.fits

```
(?P<obj>[^_]*)-(?P<sn>[0-9]{3,6})(?P<band>[a-zA-Z]{0,1}).fit(s{0,1})
```

- 下划线分割模式 flat\_R\_003.fit TCrB\_V\_001.fits

```
(?P<obj>[^_]*)_(?P<band>[a-zA-Z]{0,1})_(?P<sn>[0-9]{3,6}).fit(s{0,1})
```

- 自动平场模式 flat\_R\_1\_R\_001.fit

```
(?P<obj>flat)_(?P<band>[a-zA-Z])_(?P<sn>[0-9]{1,2})_[a-zA-Z]_([0-9]{3,6}).fit(s{0,1})
```

如果未指定执行本步骤，系统也会自动检索目录，确定要处理的目标和波段，但是不会更新列表文件。如果有特殊需求，可以手动指定列表。

## 本底合并 **b**

合并本底，保存为 `{red}/bias.fits`。

注意：本程序不处理暗场，特殊情况需要处理的，可以用暗场取代本底进行近似处理。如果确需高精度处理，那么可以自行完成 `bfi` 三步骤。

## 平场合并 **f**

合并平场，保存为 `{red}/flat_{band}.fits`。

相关配置参数：

- `flat_limit_low = 5000` 平场中值上下限，过高或者过低的会被丢弃
- `flat_limit_high = 50000` 如果当天实在没有高质量平场，不得不用，请修改

## 图像改正 **i**

图像改正，保存为 `{red}/{obj}_{band}/{rawname}.bf.fits`。

改正的同时，会对fits头进行修正和部分项目的计算，如观测时间 `JD`、`MJD`，根据目标坐标计算出 `HJD`、`BJD`，计算观测条件 `AZ`，`ALT`，`AIRMASS` 等。

实践中，我们发现许多图像边缘质量较差，因此本程序会进行边缘裁剪。

相关配置参数：

- `site_tz = 8` 观测站时区，默认值为兴隆站
- `site_lon = 117.57722` 观测站经度 (117.34.38)
- `site_lat = 40.395833` 观测站纬度 (+40.23.45)
- `site_ele = 960` 观测站海拔
- `border_cut = 0` 边缘裁剪像素

本步骤部分处理调用 `astropy`、`pyastronomy`、`qastutil` 等包。

## 偏移 **o**

计算每张图像相对基准图像的偏移，保存为 `{red}/offset_{obj}_{band}.pkl/txt`。

pkl文件内容为4个变量，分别是JD、X偏移、Y偏移、文件名。最好根据文件名去匹配，避免顺序不同导致混乱。

文本文件为方便阅读版本，不作为后续数据处理使用。

相关配置参数：

- `offset_max_dis = 250` 最大距离，单位为像素，如果图像偏移超过这个值，并且确认图像可用，请加大。

本步骤调用 `qmatch` 包，算法见 `2024NewA..11002224Z`。

## 找星和测光 **p**

调用了 `Source Extractor`，对图像找星和测光，SE结果经过清洗，剔除边缘目标后，保存为 `{red}/{obj}_{band}/{rawname}.cat.fits`。包括位置、不同孔径的流量、星等。

如果希望自行配置 `default.sex` 文件，可以在当前路径下另存一份并进行配置，优先使用当前路径下的。输出字段文件 `default.param` 固定使用系统内置版本。

相关配置参数：

- `draw_phot = False` 是否为每幅图像都输出png图像并且标上找到的亮星
- `draw_phot_err = 0.05` 误差比该值小的星才会显示在星图上

## 天体位置测量定标 **w**

对每一幅图像进行定标，尚未实现。

## 目标选择 **k**

如果明确指定要执行本步骤，或者未指定目标时，会自动选择目标。

选择目标的原则是，从图像中选出质量较好的星（误差小于阈值），匹配使用图像中的优质星，并对流量进行粗略对齐。分析每颗星在所有数据中的变化情况，最稳定的星作为候选比较星的，变化较大的星则是候选目标星。要求候选星在所有图像中的缺失率（也就是匹配失败）足够低，候选比较星标准差小且最亮最暗相差较小，如果候选比较星较多，则只选择标准差最低的若干颗。候选目标星要求标准差足够大。对于非掩食阶段的目标星，本算法基本上找不出来，只能靠人工指定。

相关配置参数（注意有些是上限，有些是下限）：

- `pick_err_max = 0.02` 候选星在单幅图像中的AUTO测光误差阈值
- `pick_bad_max = 0.2` 候选星的缺失率上限。（ $0.2 = 20\%$ ）
- `pick_var_std = 0.05` 判定为变星的标准差下限
- `pick_var_rad = 0.5` 变星在图像中的位置上限，不能在图像边缘
- `pick_ref_n = 20` 最多选择的比较星个数
- `pick_ref_std = 0.05` 候选比较星的整晚标准差上限
- `pick_ref_dif = 0.10` 候选比较星的整晚最亮最暗差异上限

## 星表 **c**

根据 `starxy` 参数，生成星表，从每张图中找出对应位置的星的星等和流量，汇总为总仪器星表：

`{red}/cata_{obj}_{band}.pk1`，内容为总星表、输入的星位置、孔径列表。同时还生成fits格式星表：`{red}/cata_{obj}_{band}.fits`。



注意：当执行了 `k` 选星步骤时，`starxy` 参数自动无效，使用自动找到的星。

相关配置参数：

- `match_max_dis = 10` 匹配目标的最大距离，单位为像素

输出的 `xxx.pkl` 文件，可以用以下方式读取：

```
a, b, c = pickle.load(open("xxx.pkl", "rb"))
```

## 星表较差定标 `d`

利用指定的比较星，对目标星、进行定标。输出为定标后星表：`{red}/cali_{obj}_{band}.pkl`，内容为总星表（来自上一步），定标后星表（内容和总星表不重复，行数一致，需要联合使用），孔径列表，输入的星位置，目标星下标列表，比较星下标列表，检验星下标列表。

注意：当执行了 `k` 选星步骤时，`ind_xxx` 参数自动无效，使用自动找到的星。

## 绘图 `g`

根据较差星表画图，输出为：`{red}/lc_{obj}_{band}.png/pdf`。

## 配置文件（必选）

如果为 `None` 则使用默认配置。指定方式可以为单个字符串，或者多个字符串组成的列表。

使用配置文件，相对在函数参数中加配置而言，具有持久性，方便多次使用以及追溯处理过程。

```
ini_file=["xl85.ini", "mac.ini"],
```

## 其他参数

其他参数一律作为配置文件。

可以通过多种渠道提供配置，优先级为：函数参数 - 配置文件 - 默认值。

## 配置参数汇总

---

**【】** 内表示这一部分参数起作用的步骤代码。

```

1  # 日志【全局】
2  self.file_log = logging.DEBUG      # 文件日志等级
3  self.scr_log = logging.INFO       # 屏幕显示日志等级
4  # 观测站【i】
5  self.site_lon = 117.57722         # 观测站经度
6  self.site_lat = 40.395833        # 观测站纬度
7  self.site_ele = 960              # 观测站海拔
8  self.site_tz = 8                 # 观测站时区
9  # 平场采纳【f】
10 self.flat_limit_low = 5000        # 平场图像采纳下限
11 self.flat_limit_high = 50000     # 平场图像采纳上限
12 # 图像裁剪【i】
13 self.border_cut = 0              # 裁边像素
14 # 绘制测光结果【p】
15 self.draw_phot = False           # 是否画图
16 self.draw_phot_err = 0.05        # 标上的星的误差上限
17 # 图像对齐【o】
18 self.offset_max_dis = 250        # 对齐最远距离
19 # 目标匹配【kc】
20 self.match_max_dis = 10.0        # 目标匹配最大误差
21 # 选星【k】
22 self.pick_err_max = 0.02         # 优质星误差上限
23 self.pick_bad_max = 0.2          # 整晚的数据缺失率上限
24 self.pick_var_std = 0.05         # 变星标准差下限
25 self.pick_var_rad = 0.5          # 变星在图像中的区域
26 self.pick_ref_n = 20             # 比较星最多个数
27 self.pick_ref_std = 0.05         # 比较星整晚标准差上限
28 self.pick_ref_dif = 0.10         # 比较星整晚最亮最暗差上限

```

## 输出路径结构

- {red}/
  - log/ 日志文件夹
    - {step}.log 每个步骤的日志文件
  - lst/ 列表文件夹
    - {obj}\_{band}.lst 列表文件
  - bias.fits 本底
  - flat\_{band}.fits 平场
  - {obj}\_{band}/ 目标文件夹
    - {rawname}.bf.fits 改正后的图像
    - {rawname}.cat.fits 找星结果（多种格式）
  - off\_{obj}\_{band}.pk1 偏移
  - cata\_{obj}\_{band}.pk1 总星表
  - cali\_{obj}\_{band}.pk1 定标后星表

- `lc_{obj}_{band}.png/pdf` 光变曲线
- `cali_{obj}_AP{aper}/` 较差结果保存
  - `{obj}_{band}_chk{k}_{refs}.txt` 检验星改正后结果
  - `{obj}_{band}_vc{k}_{refs}.txt` 目标星改正后结果

## 附注

---

参数举例最后都有逗号，自行酌情保留或删除，建议哪怕最后一个参数也保留逗号。

除明确说明为整数的参数或者配置文件，否则均可以输入浮点数，使用浮点数时，输入整数也是可以的。必须使用整数的，通常是下标，或者图像切片参数。

本版本输出文件以pickle文件为主，fits文件为辅。txt文件仅用于人工阅读检查，不作为输入数据。