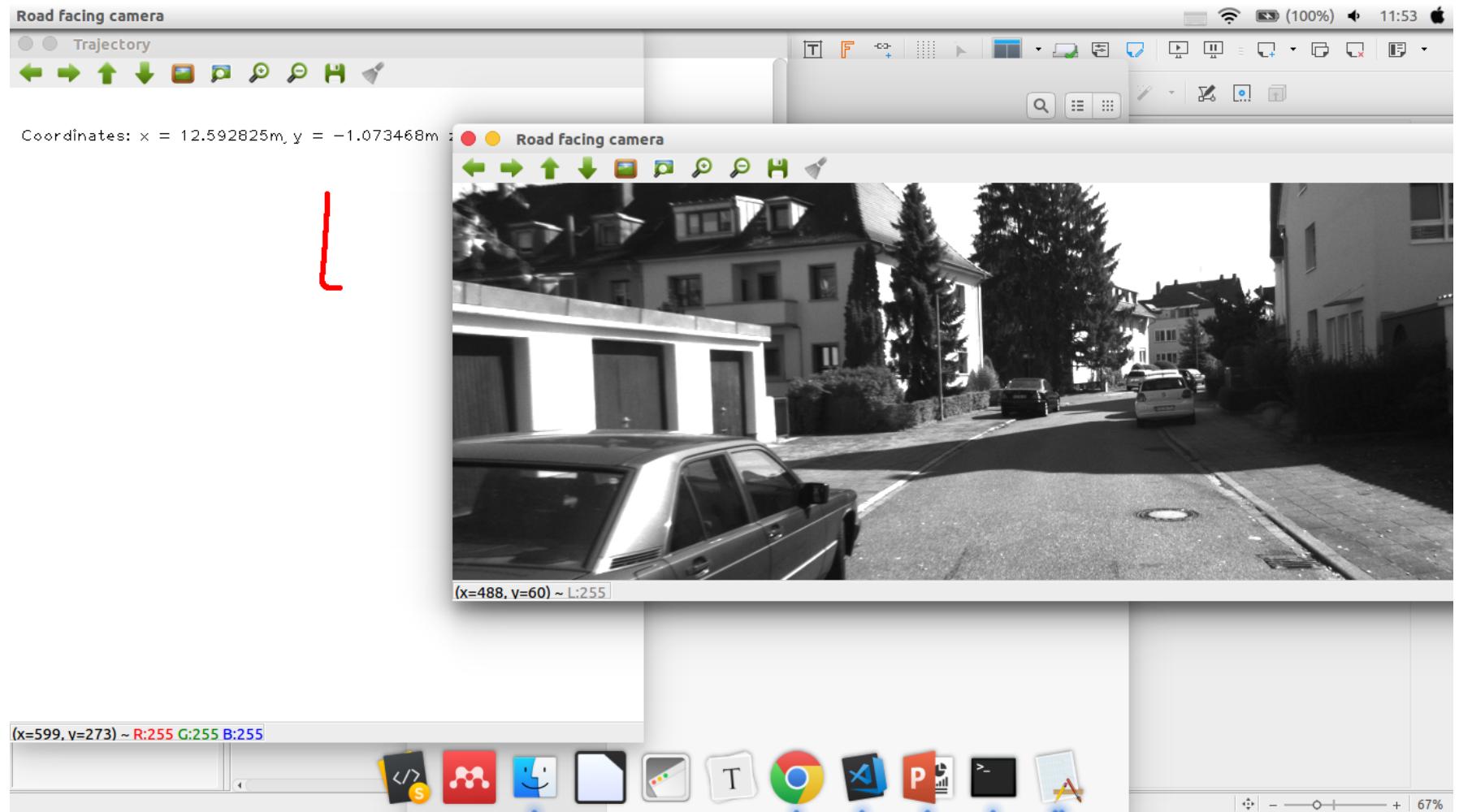


MonoVo

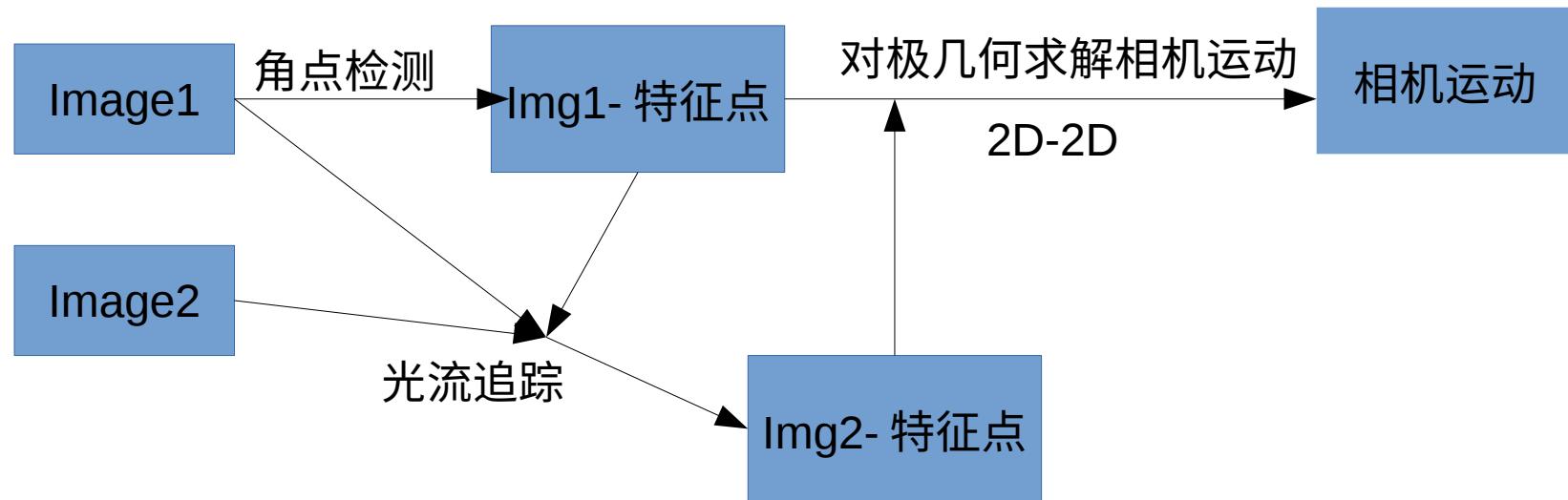
李建峰 冯亚炫

工作比例：李 6 : 冯 4

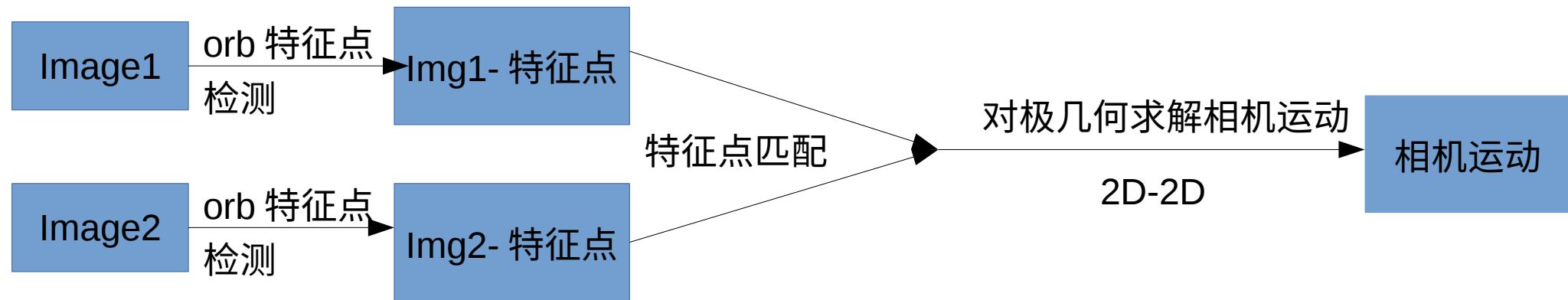
效果展示



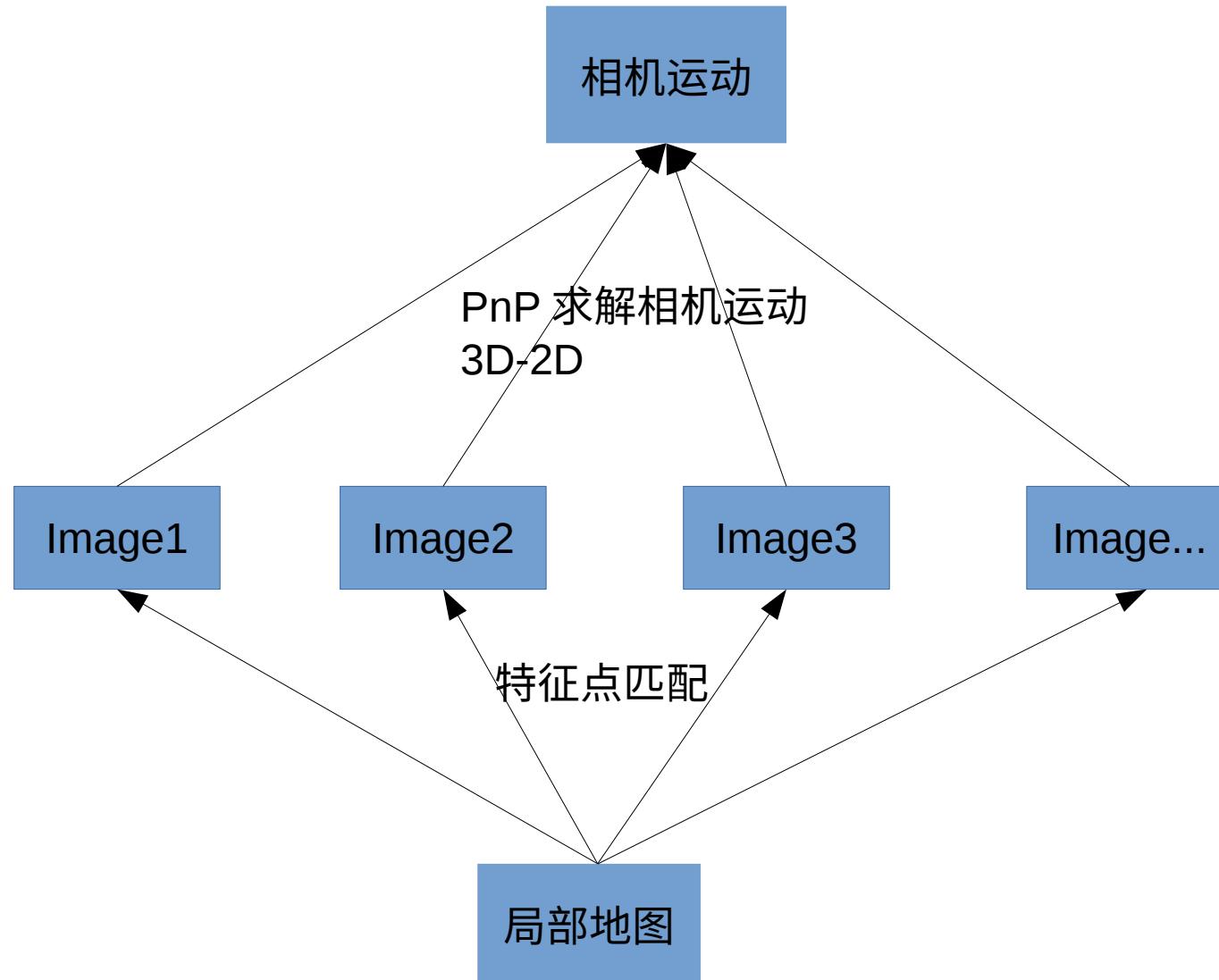
基于光流追踪的视觉里程计



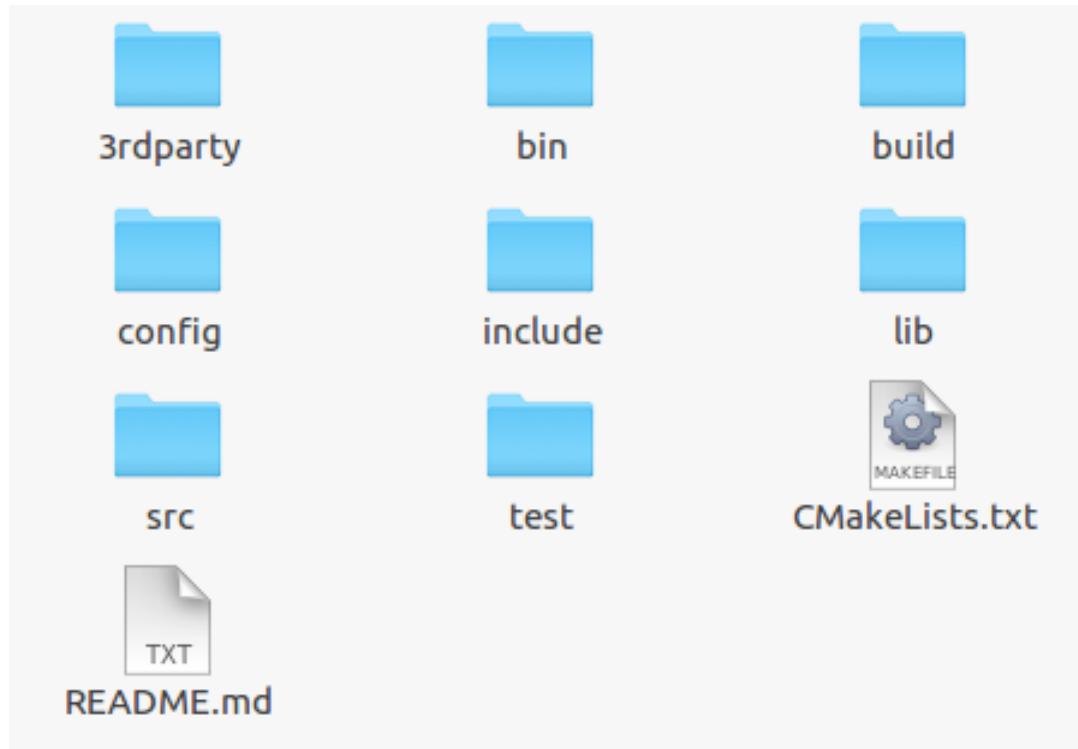
基于特征点匹配的视觉里程计



基于匹配附加局部地图的视觉里程计



代码实现



```
▲ include
C camera.h
C common_include.h
C config.h
C frame.h
C localmap.h
C map.h
C mappoint.h
C simplevo.h
C vo_track.h
C vo.h
```

代码实现—类的组织与功能

```
namespace slam
{
class Camera
{
private:
    /* data */
public:
    typedef std::shared_ptr<Camera> Ptr;
    float fx_, fy_, cx_, cy_, s_, depth_scale_;

    Camera();
    Camera(float fx, float fy, float cx, float cy, float s = 0, float depth_scale = 1):
        fx_(fx), fy_(fy), cx_(cx), cy_(cy), s_(s), depth_scale_(depth_scale)
    {}
    ~Camera();

    cv::Point2f pixel2cam_cv( const cv::Point2d& p, const cv::Mat& K );

    Vector3d world2cam(const SE3& T_c_w, const Vector3d& point_world);
    Vector2d cam2pixel(const Vector3d& point_camera);
    Vector3d pixel2cam(const Vector2d& point_image, double depth = 1);
    Vector3d cam2world(const SE3& T_c_w, const Vector3d& point_camera);

    Vector2d world2pixel(const SE3& T_c_w, const Vector3d& point_world);
    Vector3d pixel2world(const SE3& T_c_w, const Vector2d& point_image, double depth = 1 );
};

}
```

```
namespace slam
{
    class MapPoint;
    class Frame
    {
        private:
            /* data */
        public:
            typedef std::shared_ptr<Frame> Ptr;
            long id_;
            cv::Mat rgb_, depth_;
            cv::Mat R_, t_;
            Camera::Ptr cam_;
            SE3 T_c_w_;
            cv::Mat Tcw;
            double time_stamp_;

        public:
            Frame();
            Frame(long id, double time_stamp=0, SE3 T_c_w=SE3(),
                  Camera::Ptr cam=nullptr, Mat color=Mat(), Mat depth=Mat());
            ~Frame();
            static Frame::Ptr createFrame();
            double findDepth(const cv::KeyPoint& kp);
            Vector3d getCameraCenter() const;
            bool isInFrame(const Vector3d& pt_world);
            void se3ToT34();
    };
}
```

```
namespace slam
{
class Mappoint
{
private:
    /* data */
public:
    typedef std::shared_ptr<Mappoint> Ptr;
    unsigned long id_;
    long _frame_id;
    Vector3d point_pos_;
    Vector3d view_direction_;
    int observed_times_;
    int correct_times_;
    Mat descriptor_;
public:
    Mappoint();
    Mappoint(long id, Vector3d point_pos, Vector3d view_direction);
    Mappoint(long id, long frame_id, Vector3d point_pos, Vector3d view_direction);
    inline cv::Point3f getPositionCV() const {
        return cv::Point3f( point_pos_(0,0), point_pos_(1,0), point_pos_(2,0) );
    }
    ~Mappoint();

    static Mappoint::Ptr createPoint();
    static Mappoint::Ptr createPoint( Vector3d point_pos );
    static Mappoint::Ptr createPoint( Vector3d point_pos, long frame_id );
};

} // myslam
```

```
namespace slam
{
    class Localmap
    {
        public:
            int _max_key_frames;
            std::vector<Frame::Ptr> key_frames_;
            std::vector<Mappoint::Ptr> map_points_;
        public:
            typedef std::shared_ptr<Localmap> Ptr;
            Localmap() {}
            ~Localmap() {}
            Localmap(int max_key_frames)
                : _max_key_frames(max_key_frames)
            {
            }
            void addKeyFrame(Frame::Ptr frame);
            void addMapPoint(Mappoint::Ptr point);
    };
} // namespace slam
```

```
class Trackvo
{
public:
    typedef std::shared_ptr<Trackvo> Ptr;
    typedef cv::Size2i Size;
    enum VoState {
        INITIALIZING=-1,
        LOST=0,
        NORMAL=1
    };
    VoState vo_state_;
    Frame::Ptr ref_frame_;
    Frame::Ptr cur_frame_;

    std::vector<cv::Point2f> key_point_curr_;
    std::vector<cv::Point2f> key_point_ref_;

    cv::Mat relative_R_;
    cv::Mat relative_t_;

    double absolute_scale_;

public:
    Trackvo(/* args */);
    ~Trackvo();

    bool addFrame(Frame::Ptr frame);

public:
    void epipolarSolve();
    void getAbsoluteScale(long frame_id);

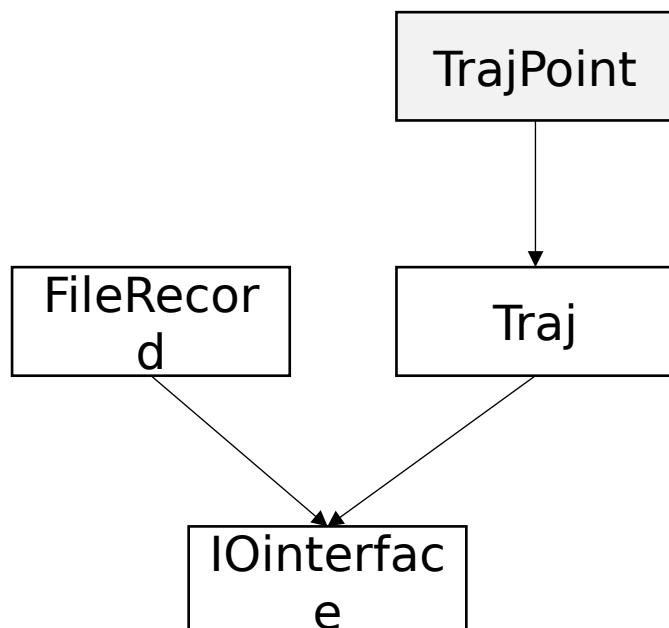
    void featureDetection();
    void featureTracking();
};
```

```
public:  
    Vo(/* args */);  
    ~Vo();  
  
    bool addFrame(Frame::Ptr frame);  
  
public:  
    void extractKeyPoints();  
    void computeDescriptors();  
    void featureMatch();  
    void featureMatchFromMap();  
    void poseEstimatePnP();  
    void setRefPoint3d();  
    void addKeyFrame();  
    void updateMap();  
    void epipolarSolve();  
  
    bool checkEstimatedPose();  
    bool checkKeyFrame();  
  
    int hanmingDistance(Mat str1, Mat str2);
```

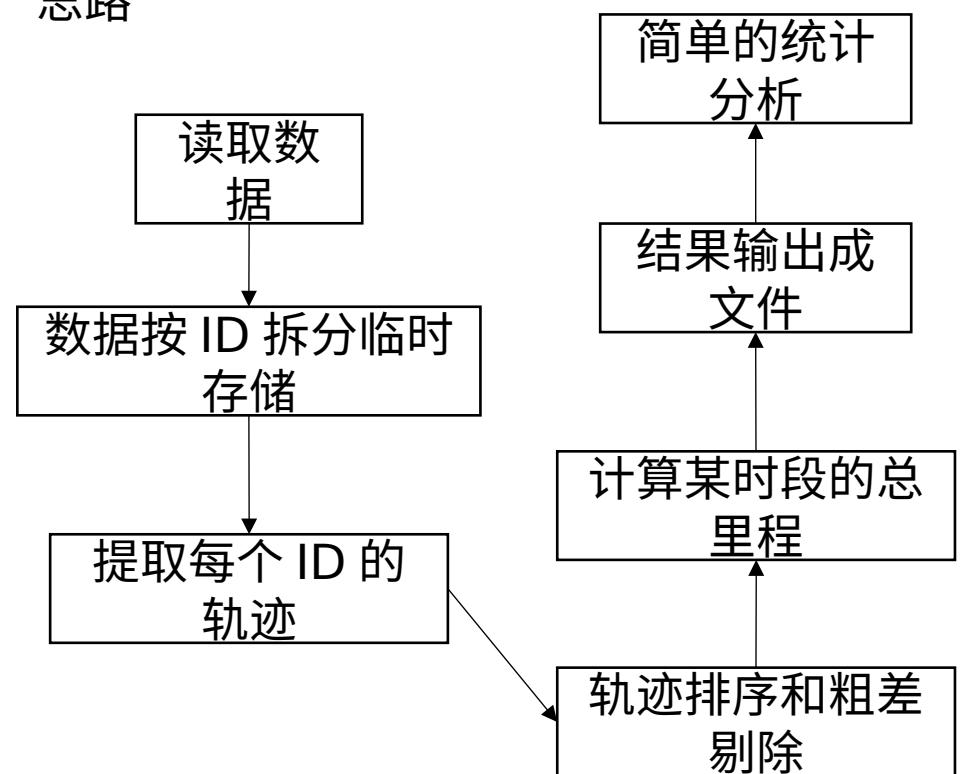
课程作业

- 项目名称：
TrajectoryExtraction
- 环境：ubuntu 14.04
cmake 工程
- 数据：
20161001.txt

项目结构



思路



- `FileRecord` 类：存储数据的每一条记录；
- `TrajPoint` 结构体：存储每个位置点的时空属性；
- `Traj` 类：存储每辆车的轨迹，并对轨迹进行排序、去粗差、计算累计距离等操作；
- `IOinterface` 类：用于文件的输入输出。

FileRecord 类

成员变量

```
bool           isGood = false;  
time_t         time;  
string         company;  
int            id;  
double         longitude;  
double         latitude;  
double         speed;  
double         azimuth;  
int            status;  
int            flag;  
string         reserved_field;  
time_t         unloading_time;
```

构造函数

```
FileRecord();  
FileRecord(string _str);
```

Traj 类

TrajPoint 结构体

```
struct TrajPoint{
    time_t
time;
    double
longitude;
    double
latitude;
    double
speed;
    double
azimuth;
    int
status;
};int
string
list<TrajPoint>
```

构造函数

```
Traj();
Traj(vector<Trajectory::FileRecord>
frs);
```

方法

```
void BL2xy(double lon, double lat, double &x,
double &y);
double calDist(TrajPoint tp1, TrajPoint tp2);
double calVel(TrajPoint tp1, TrajPoint tp2);
int eraseError();
void sortTrajPoints();
double calTotalDist(time_t tstart, time_t
tend);
```

IOinterface 类

方法

```
static void readFile(string path, vector<Trajectory::FileRecord> &records);  
  
static int splitFilewithID(string path, bool IDflag[], int &recordnum);  
  
static void deleteFilewithID(bool IDflag[]);  
  
static void getFileNames(string path, vector<string>& filenames);  
  
static time_t inputTime(int year,int month,int day,int hour,int minute,int second);  
  
static void outTrajLenth(string dataPath, string outPath, time_t tstart, time_t tend);  
  
static void outTrajLenthUsingThread(string dataPath, string outPath, time_t tstart, time_t tend);
```

IOinterface 类

多线程

```
static void outTrajLengthUsingThread(string dataPath, string outputPath, time_t tstart, time_t tend);
void *mythread(void *data)
{
    ioPath *iopath=(ioPath*)data;
    vector<Trajectory::FileRecord> frs;
    Trajectory::IOinterface::readFile(iopath->inPath,frs);
    Trajectory::Traj traj = Trajectory::Traj(frs);
    traj.sortTrajPoints();
    int errornum=traj.eraseError();
    double trajdist = traj.calTotalDist(iopath->t_start,iopath->t_end);
```

```
pthread_mutex_lock(&mutex);
ofstream outfile;
outfile.open(iopath->outPath, ios::out | ios::app);
if(outfile.is_open()){
    stringstream ss1,ss2;
    ss1<<traj.id ;
    ss2<<trajdist;
    string str =ss1.str() + "," + ss2.str();
    outfile <<str <<endl;
}
outfile.close();
pthread_mutex_unlock(&mutex);
```

思路：

用 5 个线程读约 5800 个文件，
处理后写入同一个输出文件。

互斥锁

过程与结果

数据读取存在乱码：

```
Starting /home/felix/桌面/build-TrajectoryExtraction-Desktop_Qt_5_7_0_GCC_64bit-Release/TrajectoryExtraction...
record number = 17936963
good number = 10855832
time cost = 169.857 seconds.
/home/felix/桌面/build-TrajectoryExtraction-Desktop_Qt_5_7_0_GCC_64bit-Release/TrajectoryExtraction exited with code 0
```

数据异常：

ction - Qt Creator

(控件(W) 帮助(H)

46 tempPath = tempPath + "/TempFile";
47 mkpath(tempPath,0755);
48 ifstream infile;
49 infile.open(path.c_str());
50 string s;
51 int step=0;
52 while(getline(infile,s))
53 {
54 Trajectory::FileRecord tempfr = Tra
55 if(tempfr.isGood){
56 recordnum++;
57 if(flag[tempfr.id] == true;
58
59 stringstream ss;
60 ss << setfill('0') << setw(5) << tem
61 string outpath = tempPath + "/" + s
62 ofstream outfile;
63 outfile.open(outpath,ios::out | io
64 if(outfile.is_open()){
65 outfile << s << endl;
66 }
67 outfile.close();
68 }
69 cout <<

| 名称 | 值 |
|-----------|---|
| outpath | "/home/felix/桌面/build-TrajectoryExtraction-Desktop_Qt_5_7_0_GCC_64bit-Debug(TempFile/12225.txt" |
| path | "home/felix/桌面/TrajectoryExtraction/data/20161001.txt" |
| recordnum | 13973087 |
| s | "20000119,230640,TR,1107826694,-50.665278,31.000004,121,65,0,196,1,2016-10-01 07:02:19\r" |
| ss | @0x7fffffe5580 |
| step | 13973086 |
| tempPath | "home/felix/桌面/build-TrajectoryExtraction-Desktop_Qt_5_7_0_GCC_64bit-Debug(TempFile" |
| tempfr | @0x7fffffe5520 |
| azimuth | 65 |
| company | "TR" |
| flag | 196 |
| id | 1107826694 |
| isGood | true |
| latitude | 31.000004000000001 |
| longitude | -50.665278000000001 |

Traj::calVel(tp,*it) <no such value>
*it <no such value>

3.8G 数据读取用时

```
应用程序输出 | < > ▶ ━ + -  
TrajectoryExtraction X TrajectoryExtraction X  
Starting /home/felix/桌面/build-TrajectoryExtraction-Desktop_Qt_5_7_0_GCC_64bit-Release/TrajectoryExtraction...  
time cost = 1850.77 seconds.  
/home/felix/桌面/build-TrajectoryExtraction-Desktop_Qt_5_7_0_GCC_64bit-Release/TrajectoryExtraction exited with code 0
```

3.8G 数据处理输出用时

```
应用程序输出 | < > ▶ ■ + -  
TrajectoryExtraction X TrajectoryExtraction X  
Starting /home/felixx/桌面/build-TrajectoryExtraction-Desktop_Qt_5_7_0_GCC_64bit-Release/TrajectoryExtraction...  
time cost = 446.119 seconds.  
/home/felixx/桌面/build-TrajectoryExtraction-Desktop_Qt_5_7_0_GCC_64bit-Release/TrajectoryExtraction exited with code 0
```

过程与结果

按 ID 拆分的临时文件中的两个个：

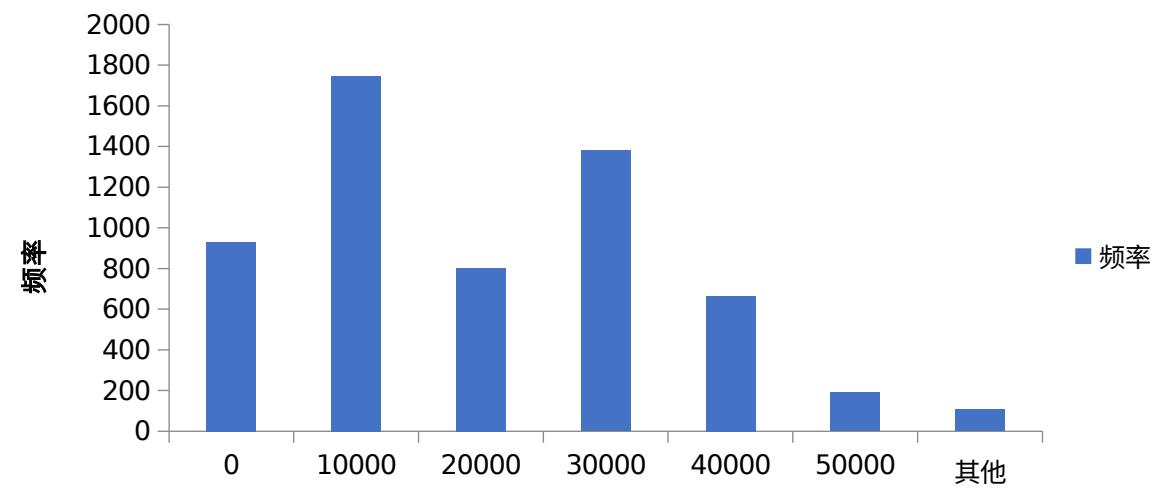
```
10068.txt - 记事本
文件(F) 编辑(E) 格式(O) 查看(V) 帮助(H)
20160901,235950,TR,10066,121,25460,31,250624,27,143,0,0,1,2016-10-01 00:00:44;
20161001,000000,TR,10066,121,25719,31,250267,1,134,0,0,1,2016-10-01 00:00:54;
20161001,000010,TR,10066,121,25679,31,250233,0,134,0,0,1,2016-10-01 00:01:05;
20161001,000020,TR,10066,121,25944,31,250022,27,142,0,0,1,2016-10-01 00:01:14;
20161001,000030,TR,10066,121,426435,31,249517,37,142,0,0,1,2016-10-01 00:01:26;
20161001,000040,TR,10066,121,427055,31,24666,42,147,0,0,1,2016-10-01 00:01:34;
20161001,000050,TR,10066,121,427933,31,247706,38,156,0,0,1,2016-10-01 00:01:45;
20161001,000010,TR,10066,121,427931,31,246946,39,164,0,0,1,2016-10-01 00:01:54;
20161001,0000110,TR,10066,121,42807,31,245912,36,168,0,0,1,2016-10-01 00:02:05;
20161001,0000120,TR,10066,121,42859,31,244921,38,157,0,0,1,2016-10-01 00:02:21;
20161001,0000130,TR,10066,121,429241,31,244320,38,127,0,0,1,2016-10-01 00:02:24;
20161001,0000140,TR,10066,121,430428,31,243850,42,101,0,0,1,2016-10-01 00:02:34;
20161001,0000150,TR,10066,121,431259,31,243428,39,121,0,0,1,2016-10-01 00:02:45;
20161001,0000200,TR,10066,121,431938,31,243023,14,118,0,0,1,2016-10-01 00:02:54;
20161001,0000210,TR,10066,121,432034,31,242933,0,135,0,0,1,2016-10-01 00:03:04;
20161001,0000220,TR,10066,121,432042,31,242909,9,135,0,0,1,2016-10-01 00:03:15;
20161001,0000230,TR,10066,121,432194,31,242609,1,112,0,0,1,2016-10-01 00:03:24;
20161001,0000240,TR,10066,121,43226,31,242057,0,112,0,0,1,2016-10-01 00:03:36;
20161001,0000250,TR,10066,121,43265,31,242924,0,112,0,0,1,2016-10-01 00:03:44;
20161001,0000300,TR,10066,121,43286,31,242958,0,112,0,0,1,2016-10-01 00:03:55;
20161001,0000310,TR,10066,121,432310,31,242996,0,112,0,0,1,2016-10-01 00:04:05;
20161001,0000320,TR,10066,121,432672,31,242910,28,121,0,0,1,2016-10-01 00:04:15;
20161001,0000330,TR,10066,121,432924,31,242533,14,198,0,0,1,2016-10-01 00:04:25;
```

```
10155.txt - 记事本
文件(F) 编辑(E) 格式(O) 查看(V) 帮助(H)
201601,061859,TR,10155,121,476763,31,303242,12,190,0,0,1,2016-10-01 06:20:35;
20161001,072032,TR,10155,121,479543,31,288679,87,157,0,1,1,2016-10-01 07:22:29;
20161001,072143,TR,10155,121,478714,31,288654,69,201,0,1,1,2016-10-01 07:22:37;
20161001,072153,TR,10155,121,478290,31,28526,56,188,0,1,1,2016-10-01 07:22:53;
20161001,072203,TR,10155,121,478245,31,283995,69,180,0,1,1,2016-10-01 07:22:57;
20161001,072213,TR,10155,121,478136,31,282056,75,186,0,1,1,2016-10-01 07:23:10;
20161001,072223,TR,10155,121,477030,31,280439,77,226,0,1,1,2016-10-01 07:23:17;
20161001,072233,TR,10155,121,475144,31,279440,75,237,0,1,1,2016-10-01 07:23:27;
20161001,072243,TR,10155,121,473853,31,277946,78,204,0,1,1,2016-10-01 07:23:37;
20161001,072253,TR,10155,121,473236,31,276146,74,195,0,1,1,2016-10-01 07:24:27;
20161001,072303,TR,10155,121,472655,31,274385,73,197,0,1,1,2016-10-01 07:24:27;
20161001,072313,TR,10155,121,471736,31,278222,65,210,0,1,1,2016-10-01 07:24:27;
20161001,072323,TR,10155,121,470651,31,271381,69,215,0,1,1,2016-10-01 07:24:28;
20161001,072333,TR,10155,121,469283,31,270171,70,228,0,1,1,2016-10-01 07:24:28;
20161001,072343,TR,10155,121,467877,31,269145,59,229,0,1,1,2016-10-01 07:24:38;
20161001,072353,TR,10155,121,466780,31,269790,66,2140,1,1,2016-10-01 07:24:50;
20161001,072403,TR,10155,121,465496,31,266722,70,224,0,1,1,2016-10-01 07:25:02;
20161001,072413,TR,10155,121,463999,31,265643,67,233,0,1,1,2016-10-01 07:25:07;
20161001,072423,TR,10155,121,462333,31,264835,70,239,0,1,1,2016-10-01 07:25:20;
20161001,072433,TR,10155,121,460368,31,264050,77,244,0,1,1,2016-10-01 07:25:27;
20161001,072443,TR,10155,121,459305,31,263284,74,246,0,1,1,2016-10-01 07:25:39;
20161001,072453,TR,10155,121,456537,31,263045,50,279,0,1,1,2016-10-01 07:25:51;
20161001,072503,TR,10155,121,455929,31,262388,42,164,0,1,1,2016-10-01 07:25:57;
```

输出结果：

10066, 27059.2
10068, 19148
10073, 7955.76
10074, 48181.8
10081, 20162.9
10082, 350.232
10091, 36596.6
10093, 18672.9
10096, 22399.5
10100, 42828.7

20161001 00:00:00-01:00:00



大作业

- 项目名称: MonoVO
- 环境: ubuntu 14.04 cmake

| | |
|---------------|-------------------|
| camera.h | * camera.cpp |
| global.h | * imgpro.cpp |
| imgpro.h | * initialise.cpp |
| initialise.h | * iointerface.cpp |
| iointerface.h | * test.cpp |

- Camera 类: 记录 K 矩阵和畸变矫正系数;
- ImgPro 类: 关于图像处理的静态方法;
- IOinterface 类: 文件读取输出静态方法;
- Initialise 类: 计算两帧影像 R , t 以及特征点深度。

背景:

1、特征点法初始化过程，一般利用本质矩阵进行初始位姿的估计，这个过程可以利用 opencv 实现。这个过程一般没有考虑约束条件。

2、可以使用摄影测量的相对定向过程求解，但由于这个过程元素进行展开求解，并使用行列式等复杂形式，公式不够简洁。

我的工作:

1、利用李代数左乘扰动进行优化，加入基线长约束，对特征点法的初始化过程求解。

大作业

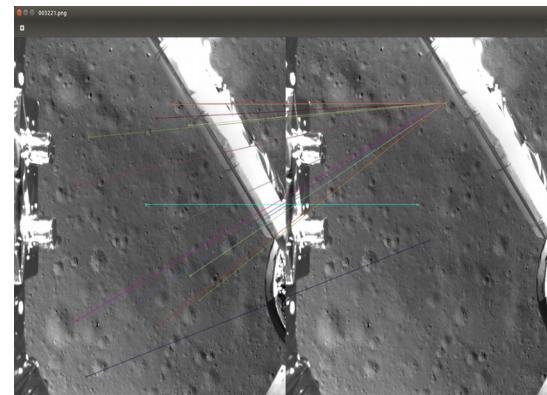
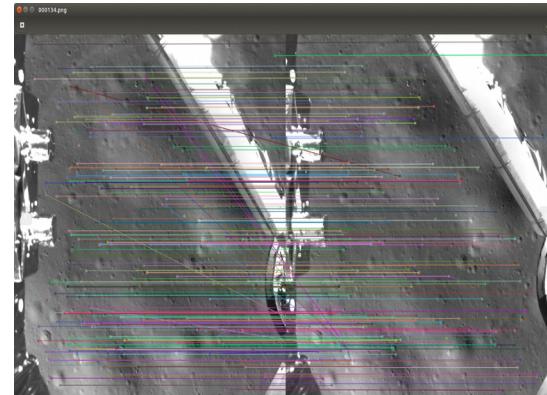
特征匹配

问题:

- 1、特征点多时，大部分正确，小部分出错；
- 2、条件苛刻时，点数太少难以计算；
- 3、利用单应矩阵进行 RANSAC，会出现多对一的情况。

解决思路:

- 1、首先使用 ORB 特征匹配；
- 2、利用单应矩阵进行 RANSAC，分析匹配结果；
- 3、结果不满足要求就用 sift 特征匹配，并进行 RANSAC



大作业

视频

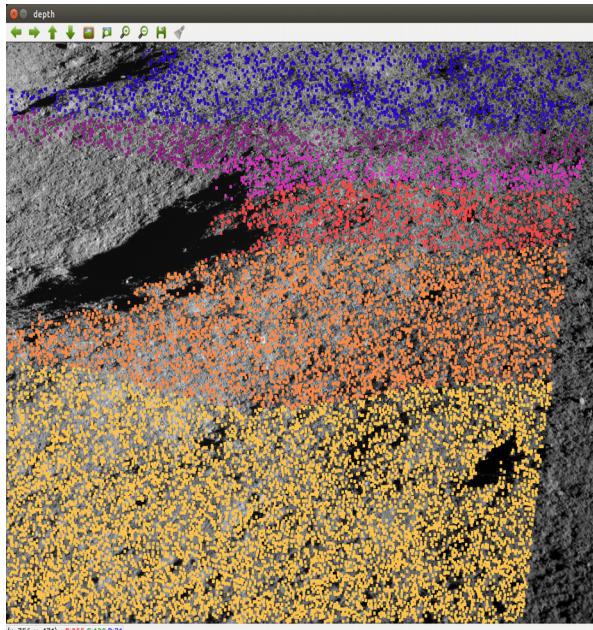
大作业

数据:

左右相机的影像

序列

特征点深度可视化



结果

Debugging starts

```
&"warning: GDB: Failed to set controlling terminal: \34:  
fai = 0.00324311 , -0.0135707 , -0.00762696  
theta = 0.911078  
euler = 0.00329503 , -0.0135582 , -0.00764942  
t = 199.93 , -4.66564 , -2.47558  
good result !
```

Debugging starts

```
&"warning: GDB: Failed to set controlling terminal: \34:  
fai = 0.00319594 , -0.00773234 , -0.0029868  
theta = 0.509011  
euler = 0.00320755 , -0.00772754 , -0.00299921  
t = 199.91 , -5.4348 , -2.51216  
good result !
```

Debugging starts

```
&"warning: GDB: Failed to set controlling terminal: \34:  
fai = 0.00332873 , -0.01064 , -0.00530976  
theta = 0.707515  
euler = 0.00335709 , -0.0106311 , -0.00532766  
t = 199.936 , -4.66846 , -1.96573  
good result !  
init done  
opengl support available  
Debugging has finished
```