

Design and Implementation of a Python-Based Automatic Photometry Program (Postprint)

Authors: Huiyu Yuan, Zhao Juan, Yang Yuangui

Date: 2023-01-18T00:00:00+00:00

Abstract

This paper designs and implements an automatic photometry program based on Python. First, file names are scanned to achieve file classification; then data checking, preprocessing, and aperture photometry are performed, and the photometric data are saved; finally, the observer specifies the target star and comparison stars, and the program extracts the corresponding magnitude and observation time data from the photometric results to obtain a light curve based on differential photometry. To solve the star map matching problem in data extraction, a vector-feature-based star map matching algorithm for automatically tracking moving targets is proposed, which can automatically match star maps with large offsets and track continuously moving asteroids. Using this program to process binary star and asteroid observation data separately and comparing with IRAF (Image Reducation and Analysis Facility) photometry results yields maximum deviations of 0.04 mag and 0.043 mag, respectively, and standard deviations of ± 0.005 mag and ± 0.007 mag, respectively.

Full Text

Design and Implementation of Automatic Photometry Program Based on Python

Yuan Huiyu¹, Zhao Juan², Yang Yuangui^{2*} (1. College of Information, Huaibei Normal University, Huaibei, Anhui 235000, China 2. Huaibei Normal University, Huaibei, Anhui 235000, China)

Abstract

This paper designs and implements an automatic photometry program based on Python. The program first scans filenames to achieve file classification, then performs data inspection, preprocessing, and aperture photometry while saving the photometric results. Finally, after the observer specifies the target star

and comparison star, the program extracts the corresponding magnitudes and observation times from the photometric results to obtain light curves based on the differential photometry method. To solve the problem of image matching in data extraction, we propose an improved image matching algorithm based on vector features that can automatically track moving objects, match images with large offsets, and track continuously moving asteroids. The program is used to process observational data of binary stars and asteroids respectively. Compared with IRAF (Image Reduction and Analysis Facility) photometry results, the maximum deviations are 0.04 mag and 0.043 mag respectively, with standard deviations of ± 0.005 mag and ± 0.007 mag.

Keywords: Automatic Photometry; Differential Photometry; Image Matching; Asteroid Tracking

Astronomical observation is one of the common methods for obtaining electromagnetic radiation flux from celestial bodies. Many special phenomena such as binary eclipses, stellar pulsations, chromospheric activity and flares, supernova explosions, and gamma-ray bursts can all be studied through photometric data. With modern technological development, two-dimensional CCD detectors are widely used for photometry due to their high quantum efficiency and wide wavelength response range. Since absolute calibration is relatively difficult, differential photometry is commonly used to study variations in celestial brightness to eliminate sky background radiation from thermal and fluorescent emission.

In recent years, with the development of ground-based and space-based sky survey projects (such as the domestic GWAC wide-field survey and the upcoming Chinese Space Station Survey Telescope), massive amounts of photometric data require timely processing. Additionally, some transient sources discovered during observations, such as stellar flares and gamma-ray bursts, require real-time data processing. Therefore, developing an automatic photometric analysis software package is essential.

IRAF, developed by the National Optical Astronomy Observatory, is widely used for processing photometric data. However, the software is cumbersome to use and requires considerable manual operation. To improve photometric efficiency, some researchers have developed automatic photometry packages based on IRAF, which have been applied in some telescopes but have not become widely adopted. With the rise of the Python language, astronomical data processing packages have emerged, enabling us to develop an automatic photometry program based on Python that can analyze photometric data and extract light curves.

1. Algorithm Design

To eliminate the effects of atmospheric turbulence and weather conditions, differential photometry is typically used to obtain target light curves. This method uses another star of constant brightness in the same field of view as a comparison star, calculating the brightness difference between the target and comparison

stars to obtain accurate light curves. The photometric workflow in this program is divided into two parts: performing photometry on all stars in the image, and then extracting the target star brightness as needed.

1.1.1 Data Preprocessing

During observations, CCD noise, background, and pixel response non-uniformity all affect the reliability of results. Therefore, in addition to target images (img), bias and flat-field images must also be taken for background elimination and flat-field correction to remove these effects. The processing algorithm is given by equation (1). Based on this algorithm, the data preprocessing workflow is briefly described as follows:

- (1) Combine multiple bias images to obtain the master bias file (biasm).
- (2) Subtract the biasm file from flat files to obtain the flatb file.
- (3) Combine multiple flatb files to obtain the master flat-field file (flatm).
- (4) Subtract the biasm file from img files, then divide by the flatm file to obtain the true flux data image (imgr).

1.1.2 Aperture Photometry

For relatively small celestial objects in images, aperture photometry is typically used to determine their brightness. The algorithm consists of three steps: (1) sky background determination and subtraction; (2) target detection; and (3) aperture photometry. We use the SEP algorithm proposed by Bertin & Arnouts to complete aperture photometry.

Due to atmospheric scattering, scattered light enters the CCD and becomes the sky background in the imgr data. SEP divides the image into rectangular blocks and calculates the median and standard deviation of all pixels in each block. It then applies a median filter to the pixel points, removing those that deviate from the median by more than 3 times the standard deviation. This process repeats until no more pixels can be removed, ultimately obtaining the sky background data, which is then subtracted from imgr.

SEP uses a multi-threshold detection algorithm to solve the target detection problem. It first sets a minimum extraction threshold to find candidate points, then divides 30 thresholds according to the extraction threshold and peak values. Based on these thresholds, the image is constructed into a tree diagram. Starting from the branches and moving toward the trunk, it traverses and analyzes whether adjacent points belong to the same target star, thereby determining the position and shape of the target star.

Based on the obtained target positions, SEP uses a two-dimensional Gaussian surface to fit an elliptical aperture that can include all pixels of the target star. It integrates the brightness of pixels within the aperture to obtain information such as brightness, position, and shape of stars in the imgr file for use in the next step of data extraction.

1.2 Target Matching and Data Extraction

Due to telescope tracking errors, the field of view at different times will have offsets, causing the target star to shift continuously in the image. Currently, there are various algorithms for matching star positions in images, such as those based on triangles, radial and cyclic features, and vector features. We compared existing algorithms and found that the vector feature-based algorithm has higher efficiency. However, when the image offset is large, there is a possibility of matching failure. Additionally, asteroids continuously move in star images, and existing algorithms cannot automatically track asteroids. This paper proposes an improved vector feature-based star image matching algorithm that can automatically track targets.

- (1) Select one star image as the template image. First, specify the target star for which data needs to be extracted in the image and record its original coordinate values. Then generate the positioning vector feature set based on this template image. The generation method is as follows: Exclude the specified target star from the stars in the central region of the template image, sort the remaining stars by brightness, select the top $m+1$ stars to form the positioning star set, use the star S_0 closest to the center of this set as the positioning reference star, calculate the vectors of the remaining m stars relative to S_0 , and constitute the positioning vector feature set.
- (2) Use R to match other star image files. The method is as follows: Sort the stars in the image to be matched by brightness and take the top n stars to obtain the set to be matched. Let s_1 be the reference star, calculate the coordinates of the other m positioning stars based on R , and check whether these m stars exist in the image to be matched. If the new coordinates of a positioning star are outside the image, assume that the star exists and continue matching the remaining stars.
- (3) If all m positioning stars exist, the matching is successful. The offset of relative to S_0 is the offset of the image to be matched relative to the template image. Based on this offset, obtain the new position of the target star in the file to be matched and extract its brightness information.
- (4) Update the original coordinate values of the target star based on its new position and offset.
- (5) If not all m stars can be found, the matching is unsuccessful, and s_1 is not the reference star. Try assuming s_2 as the reference star, and so on. If none of the n stars match successfully, then the file matching fails.

Compared with traditional vector feature-based star image matching algorithms, the algorithm in this paper can automatically handle the problem of some comparison stars shifting out of the image due to large image offsets.

2. Data Experiments

Astronomical data is typically saved in FITS (Flexible Image Transport System) format and requires corresponding program packages for analysis and reading. This paper uses the binary star MMCom and asteroid 52Eur as examples, completing the entire data processing workflow using Python packages including *astropy*, *numpy*, *sep*, *pandas*, and *cv2*.

The observation data directory typically includes three types of files: *bias.fits* (*bias files*), *flat.fits* (flat-field files), and *obj*.fits* (target files). We scan the filenames, analyze and record the complete filenames, and then automatically complete the data preprocessing work. Figure 1 shows the original data image and preprocessed data image of MMCom. As can be seen from Figure 1, the uneven brightness phenomenon in the image disappears after preprocessing.

Using SEP for aperture photometry on preprocessed images, we obtain target position and size information. As shown in Figure 2, brightness, position, aperture, and other information are stored in separate CSV files for subsequent processes.

Select one star image as the template image and specify the position from which data needs to be extracted. As shown in Figure 3, the red circles are two manually specified targets, where #1 is the target star and #2 is the comparison star. In Figure 3(b), 52Eur continuously moves in the star image. The program then automatically selects the 5 brightest stars outside the target for positioning, where R is the reference star and the 4 yellow lines are the vector set used for star image matching. Based on this set, the program automatically matches and tracks the target in other star images and extracts data.

To verify the obtained data, we performed photometry on the data using IRAF and plotted the results from both methods in Figure 4. As can be seen from the figure, the results from the two methods are basically consistent. The maximum deviations of photometry results for MMCom and 52Eur are 0.04 mag and 0.043 mag respectively, with standard deviations of ± 0.005 mag and ± 0.007 mag respectively.

3. Conclusions and Outlook

Although IRAF is powerful, its usage and installation process is complex, and secondary development is difficult. This paper develops a fully automatic photometry program using Python, applicable to Windows, Linux, and Mac operating systems, completing all functions from directory scanning and file type analysis to final data extraction. The test results are consistent with IRAF, featuring high operational efficiency and ease of use. We hope to communicate with domestic astronomers to further improve the program's functionality.

Currently, the Huaibei Normal University astronomical research team purchased a 30cm telescope in 2015 with an optimal observation magnitude of about 7-10 mag, discovering some large flares. In 2022, they purchased a 50cm plane-wave

telescope, which is currently in the debugging and trial operation stage. We plan to retrofit these two telescopes (located at the Xinglong Station of the National Astronomical Observatories, GWAC group) for remote observation, which will generate massive amounts of photometric data. Developing this automatic photometry program in advance provides reliable technical support for real-time processing of photometric data. At the same time, it provides a data platform for the development of astronomy at local universities and also offers development opportunities for related majors such as big data and optoelectronic information technology.

References

- [1] Turpin D, Wu C, Han X, et al. The mini-GWAC optical follow-up of gravitational wave alerts—results from the O2 campaign and prospects for the upcoming O3 run[J]. *Research in Astronomy and Astrophysics*, 2020, 20(1): 13.
- [2] Gong Y, Liu X, Cao Y, et al. Cosmology from the Chinese Space Station Optical Survey (CSS-OS)[J]. *The Astrophysical Journal*, 2019, 883(2): 203.
- [3] Xin L P, Li H L, Wang J, et al. A $\Delta R = 9.5$ mag superflare of an ultracool star detected by the SVOM/GWAC system[J]. *The Astrophysical Journal*, 2021, 909(2): 106.
- [4] Kuznetsov A A, Kolotkov D Y. Stellar superflares observed simultaneously with Kepler and XMM-Newton[J]. *The Astrophysical Journal*, 2021, 912(1): 81.
- [5] Collaboration H, Abdalla H, Aharonian F, et al. Revealing x-ray and gamma ray temporal and spectral similarities in the GRB 190829A afterglow[J]. *Science*, 2021, 372(6546): 1081-1085.
- [6] Acciari V A, Ansoldi S, Antonelli L A, et al. MAGIC observations of the nearby short gamma-ray burst GRB 160821B[J]. *The Astrophysical Journal*, 2021, 908(1): 90.
- [7] Qiu G L, Liang J H, Su Y B, et al. Research for aperture in CCD photometry[J]. *Journal of Yunnan Normal University (Natural Sciences Edition)*, 2011, 31(4): 26-29.
- [8] Su Y B, Guo Y G, Zhang X. CCD photometry with IRAF[J]. *Journal of Yunnan Normal University (Natural Sciences Edition)*, 2009, 29(6): 1-4.
- [9] Zhang Y, Lin Z, Guo D F. An application of the DAOPHOT package in the IRAF[J]. *Astronomical Research & Technology*, 2011, 8(4): 410-416.
- [10] Zhang H J, Zhang X, Zheng Y G, et al. IRAF automatic data processing for the CCD photometry of blazars[J]. *Astronomical Research & Technology*, 2008, 5(2): 99-105.
- [11] Bertin E, Arnouts S. SExtractor: software for source extraction[J]. *Astronomy and Astrophysics Supplement Series*, 1996, 117(2): 393-404.

- [12] Pál A, Bakos G Á. Astrometry in wide-field surveys[J]. Publications of the Astronomical Society of the Pacific, 2006, 118(848): 1474-1483.
- [13] Groth E J. A pattern-matching algorithm for two-dimensional coordinate lists[J]. The Astronomical Journal, 1986, 91(5): 1244-1248.
- [14] Zhang G, Wei X, Jiang J. Full-sky autonomous star identification based on radial and cyclic features of star pattern[J]. Image and Vision Computing, 2008, 26(7): 891-897.
- [15] Tabur V. Fast algorithms for matching CCD images to a stellar catalogue[J]. Publications of the Astronomical Society of Australia, 2007, 24(4): 189-198.
- [16] Valdes F G, Campusano L E, Velasquez J D, et al. FOCAS automatic catalog matching algorithms[J]. Publications of the Astronomical Society of the Pacific, 1995, 107(717): 1119-1128.

Note: Figure translations are in progress. See original paper for figures.

Source: ChinaXiv – Machine translation. Verify with original.