2011 3rd International Conference on Advanced Computer Control (ICACC 2011)

An Integrated System for Astronomical Telescope Based on Stellarium

Jitong Chen
Software College
Northeastern University
Shenyang, China
e-mail:read3389@hotmail.com

Xiaonan Wang
Software College
NortheasternUniversity
Shenyang, China
e-mail:loveyiren@sina.com

Lingquan Meng
Software College
Northeastern University
Shenyang, China
e-mail:mlq2133248@163.com

Chenhui Wang
Software College
Northeastern University
Shenyang, China
e-mail:wchneu@163.com

***Abstract*—Astronomical telescope is a powerful tool for star observation, but adjusting a telescope is time-consuming. For most of astronomy amateurs, their telescopes don't have an intelligent system to ease the adjustment of telescope. Therefore, we propose an integrated system for astronomical telescope. The integrated system presents a friendly telescope control interface and provides functions including telescope monitoring, star targeting, star profile retrieval and star tracking. The most powerful feature of the integrated system is star tracking. By using camshift algorithm and PID controller, the star tracking module can help user track certain objects such as plant and comet.**

Keywords-telescope control; camshift; star tracking; integrated system

I. INTRODUCTION

With the development of telescope, an intelligence control system is required for better use of telescope. The system should be able to collect information from an astronomical telescope and control the behavior of the telescope. Although some commercial astronomical telescopes for amateurs contain a tiny control system, it cannot provide enough functions to professional observers. Besides, manually searching information for a target star or adjusting telescope by hand are time consuming. In this paper, we propose a new integrated system which can retrieve profile from database for the target star and automatically track fast moving object such as comets and planets. The star tracking function can replace human to automatically point the astronomical telescope to the target star and keep the target in the central field of vision. Our system is base on stellarium [1], an open source sky map software, which is friendly to developers for its complete documentation and API. By adding in modules to Stellarium, we enable stellarium to communicate with the astronomical telescope and perform star tracking task.

II. BACKGROUND

A. Equatorial Coordinate System

Because the altitude and azimuth of a star are constantly changing, it is not possible to use the horizontal coordinate system in a catalogue of positions. A more convenient coordinate system for cataloguing purposes is based on the celestial equator [2] and the celestial poles [3]. Ascension and declination are defined in a similar manner to latitude and longitude on the surface of the Earth. The equatorial coordinate system (shown in Fig. 1) allows all earthbound observers to describe the apparent location in the sky of sufficiently distant objects using the same pair of numbers: the right ascension α and declination δ [4]. A given star's position remains constant in equatorial coordinate system. Therefore, we should adjust the telescope to fit equatorial coordinate system for convenience of observation. In other words, we should align telescope mount with equatorial coordinate system or set the polar axis of the telescope parallel to Earth's axis.

B. Tracking a Star

Since the position of a star is roughly constant in equatorial coordinate system, the right ascension α and declination δ are constant. However, the telescope is moving as Earth rotates. From the perspective of telescope, δ is constant while α has changed to $\alpha + \Delta\alpha$. $\Delta\alpha$ indicates Earth rotation. Therefore, we can keep the star in the central field of vision by offsetting the effect from Earth rotation. In other words, telescope should revolute around the telescope polar axis, thus eliminating $\Delta\alpha$.

C. Tracking a Planet or Comet

Tacking a star doesn't require a complex control of telescope while tracking a planet or comet can be very difficult.

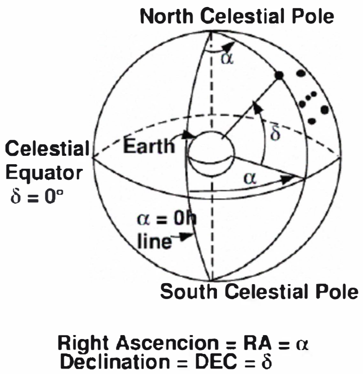

Figure 1. Celestial coordinate system

The ascension and declination of a planet or comet are both changing obviously from time to time because they are much closer to Earth than stars do. In this case, we need an algorithm to identify the target planet in the video from the telescope and a control algorithm to adjust the telescope so as to keep the target in the central field of vision.

III. SYSTEM ARCHITECTURE

As showed in Fig. 2, the system is mainly composed of four functional modules namely telescope communication module, stellarium module, tracking module and database retrieval module. First, the telescope communication module is responsible for gathering information from telescope and receiving command from tracking module and stellarium module. The information gathered from the telescope includes real-time video, right ascension α and declination δ of the pointing position of the telescope. Second, the stellarium module interacts with the other three modules to provide a user interface where the observer can clearly see the current target object in star map. From the interface the user can see which constellation the target object belongs to and a profile of the target object. The profile is retrieved from database where profiles of stars and planets are recorded. Third, the tracking module performs tracking algorithms to adjust the telescope. The tracking module also interacts with stellarium and stellarium module can inform tracking module to change the tracking target. Fourth, the database retrieval module is responsible for matching profiles to target objects and updating profiles of objects.

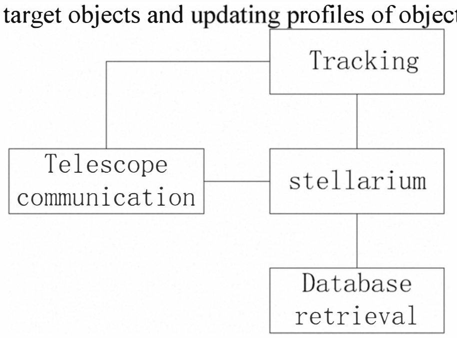

Figure 2. System architecture

IV. SYSTEM IMPLEMENTATION

In this section, we give a detailed introduction of stellarium module and tracking module illustrated in the architecture of system because of their complexity. Besides, brief introductions of communication module and database retrieval model are also presented.

A. *Stellarium Module*

Stellarium is a free open source planetarium for computer. It shows a realistic sky in 3D, just like what you see with the naked eye, binoculars or a telescope. In our system, the pointing position of telescope is sent to Stellarium module and the position is marked on simulated sky rendered by Stellarium. Therefore the user can easily find where the telescope is pointing. Also, the user can mark a new target on the simulated sky on Stellarium, and then stellarium module will send command to the telescope and point the telescope to the new target. Stellarium module also shows the profile of selected object in the simulated sky, which is quite convenient for observer to get the various parameters of the target star or planet. The interface of stellarium module is shown in Fig. 3.

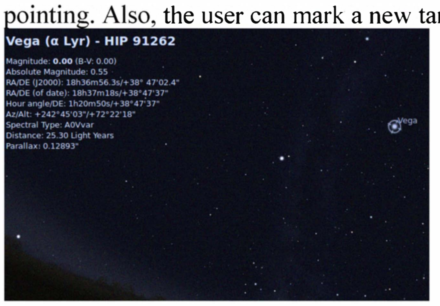

Figure 3. Stellarium interface shows the simulated sky

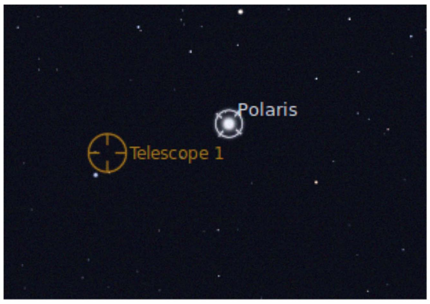

Figure 4. Mark of telescope and target

In Fig. 4, the bigger circle represents the current pointing position of the telescope and the smaller circle stands for the new target. After we set the new target, stellarium sends command to telescope and the telescope is gradually adjusted to the new target.

B. *Tracking module*

In order to track a moving planet or comet, the tracking module receives the real-time video from the telescope and performs target recognition algorithm and telescope adjustment algorithm.

1) *Target recognition algorithm*

Target recognition algorithm is base on camshift (Continuously Adaptive Mean Shift) algorithm [5]. Camshift is a modified version of the mean shift algorithm [6]. We choose camshift algorithm for two reasons. First, camshift has been proved to be efficient in real-time object tracking [7]; second, camshift is computationally inexpensive and easy to implement. Camshift algorithm has following steps:

a) Set the target and mark tracking target in the first frame of the video. Then a histogram is built according to color image in the target region. The histogram is only created once and it represents the color distribution of target. As showed in Fig. 5, the colored bar in the histogram represents a hue and the height of that bar indicates how many pixel in target region have this hue. The histogram will be used as a reference in following video frame.

b) When a new video frame arrives, each pixel of that frame is assigned a target probability. The probability is calculated by comparing the hues of a pixel and the histogram created in the previous step. A pixel which has high similarity with the color distribution described in the histogram is assigned a high probability, which means that pixel is likely to be one of the pixels in target region.

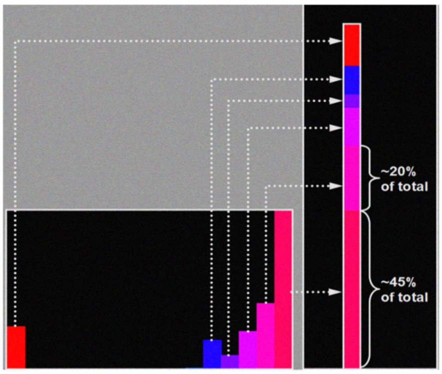

Figure 5. Histogram of target region

c) Camshift estimates the location of target in the new frame and keep it centered over the area with high target-probability pixels. Camshift starts searching the new location from the previous location in the last frame, calculates the center of gravity of the target-probability values with a rectangle and put the rectangle over the center of gravity. The algorithm keeps searching until find the new location of target. In the process, camshift not only keeps moving the rectangle but also adjust the size and angle of the rectangle to find the new location of the target.

A more detailed process of camshift algorithm is illustrated in Fig. 6.

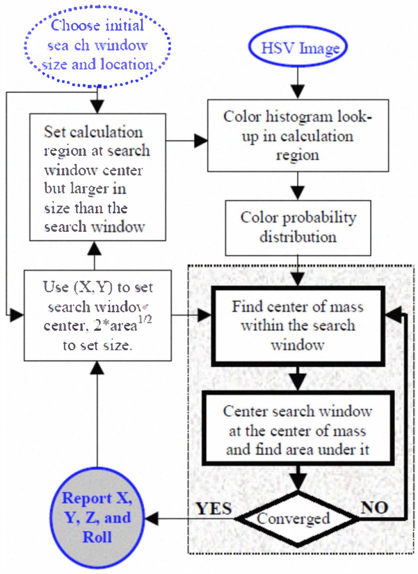

Figure 6. Process of camshift algorithm

Fig. 7 shows an example of application of camshift on a video taken from an astronomical of telescope. We apply camshift algorithm to track Saturn in the video taken from the telescope. We use circle to mark the target (Saturn) in the video. In the right of figure 7 is a frame of video. On the left of figure 7 is the color histogram of target (Saturn).

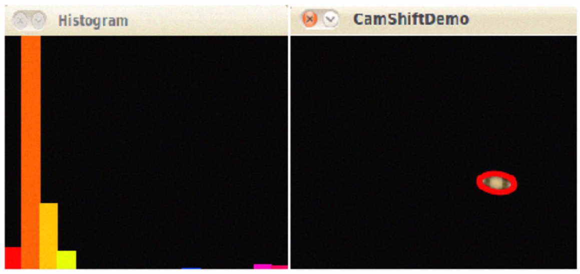

Figure 7. Application of camshift algorithm on video taken from telescope, the target is Saturn

The Target recognition algorithm is implemented with OpenCV [8], a library of programming functions for real time computer vision. Here are build-in functions for camshift in OpenCV:

- cvCreateHist() // create histogram
- cvCalcBackProject() // assign target-probability to each pixel
- cvCamShift() //shift the search windows until find the new location of target

2) Telescope Adjustment Algorithm

We use PID controller to adjust the telescope in order to keep the target in the center of telescope view. As showed in Fig. 8, (x, y) denotes the position of the object in the image.

The angle between the target and the center of telescope view is:

$$\theta=\left(\frac{(x-x_c)\omega}{L},\frac{(y-y_c)\omega}{L}\right) \tag{1}$$

Where L denotes the diameter of view in the image, ω denotes the field of view, (x_c, y_c) the coordinate of image center.

The process of telescope adjustment is described in Fig. 9. ϕ is a parameter which decides how often the telescope should be adjusted. ϕ should be tuned in order to roughly keep the target in the central field while minimize the frequency of adjustment.

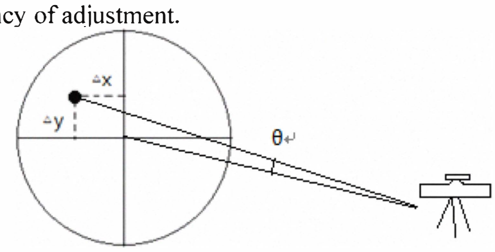

Figure 8. View from telescope

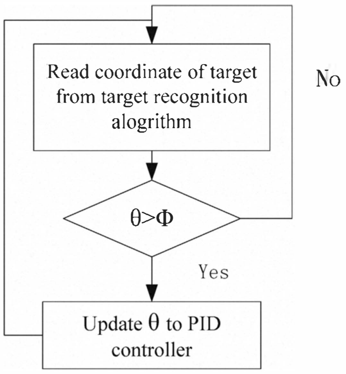

Figure 9. Telescope adjustment

3) *Telescope Communication module*

The Telescope Communication module is implemented on embedded system and provides wireless connection. The communication module is responsible for taking command and transmitting real time video and parameters of telescope. The parameters of the telescope include right ascension and declination of telescope, longitude and latitude of the telescope, altitude of telescope and direction of telescope. The longitude and latitude of the telescope is read from GPS and the direction of telescope is read from electronic compass.

4) *Database module*

The database module is implemented in MySQL. The database keeps profiles of objects. To add a new profile, user should fill in a form such as table 1. The attributes are not limited in these showed in table 1. Keeping the record of unknown object is usually helpful for the observer to revisit the object and carry on a study on the object. When the star tracking module is activated, the Database mode in corporation with other modules will automatically record the parameter of the tracking target. Based on these records, we can draw the track of the moving target on stellarium simulated sky. It can contribute to the further study of that object.

TABLE I. OBJECT PROFILE

Name	RA /DEC	Magnitude

V. EXPERIMENT

Our system is implemented in Ubuntu 10.04. We use our system to track the moving of Jupiter. The result shows our system can steadily track the moving Jupiter and keep Jupiter roughly in the center of telescope view.

VI. CONCLUSION

This paper proposes a new integrated system for astronomical telescope. The system provides various functions including telescope control, sky simulation, target profile retrieval and target tracking. The system can ease the work of observer and contribute to astronomy research. However, the tracking module does not perform well under some circumstance, especially when the tracking target is partially masked by another object. In future work, we will try to improve the target tracking module and make it more stable.

REFERENCES

[1] stellarium http://www.stellarium.org/ [last accessed on July 8, 2010]

[2] celestial equator http://en.wikipedia.org/wiki/Celestial_equator [last accessed on July 8, 2010]

[3] celestial pole http://en.wikipedia.org/wiki/Celestial_pole. [last accessed on July 8, 2010]

[4] Hong yunfang, "The handbook for astronomy fans" Sichuan Dictionary Press, Jan . 2006, pp. 187–194

[5] Bradski, G.R., "Computer Vision Face Tracking for Use in a Perceptual User Interface," IEEE Workshop on Applications of Computer Vision, Princeton, NJ, 1998, pp.214-219.

[6] D. Comaniciu, V. Ramesh, and P. Meer, "Real-Time Tracking of Non-Rigid Objects Using Mean Shift," Proc. IEEE Conf. Computer Vision and Pattern Recognition, vol. II, pp. 142-149, June 2000

[7] J. G. Allen, R. Y. D. Xu, and J. S. Jin. "Object tracking using CamShift algorithm and multiple quantized feature spaces," In Conferences in Research and Practice in Information Technology, volume 36, 2003.

[8] OpenCv wiki http://opencv.willowgarage.com/wiki/ [last accessed on July 8, 2010]