

# 太阳影子定位

## 摘要

本文建立了几何模型，无约束优化模型，采用了线性最小二乘法拟合、遍历以及数字图像处理的方法进行求解，对根据影长或影尖轨迹确定拍摄地点和日期提供了参考。

针对问题一，建立了几何模型，推导出影长关于经纬度、直杆长度、赤纬角和时角的关系式。借鉴地平式日晷时间刻度的标识原理，求出影尖坐标的参数方程。求解在考虑真平太阳时差值的影响下，影长随当地时间的变化关系，对比两种情况下影长变化的区别，得出考虑真太阳时的情况下北京当地时间正午为 12 时 08 分，最短影长为 3.70 米，上午 9 时的影长为 6.99 米，下午 3 时的影长为 6.28 米，图像无对称性。

针对问题二，建立了几何模型，使用地理学中的太阳高度角公式，推导出高度角和赤纬角、纬度以及时角之间的数学关系。采用线性最小二乘法拟合求出最短影长及其对应时刻，并和北京当地太阳时的正午时刻进行比对，求出直杆安置点的经度；再将最短影长代入太阳高度角公式，求出纬度。结果显示，改变杆长，经度在 110.06 度没有变化，纬度会随杆长的增长而减小，纬度在 10.6 度至 40.4 度内变化。

针对问题三，在第二问的模型上，把太阳高度角公式中纬度和时角都用赤纬角以及真平太阳时差值表示。将进行过变量代换的太阳高度角公式作为目标函数，建立无约束优化模型。遍历 365 对太阳赤纬角和真平太阳时差值，变化  $L$  找出几对对应的最优解。结果显示，附件 2 对应的可能地点均在西藏地区（东经 83 度、北纬 23 度），日期处在 6 月及 12 月；附件 3 对应的可能地点基本集中在一条经线上（东经 105 度），且距北京的经度仅差 15 度，而且当杆长发生变化时，日期均处在 6、7 月份。

针对问题四，问题四较二三两问只是数据的获取方式不同，本问需要通过视频处理，直杆以及影长的像素长度，由实际杆长，通过相似求出影长在 40 分钟内的变化值，将数据带入问题二问题三模型，求解出可能的地点及日期。在已知日期的情况下求得的地点为内蒙古呼伦贝尔附近（东经 120.5 度，北纬 48.14 度）；在未知日期的情况下求得的地点在内蒙古呼伦贝尔附近（东经 119.02 度，北纬 48.83 度），日期为 6 月 15 日。

**关键词：**经纬度 赤纬角 时角 太阳时 无约束优化 线性最小二乘拟合

## 一、问题重述

### 1.1 背景介绍

如何确定视频的拍摄地点和拍摄日期是视频数据分析的重要方面，太阳影子定位技术就是通过分析视频中物体的太阳影子变化，确定视频拍摄的地点和日期的一种方法。

### 1.2 问题提出

1. 建立影子长度变化的数学模型，分析影子长度关于各个参数的变化规律，并应用你们建立的模型画出 2015 年 10 月 22 日北京时间 9:00–15:00 之间天安门广场（北纬 39 度 54 分 26 秒, 东经 116 度 23 分 29 秒）3 米高的直杆的太阳影子长度的变化曲线。

2. 根据某固定直杆在水平地面上的太阳影子顶点坐标数据，建立数学模型确定直杆所处的地点。将你们的模型应用于附件 1 的影子顶点坐标数据，给出若干个可能的地点。即已知影长和日期反求地点。

3. 根据某固定直杆在水平地面上的太阳影子顶点坐标数据，建立数学模型确定直杆所处的地点和日期。将你们的模型分别应用于附件 2 和附件 3 的影子顶点坐标数据，给出若干个可能的地点与日期。即已知影长求可能地点和日期。

4. 附件 4 为一根直杆在太阳下的影子变化的视频，并且已通过某种方式估计出直杆的高度为 2 米。请建立确定视频拍摄地点的数学模型，并应用你们的模型给出若干个可能的拍摄地点。如若拍摄日期未给出，也请确定出拍摄地点与日期。跟二三问不同的是要通过视频处理获取影长的数据。

## 二、问题的分析

### 2.1 问题一的分析

问题一要求建立反映影子长度变化的数学模型，分析影长关于直杆安置点、直杆长度、日期和当地时间的变化规律。

首先需要量化几个影响因素。直杆的安置点的不同反映在经纬度的不同；直杆长度会显著影响影长，对某地的同一个太阳高度，若直杆越长，则影长越长；

地球的公转引起了日期的变化，它会使得太阳直射点于一年中不断地在南北回归线之间移动，所以人们会有“夏天影短，冬天影长”的生活常识，在地理学上通常用太阳赤纬角（地球赤道平面与太阳和地球中心连线之间的夹角）反映地球绕太阳公转的几何特征；而当地时间的变化主要由地球的自转引起，在地理学上通常用时角（地球自转某度数所经历的时间）来表征时间的推移。

日晷就是通过影子的长度来读取时间，所以可以借鉴地平式日晷（晷针垂直地平面）时间刻度的标定方法，反求影子长度。

预测将可以通过几何关系求出影尖在地平面坐标上的参数方程，进而求出影长在一段时间内的变化规律。

## 2.2 问题二的分析

问题要求根据影子的轨迹、时间、日期，反求可能的地点。

这显然是问题一的反问题，但由于附件没有给出坐标建立的方向，因此想反解第一问当中的非线性方程是很难的。

查阅资料了解到，太阳高度角公式可以回避坐标的问题，只需要考虑影长和太阳高度角的关系。

预测可以通过已知的数据，拟合出正午的影长以及对应的时间，将求得的正午时间与北京当日实际的正午时间进行对比，求出经度，估计经度是一个较为固定的值。

再化简太阳高度角公式，找出纬度与太阳高度角的关系，取多个杆长的值，求出若干个可能的纬度，即找出了可能的若干个地点。

## 2.3 问题三的分析

问题要求根据影子的轨迹、时间，反求可能的地点和日期。

日期主要会影响每日的太阳时长短和赤纬角，预测可以在第二问的模型上，采用变量代换的方法，将纬度和时角都用赤纬角以及真太阳时和平太阳时的差值表示。

由于一年 365 天每一天都对应唯一的一对太阳赤纬角和太阳时差值，所以可以采用优化的方法，设立目标函数，遍历 365 对太阳赤纬角和太阳时差值，找出

最符合的那一对，既可以求出对应的位置和日期。

## 2.4 问题四的分析

第四问要求通过视频获取影长信息，估计可能的日期的地点。

因此，我们将通过 matlab 软件的图像处理工具，将原始图像转化为二值图像，以图像的左上角为像素原点，找出每一帧中影尖距离杆底以及杆尖距离杆底的像素距离，通过相似求出它们的真实长度，将问题转化为已知影长求地点和日期的问题，代入第二问和第三问的模型进行求解。

## 三、基本假设

1. 假设地平面坐标系的  $x$  轴正方向指向正东， $y$  轴正方向指向正北；
2. 假设直杆的安置点地面水平，不考虑其安置在不平处的情况；
3. 由于太阳到地球的距离很远，因此可忽略海拔对结果的影响；
4. 假设所给的数据均是准确可靠的；
5. 假设直杆安置点的天气状况良好。

## 四、符号设定

1.  $\alpha$  表示直杆安置点的地理纬度，约定北半球取正值，南半球取负值；
2.  $\beta$  表示太阳赤纬角，其表征地球公转和节气变化的一个参数，在这里约定秋分到春分取正值，春分到秋分取负值；
3.  $L$  表示直杆的长度；
4.  $t$  表示时角，其表征地球自转的一个参数，约定直杆安置点的真太阳时正午时刻  $t=0^\circ$ ，在这个时刻之前取负，在这个时刻之后取正。
5.  $\gamma$  表示影子与正北的方位夹角，约定  $\tan \gamma = x/y$ ；
6.  $h$  表示太阳高度角；
7.  $l$  表示影长；
8.  $\Delta$  真太阳时较平太阳时的差与值；
9.  $\rho$  表示经度，东经取正，西经取负。

注：其他局部符号在引用时，将给出具体说明。

## 五、问题的模型与求解

### 5.1 问题一的模型建立与求解

我们将建立反映影长变化规律的几何模型，找出影尖坐标与经纬度、直杆长度、太阳赤纬以及时角的几何变化关系，求出影尖的参数方程，绘出竖直立在天安门广场的一个 3 米长的直杆在北京时间 2015 年 10 月 22 日 9 时至 12 时影长关于时间的变化曲线。

#### 5.1.1 模型的建立

##### 1. 不考虑太阳时的情况

在这里需要借鉴地平式日晷（晷针垂直地平面）<sup>[1]</sup>时间刻度的标识原理，通过几何关系求出影尖的参数方程。

##### （1）时角

时角是表示地球自转的一个参数，即将地方时间的变化转化为自转角度的变化，由于地球的自转一圈需要 24 个小时，因此地球自转的角速度为：

$$\frac{360^\circ}{24h} = 15^\circ / h$$

在地理学上，约定当地时间正午 12 时， $t = 0^\circ$ ，上午时角为负，下午为正。跟据地球自转的角速度可以标定 5 时至 12 时对应的时角范围为  $t = -105^\circ \sim 0^\circ$ ，12 到 19 时对应的时角范围为  $t = 0^\circ \sim 105^\circ$ 。

##### （2）太阳赤纬角

太阳赤纬角<sup>[2]</sup>是地球赤道平面与太阳和地球中心连线之间的夹角，它以年为周期，其计公式为：

$$\beta = -23.45 \sin\left(\frac{2\pi(284+n)}{365}\right)$$

单位为度，其中  $n$  表示日期序号，例如：1 月 1 日， $n = 1$ ；3 月 22 日， $n = 81$ 。

##### （3）求解影尖的参数方程

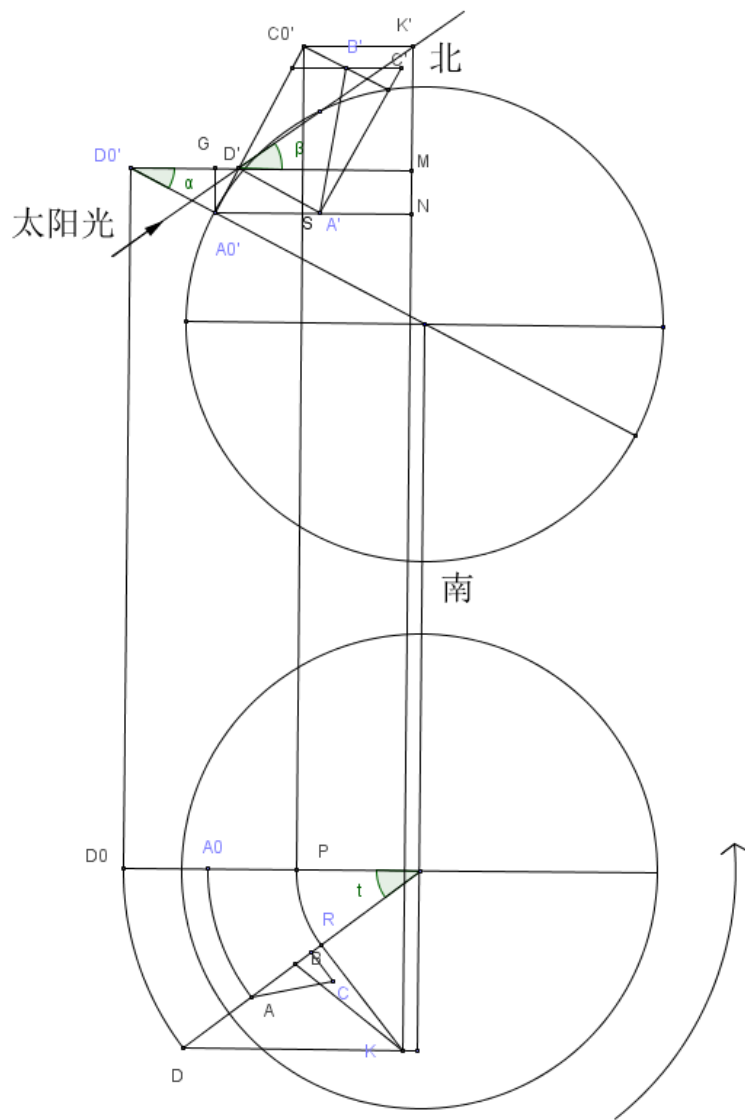


图 1 地球的侧视图和俯视图

图 1 的上图和下图分别表示地球的侧视图和俯视图。如图所示,  $ABC$  表示直杆放置的地平面,  $AD$  表示直杆, 它垂直于安置点地平面。取地方时正午 12 时直杆所在的位置为初始位置, 这时候直杆安置平面和直杆的水平投影和正面投影分别为图 1 中的  $A_0B_0C_0$ 、 $A_0D_0$  以及  $A'_0(B'_0)C'_0$  (由于观测角度的原因  $B'_0$  点与  $C'_0$  点重合, 图上无法标注)、 $A'_0D'_0$ , 直杆安置点所在的经线圈从正午转过时角  $t$  后, 直杆随之由  $A_0B_0C_0$ 、 $A_0D_0$  位置转至  $ABC$ 、 $AD$  的一般位置, 此时水平投影和正面投影分别为  $ABC$ 、 $AD$  及  $A'B'C'$ 、 $A'D'$ 。

想要求直杆一般位置时，直杆顶端  $D$  在地平面  $ABC$  上的影尖  $K$ ，从几何意

义上说，求  $K$  就是求过顶端  $D$  的一束阳光（一条直线）和地平面  $ABC$  的交点，这是一直线与一般位置平面的求交问题。

由上图可知，在  $\triangle DRK$  中：

$$x = \frac{DK}{\sin t}$$

由于  $D$  和  $D'$  是不同视角的同一个点，由  $DK = D'M$  可得：

$$DK \sin t = D'M \sin t$$

又在  $\triangle D'MK'$  中：

$$D'M = K'M \cot \beta$$

则

$$D'M \sin t = K'M \cot \beta \sin t$$

$$K'M = K'N - MN$$

其中  $K'N = B_0'S = y \cos \alpha$ ， $MN = L \sin \alpha$ 。

又由上图可知，在  $\triangle A'SC_0'$  中：

$$\angle SC_0'A_0' = \angle A_0'D_0'G = \alpha$$

则

$$y = \frac{A_0'S}{\sin \alpha}$$

其中  $A_0'S = A_0P = D_0P - D_0A_0$ ，过  $A_0$  和  $B_0$  点分别作弧交  $DR$  于  $A$ 、 $B$  两点，

因此：

$$D_0P = DR$$

$$D_0A_0 = D_0'G$$

在  $\triangle DKR$  中：  $DR = x \cos t$

在  $\triangle D_0'A_0'G$  中：  $D_0'G = L \cos \alpha$

综上，影尖的参数方程如下：

$$\begin{cases} x = (y \cos \alpha - L \sin \alpha) \cot \beta \sin t \\ y = \frac{1}{\sin \alpha} (x \cot t - L \cos \alpha) \end{cases}$$

化简得：

$$\begin{cases} x = \frac{L \sin t}{\cos \alpha (\cos t - \tan \alpha \tan \beta)} \\ y = \frac{L (\cos t \tan \alpha + \tan \beta)}{\cos t - \tan \alpha \tan \beta} \end{cases}$$

则影长表示为：

$$l = \sqrt{x^2 + y^2}$$

## 2. 考虑太阳时的情况

由于地球绕太阳运行的轨道是一个椭圆，则求相对于太阳的自转并不是均匀的，一天的真太阳时并不都是 24 小时，因此当地时间正午的时刻和地球自转的角速度都会发生变化，需要对之前的公式进行修正。

查阅真太阳时表<sup>[3]</sup>，在 10 月 22 日这一天，真太阳时在 24 小时的基础上会增加 15 分 31 秒，则北京当地时间的正午时刻为：

$$12 + \frac{24 + 15.5}{2} \frac{60}{60} \approx 12.065h$$

即当地时间正午时刻约为：12 时 4 分。

同样地球的自转角速度也会发生变化，计算得到的角速度为：

$$\frac{360}{24 + 15.5 / 60} \approx 14.84^\circ / h$$

这时候需要重新标定每个时刻对应的时角值，标定方法不变。

## 5.12 模型的求解

通过 matlab 软件求出 9 时至 12 时的影长并绘出影长随时间的变化曲线。

### 1. 不考虑太阳时的情况

下表为 9 时至 12 时几个重要时刻的影长。

表 1 不考虑太阳时情况下几个时刻的影长

当地时刻	9:00	10:00	11:00	12:00	13:00	14:00	14:52
影长/m	6.69	4.82	3.96	3.70	3.96	4.82	6.69

下图为 9 时至 12 时，影长随时间变化的曲线图。



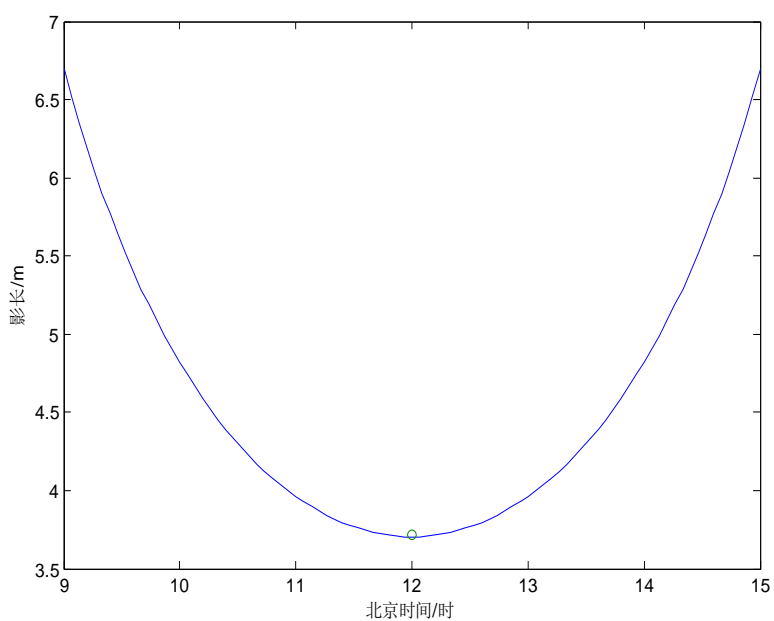


图 2 不考虑太阳时情况下影长变化曲线

## 2. 考虑太阳时的情况

下表为 9 时至 12 时，几个重要时刻的影长。

表 2 考虑太阳时情况下几个时刻的影长

当地时刻	9:03	10:10	11:02	12:08	13:02	13:59	14:52
影长/m	6.99	4.87	3.94	3.70	3.98	4.78	6.28

其中 12 时 8 分是影长最短的时刻，即当地时间正午时刻。

下图为 9 时至 12 时，影长随时间变化的曲线图。

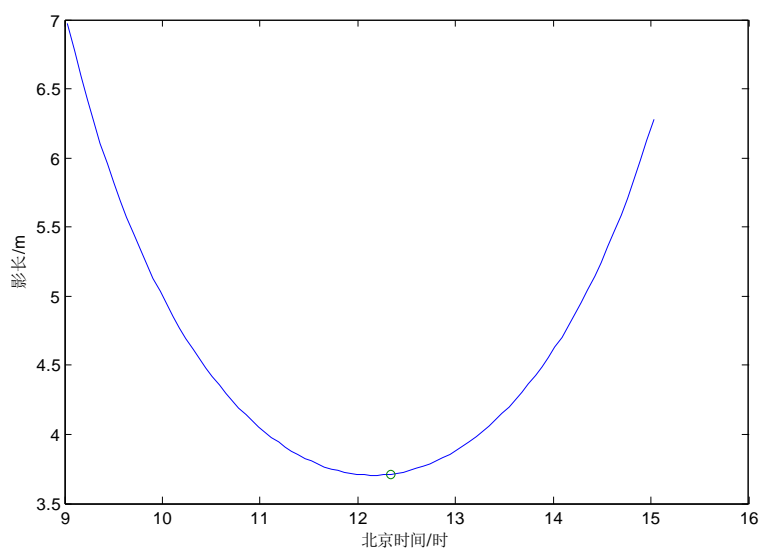


图 3 考虑太阳时情况下影长变化曲线

### 5.13 分析与总结

#### 1. 两种情况下的相同点

随着时间的推移，影子的长度先减小，后增大，且减小过程中的减小的速度会变得越来越慢，增加过程中增加的速度会越来越快。

#### 2. 两种情况下的不同点

(1) 图 2 中最低点两侧的曲线几何完全对称，而图 3 中 9 时对应的影长明显大于 15 时对应的影长。

(2) 通过对比表 1 和表 2 最短影长的精确数据发现，考虑太阳时的情况下，最短影长所对应的时刻较 12 时有 8 分钟的偏移。

综上：在考虑太阳时误差的影响下，2015 年 10 月 22 日最短影长出现在 12 时 8 分，这是北京当日真实的正午时刻；由于正午不在 12 时，图像呈现不对称性，9 时对应的影长为 6.99m，15 时对应的影长为 6.28m。

考虑太阳时造成的误差对后几个小问更精确地估计地点和日期具有实际意义。

## 5.2 问题二的模型建立与求解

我们将建立几何模型，使用地理学中的太阳高度角公式，推导出高度角和赤纬角、纬度以及时角之间的数学关系。通过给出的数据拟合出最短影长以及其对应的北京时间，并和北京当地太阳时的正午时刻进行比对，求出直杆安置点的经度；再将已知量代入几何模型，求出太阳高度角，再计算  $L$  在 0.05m-5m 范围内，不同杆长对应的纬度。

### 5.21 模型的建立

#### 1. 求纬度

地理学中的太阳高度角<sup>[4]</sup>公式为：

$$\sin h = \sin \alpha \sin \beta + \cos \alpha \cos \beta \cos t$$

之前约定直杆安置点真太阳时的正午时刻，时角为 0 度，由于之后我们将通过线性最小二乘法拟合出最短影长及其对应的时间，这个时间就是正午时刻，则公式可以简化为：

$$\sin h = \sin \alpha \sin \beta + \cos \alpha \cos \beta$$

由三角函数的和差公式得：

$$\sin h = \cos(\alpha - \beta)$$

又约定北半球的纬度为正，则

$$h = 90^\circ - (\alpha - \beta)$$

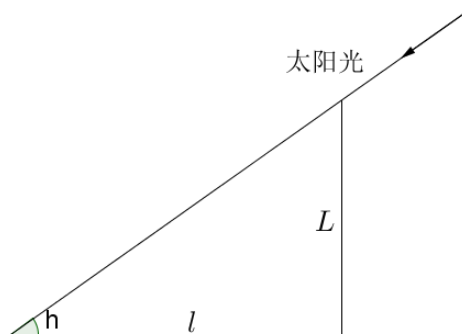


图 4 影长与杆长的函数关系

由上图可知，影长和太阳高度角的关系为

$$\tan h = \frac{L}{l}$$

则

$$h = \arctan \frac{L}{l}$$

最终纬度表示为

$$\alpha = 90^\circ + \beta - \arctan \frac{L}{l}$$

## 2. 求经度

通过拟合可以得到最短影长对应的时刻，且是用北京时间表示的，将其与北京地区真太阳时的正午时刻进行比对，求出经度。

设最短影长对应的北京时刻为  $t'$ ，假设某天北京真太阳时的正午时刻为  $\frac{24+\Delta}{2}$ ，又北京的经度为东经  $120^\circ$ ，则直杆安置点的经度为：

$$\rho = 120 - (t' - \frac{24+\Delta}{2}) \times \frac{360}{24+\Delta}$$

单位为度。

## 5.22 模型的求解

### 1. 求最短影长及其对应的北京时间

通过 matlab 软件采用线性最小二乘法，拟合出影长关于北京时间的函数图像，图像如下：

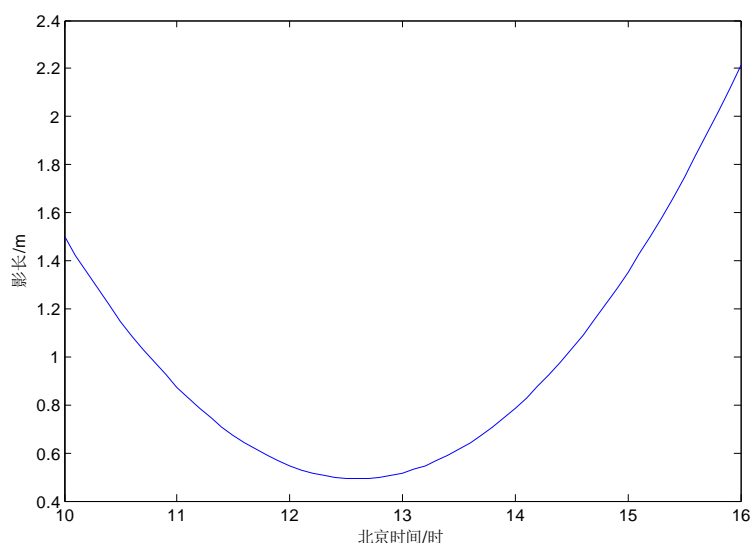


图 5 拟合出的影长随时间的变化过程

从 matlab 算出的数值结果可以得出，最短影长为 0.49m，其对应的北京时间为 12 时 36 分。

## 2.求经度

根据最短影长对应的北京时间，可以计算出直杆安置点的经度约为：东经 111.06°。

## 3.求纬度

根据影长与直杆的几何关系，可以求得高度角为：

$$h = \arctan \frac{L}{0.49}$$

将上式与  $h = 90^\circ - (\alpha - \beta)$  联立，其中赤纬角可以通过日期求出， $L$  的变化范围取 0.05m 到 5m，并以 0.05m 为间隔求出 100 个可能的地点。

在求解时，通过分析数据，发现杆长的不同只会影响纬度的变化。且直杆不能安放在海上，可以排除不可能的点，最终可能的位置如下表。

表 3 附件 2 的可能位置

杆长/m	纬度/度	经度/度	地点
0.4	40.4	111.06	内蒙古鄂尔多斯

0.45	37.1	111.06	山西吕梁市
0.5	34.1	111.06	河南三门峡市
0.55	31.3	111.06	湖北宜昌市
0.6	28.9	111.06	湖南常德市
0.65	26.7	111.06	湖南邵阳市
0.7	24.6	111.06	广西贺州市
0.75	22.8	111.06	广西梧州市
0.85	19.6	111.06	海南文昌市

### 5.23 分析与总结

最终的可能地点是一个纬度带，经度在 111.06 度附近，又由于杆长不同引起的纬度的变化，其变化范围是 10.6 度到 40.4 度。

### 5.24 模型的验证

#### 1. 误差分析

一般表示拟合程度是否良好，可以借助均方根误差（RMSE）<sup>[5]</sup>公式：

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n (observed_i - predicted_i)^2}$$

其中  $observed_i$  为观测的影子长度， $predicted_i$  为拟合出的影子长度。通过 matlab 软件求出  $RMSE$  的值为 0.00064，可以认为误差接近于 0，说明结果具有合理性。

#### 2. 验证

可以将第一问的影长和直杆长度代入问题二模型，求出地点的经纬度，并和天安门经纬坐标进行对比，看是否接近，若接近，可证明问题二模型具有合理性。

## 5.3 问题三模型建立与求解

我们将在第二问的模型上，采用变量代换的方法，将太阳高度角公式中纬度和时角都用赤纬角以及真太阳时和平太阳时的差值表示，建立无约束优化模型。遍历 365 对太阳赤纬角和太阳时差值，找出最符合即最优的那一对，既可以求出对应的位置和日期。

### 5.31 模型的建立

1. 拟合求出影长最短对应的时刻  $t_0$  及此时的影长  $l_0$ ，根据图 4 中所反应出的影长

和杆长的几何关系，可以求出太阳的高度角：

$$h_0 = \arctan \frac{L}{l_0}$$

根据正午太阳角公式  $h = 90^\circ - (\alpha - \beta)$ ，则纬度可以被表示为

$$\alpha = 90^\circ + \beta - h_0$$

2. 考虑太阳时，并将附件中给出的每个影长对应的北京时间转化为时角，即：

$$t = (t_i + \frac{\alpha - 120^\circ}{360 / (24 + \Delta)} - 12 - \frac{\Delta}{2}) \times \frac{360}{24 + \Delta}$$

则进行过变量代换之后的太阳角公式为：

$$\sin h_i = \sin(90 + \beta - h_0) \sin \beta + \cos(90 + \beta - h_0) \cos \beta \cos((t_i + \frac{\rho - 120^\circ}{360 / (24 + \Delta)} - 12 - \frac{\Delta}{2}) \times \frac{360}{24 + \Delta})$$

其中  $h_i$  是  $t_i$  时刻对应的影长，可以通过附件 2、3 得到， $\rho$  表示经度，其表达式为：

$$\rho = 120 - (t_0 - \frac{24 + \Delta}{2}) \times \frac{360}{24 + \Delta}$$

### 3. 优化模型

遍历一年每一天的  $\beta$  和  $\Delta$  值，找出最满足方程的那一对，优化模型的目标函数函数如下：

$$\min f(\beta, \Delta) = \sum_{i=1}^n |\sin h_i - \sin(90 + \beta - h_0) \sin \beta - \cos(90 + \beta - h_0) \cos \beta \cos((t_i + \frac{\alpha - 120^\circ}{360 / (24 + \Delta)} - 12 - \frac{\Delta}{2}) \times \frac{360}{24 + \Delta})|$$

该模型没有约束条件，一个无约束优化模型。

上式中  $h_i$ 、 $t_i$  是已知量， $\beta$ 、 $\Delta$  是未知量，由于每一天对应唯一的一对  $\beta$  和  $\Delta$ ，本题即要找出最匹配的一对，利用它们的值计算位置和确定日期。

## 5.32 模型的求解

通过查阅资料，作出每一天的  $\beta$ 、 $\Delta$  值表（详见），用 *matlab* 遍历求得可能的结果。

### 1. 附件 2 所对应的可能结果

表 4 附件 2 对应的可能地点和日期

杆长/m	日期	纬度/度	经度/度	地点
1	6 月 26 日	23.37	83.15	西藏阿里日土
2	12 月 16 日	23.37	83.15	西藏阿里革吉
3	12 月 15 日	23.33	83.22	西藏阿里普兰

## 2. 附件 3 所对应的可能结果

表 5 附件 3 可能对应的地点和日期

杆长/m	日期	纬度/度	经度/度	地点
2	6 月 28 日	-23.29	105.99	甘肃白银市
3	7 月 8 日	-22.48	105.66	贵州六盘水市
4	7 月 21 日	-20.44	105.62	内蒙古阿拉善盟

## 5.33 分析与总结

附件 2 对应的可能地点基本集中在西藏地区，但日期会随杆长的变化发生显著的改变，可能是冬天可能是夏天。

附件 3 对的可能地点基本集中在一条经线上，且距北京的经度仅差 15 度，而且当杆长发生变化时，日期没有发生显著的变化。

## 5.34 灵敏度分析

定性分析灵敏度，附件 2 的影尖坐标在  $x$  方向上变化不显著，所对应的地点比较集中，附件 3 的影尖坐标在  $x$  方向上变化显著，所对应的地点比较分散，因此本小问的模型对  $x$  轴方向上的变化灵敏度更高。

## 5.4 问题四的模型建立与求解

我们将通过 matlab 软件的图像处理工具，将原始图像转化为二值图像，以图像的左上角为像素原点，找出每一帧中影尖距离杆底以及杆尖距离杆底的像素距离，通过相似求出它们的真实长度，将问题转化为已知影长求地点和日期的问题，代入第二问和第三问的模型进行求解。

### 5.41 模型的建立

步骤 1: 用 matlab 软件的图像处理工具，导入视频数据；

步骤 2: 编辑代码自动将视频的每一帧导出，将导出的灰度图转化为二值图像，观察影尖像素坐标的变化规律，分析每隔多少帧，影尖坐标才会发生显著的变化；



图 6 某一帧的灰度图



图 7 某一帧的二值图

步骤 3: 分析过程中发现影尖坐标会向初始像素坐标移动, 这是由于二值图像会产生噪声引起的, 所以自动搜索的方法存在缺陷;

步骤 4: 想要精准地获得影尖像素坐标, 最好的方法是以能体现影尖坐标显著变化的帧数间隔取部分帧, 手动点出影尖坐标, 求出影长以及杆长的像素距离;

步骤 5: 通过相似, 将影长像素距离转化为实际长度, 将结果代入问题二问题三模型进行求解, 求出可能的地点及日期。

#### 5.42 模型的求解

对于已知日期的情况, 将处理出的数据代入第二问模型, 求得可能的地点为内蒙古呼伦贝尔附近 (东经 120.5 度, 北纬 48.14 度)。

对于未知日期的情况, 将处理出的数据代入第三问模型, 求得可能的地点为内蒙古呼伦贝尔附近 (东经 119.02 度, 北纬 48.83 度), 日期为 6 月 15 日。



### 5.43 分析与总结

对于未知日期情况下求得的地点和已知日期情况下求得地点基本一致，说明问题二、问题三的模型有说服力。

求得的日期与视频给出的日期基本一致，误差在一个月內，也说明求解结果有合理性。

## 六、模型的优缺点及改进方向

### 1. 模型的优点

- (1) 考虑真太阳时对当地时间正午时刻以及自转角速度的影响。
- (2) 利用太阳高度角进行问题的求解，可以避免讨论坐标系建立的方向问题。
- (3) 模型可以推广应用于全年全位置。
- (4) 拟合的结果较为准确。

### 2. 模型的缺点及改进方向

- (1) 可供拟合的点过少，拟合出的结果不够准确。若要使最后求得结果更加准确，则需在测量影长时多测几组数据，或者在拟合时用高次多项式进行逼近。
- (2) 在处理视频数据时，没有考虑到拍摄视角、拍摄距离引起的误差。要想减少这种误差，可以对原始数据进行度量纠正。

## 参考文献

- [1]吴振华,《日晷设计原理》[M], 上海:上海交通大学出版社, 2001 年
- [2]百度百科, 赤纬角, <http://baike.baidu.com/view/5777659.htm>, 2015.9.11.
- [3]百度百科, 真太阳时, <http://baike.baidu.com/view/37445.htm>, 2015.9.11
- [4]百度百科, 太阳高度角, <http://baike.baidu.com>, 2015.9.11.
- [5]百度百科, 均方根误差, <http://baike.baidu.com>, 2015.9.11.
- [6]吴振华. 常用日晷晷面线图设计[J]. 上海交通大学学报, 1996, 01: 122-128.
- [7]吴振华. 新型高精度日晷的研究与开发[J]. 上海交通大学学报, 1994, 06: 159-162.
- [8]司守奎,《数学建模算法与应用》[M], 北京: 国防工业出版社, 2011 年 5 月

## 附录

程序清单:

1. 第一问计算北京地区影长随时间变化的曲线, 并绘制图像:

不考虑太阳时:

```
t=-45*pi/180:1*pi/180:45*pi/180;
n=t*180/pi/15+12;
w=39.90777779*pi/180;
L=3;
tancw=0.196056565804194;%10月22日的赤纬角的正切值
x=L*sin(t)*(tan(w)*sin(w)+cos(w))./(cos(t)-tancw*tan(w));
y=L*(tan(w)*cos(t)+tancw)./(cos(t)-tancw*tan(w));
d=sqrt(x.^2+y.^2);
plot(n,d);
x1=xlabel('X');
x2=ylabel('Y');
hold on;
a=3.7049;
b=12.08;
scatter(b,a);
```

考虑太阳时对时角的影响:

```
t=-46.4337*pi/180:1*pi/180:42.6003*pi/180;
n=t*180/pi/14.839+12.16;
w=39.90777779*pi/180;
L=3;
tancw=0.196056565804194;%10月22日赤纬角的正切值
x=L*sin(t)*(tan(w)*sin(w)+cos(w))./(cos(t)-tancw*tan(w));
y=L*(tan(w)*cos(t)+tancw)./(cos(t)-tancw*tan(w));
d=sqrt(x.^2+y.^2);
plot(n,d);
x1=xlabel('X');
x2=ylabel('Y');
hold on;
a=3.7049;
b=12.08;
scatter(b,a);
```

2. 问题而中以附件中的坐标以及时间数据求得正午最短影长及时间:

```

clear all
y0=[1.149625826, 1.182198976, 1.215296955, 1.249051052, 1.28319534, 1.3179
93149, 1.353364049, 1.389387091, 1.426152856, 1.463399853, 1.501481622, 1.5
40231817, 1.579853316, 1.620144515, 1.661270613, 1.703290633, 1.74620591, 1
.790050915, 1.835014272, 1.880875001, 1.927918447];
x0=[14.7 14.75 14.8 14.85 14.9 14.95 15 15.05 15.1 15.15 15.2 15.25 15.3
15.35 15.4 15.45 15.5 15.55 15.6 15.65 15.7];
a=polyfit(x0,y0,2);
x=10:0.1:16;
for i=1:1:61;
    ii=i/10+9.9;
    v(i)=polyval(a,ii);
end
plot(x,v);

```

3. 凭借正午太阳高度角简化公式计算纬度值:

```

clc
clear all
for i=1:1:1000;
    LL=0.005*i;
    l=0.4936;
    chiwei=-10.51;
    h(i)=atan(LL/l)
    weidu(i)=90+chiwei-(h(i)*180/pi);
end

```

4. 第二问拟合程度的均方根误差, 并求均方根误差评价拟合是否准确:

```

clear all
y0=[1.149625826, 1.182198976, 1.215296955, 1.249051052, 1.28319534, 1.3179
93149, 1.353364049, 1.389387091, 1.426152856, 1.463399853000000, 1.50148162
2, 1.540231817, 1.579853316, 1.620144515, 1.661270613, 1.703290633, 1.74620
591, 1.790050915, 1.835014272, 1.880875001, 1.927918447];
x0=[14.7 14.75 14.8 14.85 14.9 14.95 15 15.05 15.1 15.15 15.2 15.25 15.3
15.35 15.4 15.45 15.5 15.55 15.6 15.65 15.7];
a=polyfit(x0,y0,2);
x=10:0.1:16;
for i=1:1:61;
    ii=i/10+9.9;
    v(i)=polyval(a,ii);
end
plot(x,v);

```

$n=51.56385845034887-5507098172*x0+0.296446176153666*x0.^2;$

```
rmse=sqrt(sum((n-y0).^2)/21);
```

5. 第三问以附件中的坐标以及时间数据求得正午最短影长及时间（附件2）：

```
clear all
y0=[1.247256205, 1.22279459, 1.198921486, 1.175428964, 1.152439573, 1.1299
1747, 1.10783548, 1.086254206, 1.065081072, 1.044446265, 1.024264126, 1.004
640314, 0.985490908, 0.966790494, 0.948584735, 0.930927881, 0.91375175, 0.8
97109051, 0.880973762, 0.865492259, 0.850504468];
x0=[12.68, 12.733, 12.783, 12.833, 12.883, 12.933, 12.983, 13.033, 13.083, 13.
133, 13.183, 13.233, 13.283, 13.333, 13.383, 13.433, 13.483, 13.533, 13.583, 13.
633, 13.683];
a=polyfit(x0, y0, 2);
x=11:0.1:18;
for i=1:1:71;
    ii=i/10+10.9;
    v(i)=polyval(a, ii);
end
plot(x, v);
```

6. 第三问以附件中的坐标以及时间数据求得正午最短影长及时间（附件3）：

```
clear all
y0=[3.533142184, 3.546768029, 3.561797643, 3.578100715, 3.595750783, 3.614
93428, 3.635425983, 3.657218272, 3.680541115, 3.705167836, 3.731278025, 3.7
58917911, 3.788087888, 3.818701015, 3.850809619, 3.88458522, 3.919911828, 3.
956875992, 3.99553479, 4.035750835, 4.077863059];
x0=[13.15, 13.2, 13.25, 13.3, 13.35, 13.4, 13.45, 13.5, 13.55, 13.6, 13.65, 13.7
, 13.75, 13.8, 13.85, 13.9, 13.95, 14, 14.05, 14.1, 14.15];

a=polyfit(x0, y0, 2);
x=11:0.05:18;
for i=1:1:141;
    ii=i/20+10.8;
    v(i)=polyval(a, ii);
end
plot(x, v);
```

7. 第三问搜索每天的赤纬角（附件2）：

```
clc
clear all
det=xlsread('C:\Users\Administrator\Desktop\ 赤 纬 角 与 太 阳 时
```

```

', 'B1:B365');
t=[12.683, 12.733, 12.783, 12.833, 12.883, 12.933, 12.983, 13.033, 13.083, 13.133, 13.183, 13.233, 13.283, 13.333, 13.383, 13.433, 13.483, 13.533, 13.583, 13.633, 13.683];
l=[1.247256205, 1.22279459, 1.198921486, 1.175428964, 1.152439573, 1.12991747, 1.10783548, 1.086254206, 1.065081072, 1.044446265, 1.024264126, 1.004640314, 0.985490908, 0.966790494, 0.948584735, 0.93092788, 0.91375175, 0.897109051, 0.880973762, 0.865492259, 0.850504468];
for L=1:1:10;
h0=(atan(L/0.6161))*180/pi;
for j=1:1:365;
    for i=1:1:21;
        cw_j=-23.45*sin(2*pi*(284+j)/365);
        cw_r=cw_j*pi/180;
        w_j=90+cw_j-h0;
        %w_j=h0-90+cw_j;
        w_r=w_j*pi/180;
        ew_j=120-360/(24+det(j))*(14.5-12-0.5*det(j));

m(i,j)=abs(sin(w_r)*sin(cw_r)+cos(w_r)*cos(cw_r)*cos((t(i)+(ew_j-120)/(360/(24+det(j))))-12-0.5*det(j))*360/(24+det(j))*pi/180-sin(atan(L/l(i)))));
    end
end
%end
for j=1:1:365;
    mm(:,j)=cumsum(m(:,j));
end
m;
for j=1:1:365;
    if(mm(21,j)==min(mm(21,:)))
        j
        cw_j=-23.45*sin(2*pi*(284+j)/365)
        w_j=90+cw_j-h0
        ew_j=120-360/(24+det(j))*(14.5-12-0.5*det(j))
        %w_j=h0-90+cw_j
        det(j)
    else
    end
end
end
end

```

8. 第三问搜索每天的赤纬角（附件3）：

```
clc
```

```

clear all
det=xlsread('C:\Users\Administrator\Desktop\ 赤 纬 角 与 太 阳 时
', 'B1:B365');
t=[13.15, 13.2, 13.25, 13.3, 13.35, 13.4, 13.45, 13.5, 13.55, 13.6, 13.65, 13.7,
13.75, 13.8, 13.85, 13.9, 13.95, 14, 14.05, 14.1, 14.15];
l=[3.533142184, 3.546768029, 3.561797643, 3.578100715, 3.595750783, 3.6149
3428, 3.635425983, 3.657218272, 3.680541115, 3.705167836, 3.731278025, 3.75
8917911, 3.788087888, 3.818701015, 3.850809619, 3.88458522, 3.919911828, 3.
956875992, 3.99553479, 4.035750835, 4.077863059];
for L=1:1:10;
    h0=(atan(L/3.4833))*180/pi;
for j=1:1:365;
    for i=1:1:21;
        cw_j=-23.45*sin(2*pi*(284+j)/365);
        cw_r=cw_j*pi/180;
        w_j=90+cw_j-h0;
        %w_j=h0-90+cw_j;
        w_r=w_j*pi/180;
        ew_j=120-360/(24+det(j))*(12.9-12-0.5*det(j));

m(i,j)=abs(sin(w_r)*sin(cw_r)+cos(w_r)*cos(cw_r)*cos((t(i)+(ew_j-120)
/(360/(24+det(j))))-12-0.5*det(j))*360/(24+det(j))*pi/180)-sin(atan(L/
l(i)))));
    end
end
%end
for j=1:1:365;
    mm(:,j)=cumsum(m(:,j));
end
m;
for j=1:1:365;
    if(mm(21,j)==min(mm(21,:)))
        j
        cw_j=-23.45*sin(2*pi*(284+j)/365)
        w_j=90+cw_j-h0
        ew_j=120-360/(24+det(j))*(12.9-12-0.5*det(j))
        %w_j=h0-90+cw_j
        det(j)
    else
        end
end
end
end

```

9. 提示第四问视频中的像素点:

```
clear;
filename = 'F:\\Appendix4.avi';
obj = VideoReader(filename);
numFrames = obj.NumberOfFrames;
x0 = zeros(10001);%存取每帧横坐标
y0 = zeros(10001);%存取每帧纵坐标
x0(1)=1675; %第一帧横坐标
y0(1)=867;  %第一帧纵坐标
hwait=waitbar(0,'请等待>>>>>>>');%运行进度条
for i = 2:10000
    str=['正在运行中',num2str(i/100),'%'];
    waitbar(i/10000,hwait,str);
    frame = read(obj,i);
    gray = rgb2gray(frame); %把一帧转化为灰度图
    bw = im2bw(gray,0.8);  %把灰度图二值化
    flag = 0;%退出循环的标示
    %在前一帧的影子顶点附近寻找本帧的影子顶点
    for j = -2:10
        for k = -2:10
            if bw((y0(i-1)-k),(x0(i-1)-j))==0 %判断是否为本帧影子顶点
                x0(i-1+1) = x0(i-1)-j;
                y0(i-1+1) = y0(i-1)-k;
                flag = 1;  %已找到顶点，退出本次循环
                break;
            end
        end
        if flag
            break;
        end
    end
end
end
```

10. 第四问对视频中提取数据的最小二乘拟合:

```
clear all
y0=[2.374604614, 2.356867027, 2.344991665, 2.330210078, 2.303603287, 2.297
643118, 2.288730103, 2.279860851, 2.265038254, 2.229264714, 2.223360632, 2.
211511093, 2.205606911, 2.184904395, 2.155155789, 2.143364663, 2.137434544
, 2.12266471, 2.110873246, 2.090107163, 2.081276584, 2.066558962, 2.0577283
93, 2.03415254, 2.022378307, 2.010580375, 1.995842886, 1.984068413, 1.96935
0325, 1.957575857, 1.948729445, 1.92812397, 1.92212341, 1.907405912, 1.8957
3252, 1.889674215, 1.863447178, 1.857338683, 1.842831138, 1.828201161];
x0=[8.9167, 8.933366667, 8.950033333, 8.9667, 8.983366667, 9.000033333, 9.0
```

```

167, 9. 033366667, 9. 050033333, 9. 0667, 9. 083366667, 9. 100033333, 9. 1167, 9. 1
33366667, 9. 150033333, 9. 1667, 9. 183366667, 9. 200033333, 9. 2167, 9. 233366666
7, 9. 250033333, 9. 2667, 9. 283366667, 9. 300033333, 9. 3167, 9. 333366667, 9. 350
033333, 9. 3667, 9. 383366667, 9. 400033333, 9. 4167, 9. 433366667, 9. 450033333,
9. 4667, 9. 483366667, 9. 500033333, 9. 5167, 9. 533366667, 9. 550033333, 9. 5667]
;
a=polyfit(x0, y0, 2);
x=7:0.5:15;
for i=1:1:17;
    ii=i/2+7-1/2;
    v(i)=polyval(a, ii)+0.10326667;
end
plot(x, v);

```

11. 第四问凭借正午太阳高度角简化公式计算纬度值:

```

clc
clear all

LL=2;
l=1.0327;
chiwei=-21.8255;
h=atan(LL/l)
weidu=90-chiwei-(h*180/pi);

```

12. 第四问搜索每天的赤纬角:

```

clc
clear all
det=xlsread('C:\Users\Administrator\Desktop\ 赤 纬 角 与 太 阳 时
', 'B1:B365');
l=[2.374604614, 2.356867027, 2.344991665, 2.330210078, 2.303603287, 2.2976
43118, 2.288730103, 2.279860851, 2.26503825400000, 2.22926471400000, 2.223
360632, 2.211511093, 2.205606911, 2.184904395, 2.155155789, 2.143364663000
00, 2.137434544, 2.12266471, 2.110873246, 2.090107163, 2.081276584, 2.06655
8962, 2.057728393, 2.03415254, 2.022378307, 2.010580375, 1.995842886, 1.984
068413, 1.969350325, 1.957575857, 1.948729445, 1.92812397, 1.92212341, 1.90
7405912, 1.89573252, 1.889674215, 1.863447178, 1.857338683, 1.842831138, 1.
828201161];
t=[8.9167, 8.933366667, 8.950033333, 8.96670, 8.983366667, 9.000033333, 9.0
167, 9.033366667, 9.050033333, 9.0667, 9.08336666700000, 9.100033333, 9.116
7, 9.133366667, 9.150033333, 9.1667, 9.183366667, 9.200033333, 9.2167, 9.233
366667, 9.250033333, 9.2667, 9.283366667, 9.300033333, 9.3167, 9.333366667,

```



```

9. 350033333, 9. 3667, 9. 383366667, 9. 400033333, 9. 4167, 9. 433366667, 9. 45003
3333, 9. 4667, 9. 483366667, 9. 500033333, 9. 5167, 9. 533366667, 9. 550033333, 9.
5667];

```

```

% for x=14:0.001:14.1;

```

```

L=2;

```

```

h0=(atan(L/1.0327))*180/pi;

```

```

for j=1:1:365;

```

```

    for i=1:1:40;

```

```

        cw_j=-23.45*sin(2*pi*(284+j)/365);

```

```

        cw_r=cw_j*pi/180;

```

```

        w_j=90+cw_j-h0;

```

```

%        w_j=h0-90-cw_j;

```

```

        w_r=w_j*pi/180;

```

```

        ew_j=120-360/(24+det(j))*(12.0137-12-0.5*det(j));

```

```

m(i,j)=abs(sin(w_r)*sin(cw_r)+cos(w_r)*cos(cw_r)*cos((t(i)+(ew_j-120)
/(360/(24+det(j)))-12-0.5*det(j))*360/(24+det(j))*pi/180)-sin(atan((L
-0.2)/l(i)))));

```

```

    end

```

```

end

```

```

%end

```

```

for j=1:1:365;

```

```

    mm(:,j)=cumsum(m(:,j));

```

```

end

```

```

m;

```

```

for j=1:1:365;

```

```

    if(mm(40,j)==min(mm(40,:)))

```

```

        j

```

```

        cw_j=-23.45*sin(2*pi*(284+j)/365)

```

```

        w_j=90-cw_j-h0

```

```

        ew_j=120-360/(24+det(j))*(12.0137-12-0.5*det(j))

```

```

%        w_j=h0-90-cw_j

```

```

        det(j)

```

```

    else

```

```

    end

```

```

end

```

```

% end

```

问题三附件 2 的具体结果：

L	日期	赤纬角	经度	纬度	
1	8 月 5 日	-16.83	81.65	14.8	
2	4 月 9 日	-7.15	82.35	9.97	
3	6 月 26 日	23.37	83.15	34.98	西藏阿里日土县
4	12 月 16 日	23.37	83.15	32.12	西藏阿里革吉县
5	12 月 15 日	23.33	83.22	30.36	西藏阿里普兰县
6	12 月 15 日	23.335	83.2226	29.198	
7	12 月 15 日	23.335	83.2226	28.365	
8	12 月 15 日	23.335	83.2226	27.74	
9	12 月 15 日	23.335	83.2226	27.25	
10	12 月 15 日	23.335	83.2226	26.86	

问题三附件 3 的具体结果：

L	日期	赤纬角	经度	纬度	
1	6 月 24	-23.42	106.09	50.558	
2	6 月 28 日	-23.29	105.99	36.845	甘肃省白银市
3	7 月 8 日	-22.48	105.76	26.78	贵州六盘水市
4	7 月 21 日	-20.44	105.62	20.608	内蒙古阿拉善盟
5	6 月 29 日	-23.24	105.96	11.62	
6	5 月 31 日	11.579	104.67	41.72	
7	1 月 7 日	22.42	105.7	48.87	
8	12 月 17 日	23.4	107.01	46.93	
9	12 月 17 日	23.4	107.01	44.56	
10	12 月 17 日	23.4	107.01	42.61	

第三问赤纬角与真平太阳时差值在一年中的对应关系：

第几天	赤纬角	真平太阳时差值
1	23.01163673	-0.0525
2	22.93054361	-0.060555556
3	22.84265567	-0.068333333
4	22.74799897	-0.075833333
5	22.64660154	-0.083611111
6	22.53849343	-0.090833333
7	22.42370668	-0.098333333
8	22.30227531	-0.105555556
9	22.17423529	-0.1125
10	22.03962456	-0.119444444
11	21.89848302	-0.126388889
12	21.75085248	-0.133055556

13	21.5967767	-0.139444444
14	21.43630132	-0.145833333
15	21.26947391	-0.151944444
16	21.09634389	-0.157777778
17	20.91696257	-0.163611111
18	20.73138311	-0.169166667
19	20.53966049	-0.174444444
20	20.34185152	-0.179722222
21	20.13801482	-0.184722222
22	19.9282108	-0.189444444
23	19.71250161	-0.193888889
24	19.49095119	-0.198333333
25	19.26362517	-0.202222222
26	19.03059093	-0.206111111
27	18.79191752	-0.209722222
28	18.54767565	-0.216388889
29	18.29793771	-0.221944444
30	18.04277769	-0.226944444
31	17.78227121	-0.228888889
32	17.51649546	-0.230555556
33	17.24552919	-0.232222222
34	16.9694527	-0.233611111
35	16.68834779	-0.234722222
36	16.40229777	-0.235833333
37	16.11138739	-0.236388889
38	15.81570286	-0.236944444
39	15.5153318	-0.237222222
40	15.21036321	-0.2375
41	14.90088746	-0.237222222
42	14.58699625	-0.236944444
43	14.2687826	-0.236388889
44	13.94634081	-0.235555556
45	13.61976641	-0.234722222
46	13.28915619	-0.233611111
47	12.95460809	-0.232222222
48	12.61622127	-0.230833333
49	12.27409599	-0.228888889
50	11.92833363	-0.227222222
51	11.57903665	-0.225
52	11.22630855	-0.222777778
53	10.87025385	-0.220277778
54	10.51097806	-0.184444444
55	10.14858765	-0.215
56	9.783189981	-0.211944444

57	9.414893347	-0.208888889
58	9.043806876	-0.205833333
59	8.670040531	-0.202222222
60	8.293705065	-0.198888889
61	7.914911995	-0.195277778
62	7.533773567	-0.191388889
63	7.150402718	-0.1875
64	6.764913051	-0.183611111
65	6.377418794	-0.179722222
66	5.98803477	-0.175555556
67	5.596876363	-0.171111111
68	5.20405948	-0.166944444
69	4.809700522	-0.1625
70	4.413916346	-0.157777778
71	4.016824231	-0.153333333
72	3.618541845	-0.148611111
73	3.219187206	-0.143888889
74	2.818878653	-0.139166667
75	2.417734805	-0.134444444
76	2.01587453	-0.129444444
77	1.613416909	-0.124722222
78	1.210481196	-0.119722222
79	0.807186793	-0.114722222
80	0.403653202	-0.109722222
81	-1.51E-14	-0.104722222
82	-0.403653202	-0.099444444
83	-0.807186793	-0.094444444
84	-1.210481196	-0.089444444
85	-1.613416909	-0.084444444
86	-2.01587453	-0.079166667
87	-2.417734805	-0.074166667
88	-2.818878653	-0.069166667
89	-3.219187206	-0.064166667
90	-3.618541845	-0.059166667
91	-4.016824231	-0.054444444
92	-4.413916346	-0.049444444
93	-4.809700522	-0.044722222
94	-5.20405948	-0.04
95	-5.596876363	-0.035277778
96	-5.98803477	-0.030555556
97	-6.377418794	-0.025833333
98	-6.764913051	-0.021388889
99	-7.150402718	-0.016944444
100	-7.533773567	0.012777778

101	-7.914911995	0.008333333
102	-8.293705065	0.004444444
103	-8.670040531	0.000277778
104	-9.043806876	0.003611111
105	-9.414893347	0.0075
106	-9.783189981	0.011388889
107	-10.14858765	0.015
108	-10.51097806	0.018333333
109	-10.87025385	0.021944444
110	-11.22630855	0.025277778
111	-11.57903665	0.028333333
112	-11.92833363	0.031388889
113	-12.27409599	0.034444444
114	-12.61622127	0.037222222
115	-12.95460809	0.039722222
116	-13.28915619	0.0425
117	-13.61976641	0.044722222
118	-13.94634081	0.046944444
119	-14.2687826	0.049166667
120	-14.58699625	0.051111111
121	-14.90088746	0.019444444
122	-15.21036321	0.054444444
123	-15.5153318	0.055833333
124	-15.81570286	0.057222222
125	-16.11138739	0.058333333
126	-16.40229777	0.060277778
127	-16.68834779	0.06
128	-16.9694527	0.060833333
129	-17.24552919	0.061111111
130	-17.51649546	0.061666667
131	-17.78227121	0.061666667
132	-18.04277769	0.061666667
133	-18.29793771	0.061666667
134	-18.54767565	0.061388889
135	-18.79191752	0.060833333
136	-19.03059093	0.060277778
137	-19.26362517	0.059444444
138	-19.49095119	0.058611111
139	-19.71250161	0.0575
140	-19.9282108	0.056388889
141	-20.13801482	0.055
142	-20.34185152	0.053611111
143	-20.53966049	0.051944444
144	-20.73138311	0.050277778

145	-20.91696257	0.048333333
146	-21.09634389	0.046388889
147	-21.26947391	0.044166667
148	-21.43630132	0.041944444
149	-21.5967767	0.039444444
150	-21.75085248	0.036944444
151	-21.89848302	0.034444444
152	-22.03962456	0.031666667
153	-22.17423529	0.028888889
154	-22.30227531	0.026111111
155	-22.42370668	0.023055556
156	-22.53849343	0.02
157	-22.64660154	0.016666667
158	-22.74799897	0.013333333
159	-22.84265567	0.01
160	-22.93054361	0.006666667
161	-23.01163673	0.003333333
162	-23.085911	0.000277778
163	-23.15334442	0.003888889
164	-23.21391701	0.010833333
165	-23.26761081	0.014444444
166	-23.31440992	-0.018055556
167	-23.35430046	-0.021666667
168	-23.38727062	-0.025277778
169	-23.41331063	-0.029166667
170	-23.43241276	-0.0325
171	-23.44457137	-0.036111111
172	-23.44978285	-0.039722222
173	-23.44804565	-0.043333333
174	-23.43936028	-0.046666667
175	-23.42372933	-0.050277778
176	-23.40115742	-0.053611111
177	-23.37165125	-0.056944444
178	-23.33521955	-0.060277778
179	-23.29187312	-0.063611111
180	-23.2416248	-0.066666667
181	-23.18448949	-0.069722222
182	-23.12048412	-0.072777778
183	-23.04962764	-0.075833333
184	-22.97194107	-0.078611111
185	-22.88744741	-0.081388889
186	-22.79617171	-0.083888889
187	-22.69814101	-0.086388889
188	-22.59338436	-0.088888889

189	-22.48193281	-0.091111111
190	-22.36381937	-0.093333333
191	-22.23907905	-0.095277778
192	-22.10774881	-0.097222222
193	-21.96986757	-0.098888889
194	-21.82547618	-0.100555556
195	-21.67461744	-0.102222222
196	-21.51733603	-0.103333333
197	-21.35367858	-0.104444444
198	-21.18369356	-0.105555556
199	-21.00743137	-0.106388889
200	-20.82494422	-0.106944444
201	-20.63628618	-0.1075
202	-20.44151317	-0.108055556
203	-20.2406829	-0.108055556
204	-20.03385488	-0.108055556
205	-19.82109039	-0.108055556
206	-19.60245249	-0.107777778
207	-19.37800596	-0.107222222
208	-19.14781731	-0.106666667
209	-18.91195474	-0.105833333
210	-18.67048816	-0.104722222
211	-18.4234891	-0.103611111
212	-18.17103077	-0.102222222
213	-17.91318797	-0.100833333
214	-17.65003711	-0.099166667
215	-17.38165615	-0.0975
216	-17.10812464	-0.095555556
217	-16.82952363	-0.093333333
218	-16.54593566	-0.091111111
219	-16.25744477	-0.088611111
220	-15.96413645	-0.086111111
221	-15.66609762	-0.083333333
222	-15.36341658	-0.080555556
223	-15.05618303	-0.0775
224	-14.744488	-0.074166667
225	-14.42842387	-0.070833333
226	-14.10808428	-0.067222222
227	-13.78356417	-0.063611111
228	-13.45495968	-0.06
229	-13.1223682	-0.055833333
230	-12.78588828	-0.051944444
231	-12.44561963	-0.0475
232	-12.10166307	-0.043333333

233	-11.75412053	-0.038888889
234	-11.40309498	-0.034166667
235	-11.04869045	-0.029722222
236	-10.69101196	-0.024722222
237	-10.33016549	-0.02
238	-9.966257972	0.015
239	-9.599397234	0.009722222
240	-9.229691988	0.004722222
241	-8.857251784	0.000555556
242	-8.482186986	0.005833333
243	-8.104608732	0.011388889
244	-7.724628908	0.016666667
245	-7.34236011	0.022222222
246	-6.957915612	0.027777778
247	-6.571409333	0.033611111
248	-6.182955804	0.039166667
249	-5.792670132	0.045
250	-5.400667967	0.050833333
251	-5.007065467	0.050833333
252	-4.611979266	0.056666667
253	-4.215526435	0.0625
254	-3.817824453	0.068333333
255	-3.418991168	0.074166667
256	-3.019144761	0.08
257	-2.618403717	0.086111111
258	-2.216886783	0.091944444
259	-1.814712938	0.098055556
260	-1.412001354	0.103888889
261	-1.008871364	0.109722222
262	-0.605442423	0.115833333
263	-0.201834077	0.121666667
264	0.201834077	0.1275
265	0.605442423	0.133333333
266	1.008871364	0.139166667
267	1.412001354	0.145
268	1.814712938	0.150555556
269	2.216886783	0.156111111
270	2.618403717	0.161666667
271	3.019144761	0.167222222
272	3.418991168	0.1725
273	3.817824453	0.177777778
274	4.215526435	0.183055556
275	4.611979266	0.188333333
276	5.007065467	0.193333333



277	5. 400667967	0. 193333333
278	5. 792670132	0. 198055556
279	6. 182955804	0. 203055556
280	6. 571409333	0. 207777778
281	6. 957915612	0. 212222222
282	7. 34236011	0. 216666667
283	7. 724628908	0. 221111111
284	8. 104608732	0. 221111111
285	8. 482186986	0. 225277778
286	8. 857251784	0. 229166667
287	9. 229691988	0. 233055556
288	9. 599397234	0. 236944444
289	9. 966257972	0. 240555556
290	10. 33016549	0. 243888889
291	10. 69101196	0. 247222222
292	11. 04869045	0. 250277778
293	11. 40309498	0. 253333333
294	11. 75412053	0. 189166667
295	12. 10166307	0. 258611111
296	12. 44561963	0. 261111111
297	12. 78588828	0. 263333333
298	13. 1223682	0. 265277778
299	13. 45495968	0. 266944444
300	13. 78356417	0. 268611111
301	14. 10808428	0. 27
302	14. 42842387	0. 271111111
303	14. 744488	0. 272222222
304	15. 05618303	0. 272777778
305	15. 36341658	0. 273333333
306	15. 66609762	0. 273611111
307	15. 96413645	0. 273611111
308	16. 25744477	0. 273333333
309	16. 54593566	0. 273055556
310	16. 82952363	0. 2725
311	17. 10812464	0. 271388889
312	17. 38165615	0. 270277778
313	17. 65003711	0. 269166667
314	17. 91318797	0. 2675
315	18. 17103077	0. 265555556
316	18. 4234891	0. 263611111
317	18. 67048816	0. 261388889
318	18. 91195474	0. 258888889
319	19. 14781731	0. 256111111
320	19. 37800596	0. 253055556

321	19.60245249	0.25
322	19.82109039	0.246388889
323	20.03385488	0.242777778
324	20.2406829	0.238888889
325	20.44151317	0.235
326	20.63628618	0.230555556
327	20.82494422	0.226111111
328	21.00743137	0.221388889
329	21.18369356	0.216388889
330	21.35367858	0.211111111
331	21.51733603	0.205833333
332	21.67461744	0.200277778
333	21.82547618	0.194444444
334	21.96986757	0.188333333
335	22.10774881	0.182222222
336	22.23907905	0.175833333
337	22.36381937	0.169166667
338	22.48193281	0.1625
339	22.59338436	0.155833333
340	22.69814101	0.148611111
341	22.79617171	0.141388889
342	22.88744741	0.134166667
343	22.97194107	0.126666667
344	23.04962764	0.119166667
345	23.12048412	0.111666667
346	23.18448949	0.103888889
347	23.2416248	0.096111111
348	23.29187312	0.088055556
349	23.33521955	0.08
350	23.37165125	0.071944444
351	23.40115742	0.063888889
352	23.42372933	0.055833333
353	23.43936028	0.0475
354	23.44804565	0.039444444
355	23.44978285	0.031111111
356	23.44457137	0.022777778
357	23.43241276	0.014444444
358	23.41331063	0.006388889
359	23.38727062	0.001944444
360	23.35430046	0.010277778
361	23.31440992	-0.018333333
362	23.26761081	-0.026666667
363	23.21391701	-0.034722222
364	23.15334442	-0.042777778

365	23.085911	-0.050833333
-----	-----------	--------------

第四问视频帧像素点坐标:

坐标	杆顶		杆底		影尖	
	x	y	x	y	x	y
1	893	207	869	883	1672	869
2	893	207	869	883	1666	869
3	893	207	869	883	1662	870
4	893	207	869	883	1657	870
5	893	207	869	883	1648	870
6	893	207	869	883	1646	871
7	893	207	869	883	1643	872
8	893	207	869	883	1640	872
9	893	207	869	883	1635	873
10	893	207	869	884	1624	873
11	893	207	869	884	1622	873
12	893	207	869	884	1618	874
13	893	207	869	884	1616	874
14	893	207	869	884	1609	875
15	893	207	869	885	1600	876
16	893	207	869	885	1596	876
17	893	207	869	885	1594	877
18	893	207	869	885	1589	878
19	893	207	869	885	1585	878
20	893	207	869	886	1579	879
21	893	207	869	886	1576	879
22	893	207	869	886	1571	879
23	893	207	869	886	1568	879
24	893	207	869	886	1560	880
25	893	207	869	886	1556	880
26	893	207	869	886	1552	881
27	893	207	869	886	1547	882
28	893	207	869	886	1543	882
29	893	207	869	886	1538	882
30	893	207	869	886	1534	882
31	893	207	869	886	1531	883
32	893	207	869	886	1524	883
33	893	207	868	886	1521	884
34	893	207	868	886	1516	884
35	892	207	868	886	1512	884
36	892	207	868	885	1509	884
37	892	207	869	886	1502	884
38	892	207	869	885	1499	885
39	893	207	870	886	1496	886

40	892	207	870	886	1491	886
----	-----	-----	-----	-----	------	-----

由此求得的影长：

北京时间	影长
8:55	2.374604614
8:56	2.356867027
8:57	2.344991665
8:58	2.330210078
8:59	2.303603287
9:00	2.297643118
9:01	2.288730103
9:02	2.279860851
9:03	2.265038254
9:04	2.229264714
9:05	2.223360632
9:06	2.211511093
9:07	2.205606911
9:08	2.184904395
9:09	2.155155789
9:10	2.143364663
9:11	2.137434544
9:12	2.12266471
9:13	2.110873246
9:14	2.090107163
9:15	2.081276584
9:16	2.066558962
9:17	2.057728393
9:18	2.03415254
9:19	2.022378307
9:20	2.010580375
9:21	1.995842886
9:22	1.984068413
9:23	1.969350325
9:24	1.957575857
9:25	1.948729445
9:26	1.92812397
9:27	1.92212341
9:28	1.907405912
9:29	1.89573252
9:30	1.889674215

9:31	1.863447178
9:32	1.857338683
9:33	1.842831138
9:34	1.828201161

第四问拟合曲线：

