Design and Implementation of a Wide-Field Optical Transient Source Identification System (Postprint)

Authors: Ma Dong, Wu Chao, Tian Haijun, Wei Jianyan

Date: 2017-10-20T00:00:00+00:00

Abstract

The Ground-based Wide-Angle Camera Array (GWAC), a large-field optical transient survey facility in China, employs automated preliminary screening to identify nearly a thousand transient candidates every 2.5 minutes. These candidates may correspond to supernovae, variable stars, moving objects, or noise. To further manage and conduct detailed screening and authentication of these transient candidates, we propose a design scheme for an optical transient authentication system based on the Django web development framework, which implements backend database management and frontend web-based interactive display functionalities. Through research, this scheme primarily features management of transient candidates; processing and display of light curves; cross-matching and classification with known catalogs of various types of celestial objects; and provision of a human-computer interactive manual authentication interface. Benefiting from the excellent features of the Django framework, the system can provide a good command-line interactive interface without additional development, facilitating scientists to conduct object-oriented operations on the database through Django' s interface. By applying the system to test observation data from the mini Ground-based Wide-Angle Camera Array (mini-GWAC) already constructed in China, the results demonstrate that the system can correctly classify and authenticate transient candidates, providing scientists with rich information and convenient, practical operational tools for the subsequent authentication of transient candidates. Additionally, the system holds similar practical significance for other analogous transient survey projects.

Full Text

Abstract

China' s wide-field optical transient survey facility, the Ground Wide Angle Camera (GWAC), initially selects nearly a thousand transient candidates every 2.5 minutes through automated machine screening. These candidates may be supernovae, variable stars, moving objects, or noise. To further manage and thoroughly screen these transient candidates, we propose a design scheme for an optical transient identification system based on the Django web development framework, which implements backend database management and frontend web interactive display functions. Through research, this system primarily provides: management of transient candidates; light curve processing and display; cross-identification with multiple known celestial catalogs for classification; and a human-computer interactive interface for manual verification. Benefiting from Django' s excellent features, the system can provide a good command-mode interactive interface without additional development, enabling scientists to perform object-oriented database operations through its interface. Testing the system with observational data from China' s established Mini-GWAC pilot array demonstrates that the system can correctly classify and identify transient candidates, providing scientists with rich information and convenient operational tools for subsequent verification. The system also has practical significance for other similar transient survey projects.

Keywords: Cross-identification; Optical data visualization; Django

Introduction

With the development of modern astronomical observation technology and data processing techniques, time-domain astronomy has progressed toward larger fields of view, fainter detection depths, and higher temporal sampling rates. Benefiting from these advances, transient source surveys have become a focal point in astronomy, leading to the discovery of increasingly numerous unknown transients. Major projects such as the US-supported Large Synoptic Survey Telescope (LSST), the Palomar Transient Factory (PTF), the Catalina Real-time Transient Survey (CRTS), and China' s wide-field transient survey facility, the Ground Wide Angle Camera (GWAC), exemplify this trend. GWAC employs eighteen 18-cm wide-angle cameras with a total field of view of [value], producing one observation image every 2.5 minutes. This represents a significant improvement over traditional transient surveys in both field of view and temporal sampling, while also imposing higher demands on data processing and analysis methods.

The system successfully applies classical image subtraction techniques [**?**] to GWAC for transient detection. By performing image position matching, point spread function profile matching, and flux scaling between real-time observations and template images (high-quality images from earlier observations), the subtracted residual images clearly reveal any brightening or newly appeared celestial targets. Using optimized feature parameters in a random forest classification algorithm, the method can automatically and rapidly extract transient candidates from residual images. Among these candidates are moving objects (including artificial satellites and asteroids), variable stars (including microlensing events), transient phenomena occurring in galaxies (such as supernovae and gravitational wave bursts), and noise. Rapid identification and preliminary classification of these candidates is crucial for follow-up observations, especially for short-timescale transients.

System Design

To address this need, we developed a transient identification system capable of timely classification and management of transient candidates. The system assists observers in rapid analysis and judgment through three main functional aspects: (1) Cross-identification with multiple known catalogs to determine whether candidates are known celestial objects and their nearest neighbors. The catalogs used include the International Astronomical Union' s Minor Planet Center database, the HyperLeda galaxy catalog, variable star catalogs, and the UCAC4 all-sky survey catalog. This enables identification of whether candidates are asteroids, known stars, or brightening events occurring in galaxies. (2) Processing and real-time display of light curves for transient candidates, providing auxiliary reference information for further classification. (3) Providing an interactive interface for astronomers to manually verify candidate types or noise by examining cross-identification results, light curves, and identification images, thereby determining whether follow-up observations are warranted.

This paper introduces the system design, starting from the basic requirements and functions of the transient candidate identification system. We propose a network-based architecture with both web and command-line interactive layers, detailing the system' s architecture and functional module design, implementation processes for each module, and discussing technical advantages and disadvantages. The system is applied to observational data from the established Mini-GWAC pilot array to verify its functionality and effectiveness, with results discussed and system features summarized.

Data Flow and Architecture

The system' s data input originates from transient candidates automatically identified by the random forest classification algorithm from residual images after image subtraction. The primary data characteristics include positional information, photometric data, and temporal information, with nearly a thousand candidate updates every 2.5 minutes. The system first implements management of transient candidates, completing preliminary identification and classification primarily through cross-identification with various known observation catalogs, and determining whether further follow-up observations are needed. The main known catalogs include the minor planet catalog, galaxy catalog, and variable star catalog. Finally, temporal cross-identification is performed to determine the frequency of candidate appearances and generate light curves, helping to identify noise and providing reference data for manual-assisted classification.

The system adopts a database-driven network development approach using the Django web framework, with PostgreSQL as the database platform for its good compatibility with Django and stable operation. The system provides both web-based and command-line methods for backend data query and analysis, enabling astronomers to quickly access and analyze data across different devices and facilitating deeper customized analysis through command-line data retrieval.

The system architecture is shown in [Figure 1: see original paper]. The presentation layer is designed with both web interaction and command-line modes, providing user-system interaction interfaces. Through this layer, users can obtain basic photometric parameters of transient candidates, identification images after subtraction, cross-identification results with multiple catalogs, and manual classification confirmation functions. The business logic layer is the core of the system, integrating service components required by astronomers for transient candidate identification and performing data interaction operations with the data resource layer to enable rapid candidate access and retrieval while shielding the business logic layer from underlying implementation details of the data resource layer, enhancing system transparency.

User management services provide hierarchical user permission management for data administrators and ordinary users, allowing direct backend login for data management and operations or frontend access for data viewing and modification within permission scopes. The data resource layer manages both structured data (various attribute parameters of transient candidates) and unstructured data (identification images and other image files), employing both database and file systems for management. The infrastructure layer provides a reliable physical environment for system operation. The development and testing environment uses Scientific Linux (SL) servers, with web and database servers currently on the same physical machine, though practical deployment will consider separating them.

Functional Modules

The overall functional objective of the optical transient identification system is to achieve classification and identification of transient candidates. The main functional modules include: database management platform, light curve visualization, user interactive identification, user management, temporal data cross-identification, and cross-identification with known catalogs.

Database Management Platform

The database management platform primarily designs the following data tables: "OTcands" for storing observational attribute data of transient candidates in temporal sequence; "OTuniq" for storing unique positional information of transient candidates after temporal cross-identification (this table does not store temporal data); "Fchart" for storing identification images; "Comments" for storing user annotations; "Catfiles" for information related to each image, such as Julian date and sky region number; and "Users" for system user management. The final database model is shown in [Figure 2: see original paper].

Cross-Identification

Temporal data cross-identification matches new data with historical data through two-point distance calculations, enabling classification of the same target source and generation of light curves. It also determines the historical appearance frequency of transient candidates—sporadic noise appears less frequently, allowing noise sources to be filtered by match count.

Cross-identification with known catalogs matches transient candidates against the minor planet catalog, galaxy catalog, variable star catalog, and all-sky survey catalog to determine classification attributes and provide basis for observers to decide on follow-up observations.

Light Curve Visualization

Light curve visualization rapidly queries and retrieves light curve data from the database and displays it intuitively on web pages as two-dimensional scatter plots, creating a good visual user experience.

User Interactive Identification and Management

User interactive identification and management provides observers with comprehensive auxiliary information for judging candidate types, including cross-identification results with multiple catalogs, light curves, and identification images. Astronomers can manually interactively verify candidate types, with the system completing confirmation from both status and category perspectives: "confirmed transient" , "pending confirmation" , and "confirmed non-transient" . To provide convenient interactive interfaces, the template layer uses HTML technology for webpage display, form radio buttons, and CSS styling. The view layer then calls the model layer through API to complete business logic operations.

Technical Implementation

Technical Architecture and Patterns

The system development selects the Python language platform, which has rich third-party astronomical software development kits and friendly cross-platform deployment capabilities [?]. Django is Python' s web development framework.

The designed system architecture follows the Model-View-Controller (MVC) pattern, which offers efficient system development and maintenance characteristics [?].

The database management platform uses PostgreSQL for good compatibility with Django. Based on GWAC' s transient auto-identification system metadata and the conceptual/logical/physical database model designs, Django' s built-in Object-Relational Mapping (ORM) component simplifies data access and retrieval.

The concrete implementation process for objectifying data tables involves: creating a database model layer by objectifying each database table—this is the foundation. In the models.py file, tables are created by inheriting django.db.models.Model and defining fields using Django' s field types. Before using the model layer, the settings.py configuration file must specify database connection details: database engine, name, login username, server address, and port. Notably, this configuration process is completely independent of the model layer, offering the advantage that changing the backend database type requires no modifications to the model layer, greatly reducing coupling between these layers. Through these two steps, Django' s data mapping component easily achieves table objectification while providing developers with a simple yet powerful database access interface for object-oriented database operations.

Cross-identification is the core functional module, 主要包括时序数据的交叉认证以及与已知星表的交叉认证. The basic principle of cross-identification algorithms is calculating distances between newly observed target stars and reference stars to find the nearest neighbor.

Cross-Identification Implementation

For different transient candidates appearing in the unique table after temporal cross-identification, if the continuous appearance count is less than [threshold], they are preliminarily judged as noise. Cross-identification with known catalogs then determines classification results.

First, cross-identification is performed with the International Astronomical Union' s Minor Planet Center database. The process involves: (1) Using Django' s database access interface to obtain right ascension and declination attributes of different candidates from the unique table; (2) Accessing the minor planet catalog through API; (3) Calculating the minimum distance between the target and minor planets using the great-circle distance formula. If this distance is smaller than the cross-identification radius (a quantity related to measurement error), the target is considered a known minor planet, and relevant information (minimum distance, magnitude, type, name, etc.) is updated to the unique table.

For candidates not matching the minor planet catalog, cross-identification proceeds with the galaxy catalog, primarily through the locally established Hyper-

Leda database. Transients occurring in galaxies are high-priority targets for GWAC follow-up observations. For candidates not classified into the galaxy catalog, cross-identification with variable star catalogs is performed similarly through the SIMBAD website' s variable star catalog. Testing and theoretical analysis show this process identifies numerous known and new variable stars.

For still-unmatched candidates, cross-identification continues with survey catalogs to identify known survey targets; otherwise, they may be unknown objects requiring manual examination of light curves and identification images for comprehensive analysis and follow-up observation decisions.

Light Curve Visualization Implementation

For light curve visualization, the ECharts charting library is introduced in the template layer to implement dynamic bubble tooltips and data region zooming functions. The system uses two-dimensional scatter plots for light curve data, displaying detailed node information on mouse hover and allowing data region selection when showing large datasets—particularly useful for analyzing short-period transient light curve characteristics. The visualization interface is shown in [Figure 5: see original paper].

User Interface Implementation

Manual verification by astronomers represents the final judgment 环节 for transient candidates. The system provides comprehensive auxiliary information for type determination, including cross-identification results with multiple catalogs, light curves, and identification images. Using these references, astronomers can interactively verify candidate types, with the system completing confirmation from both status and category perspectives: "confirmed transient", "pending confirmation", and "confirmed non-transient".

To provide convenient interactive interfaces, the template layer uses HTML and CSS technologies to achieve webpage display effects with form radio buttons. The view layer calls the model layer through APIs to define view functions. When observers mark candidate types and click submit, the web server passes the URL request to Django' s dispatcher, which locates the defined view function to perform corresponding business logic operations, feeding manual judgments and annotations back to the backend database for reference by other astronomers. User login is required.

The implementation uses JavaScript and jQuery [?] for webpage partial updates without reloading entire pages. When users click submit buttons, the interface automatically updates user annotation information and counters without full page reload.

Application Verification

The system' s verification testing uses observational data from Mini-GWAC, the pilot project of GWAC, which consists of [number] wide-angle cameras. Mini-GWAC has similar data processing 流程 to GWAC and is operational at Xinglong Observatory, providing real observational data. Verification employs data from [number] nights across two fields. The main interface after system operation is shown in [Figure 7: see original paper], allowing astronomers to quickly browse data sorted by temporal appearance frequency, pop up identification images, view 2D scatter light curves, and examine other observations. Verification status and types are displayed in the main interface, along with observer comments.

Statistical analysis shows a total of [number] transient candidate records, with instrumental noise points and fast-moving objects accounting for approximately 54% of total records. Through temporal cross-identification, [number] target sources appear [number] times or more. Cross-identification with known catalogs identifies [number] known asteroids (~20%), [number] candidates matching galaxy positions, and [number] matching survey catalogs. Candidates appearing [number] times or fewer are sporadic, accounting for approximately [percentage] of total records. To further verify automatic classification correctness, several typical variable stars were selected for external photometric analysis [Figure 8: see original paper], analyzing their light curve characteristics. Results show the system' s automatic variable star classification is basically correct.

The verification testing demonstrates that this transient candidate identification system can automatically perform correct preliminary classification, greatly reducing samples requiring manual verification while effectively managing all transient candidates.

Discussion and Conclusion

Aiming at the large number of transient candidates produced by China' s wide-field optical transient survey facility GWAC and its pilot project Mini-GWAC, we have developed a transient candidate management and classification identification system. This system provides astronomers with abundant auxiliary identification information to assist rapid analysis and judgment.

The system mainly implements three functions: (1) Cross-identification with multiple known catalogs to determine whether candidates are known celestial objects and their nearest neighbors, enabling identification of whether candidates are asteroids, known stars, or brightening events in galaxies; (2) Processing and real-time display of transient candidate light curves to provide auxiliary reference information for further classification; (3) Providing interactive interfaces for astronomers to manually verify candidates by examining cross-identification information, light curves, and identification images to determine types and confirm follow-up observation needs.

The Django-based implementation enhances development efficiency, achieves high cohesion and low coupling, reduces interdependence between view and template layers, and strengthens system functionality. The system greatly reduces the number of candidates requiring manual verification while effectively managing all transient candidates. Mini-GWAC observational data testing shows the system can correctly perform preliminary classification, reducing manual verification samples to [percentage]. It also provides astronomers with good operational interfaces.

The system' s functions and implementation principles have practical significance for other similar transient survey projects, especially wide-field transient surveys.

References

\cite{Astronomical Society Chun W J. Python Bissex P Forcier J 徐洋, 吴潮, 万萌, 等——国家天文台台刊, Django LSST Public website sitemap EB / OL 2015-10-31 . http / www. lsst. org / lsst / science / scientist_transient. Alard C. Image subtraction using a space-varying kernel 363-370. . Astronomy & Astrophysics Supplement Bramich D M. A new algorithm for difference image analysis . Monthly Notices of the Royal 核心编程 [L7-L81.}

[?, ?, ?, ?, ?]

[?, ?]

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