



KIET GROUP OF INSTITUTIONS

Connecting Life with Learning



A
Project Synopsis
on
NovaEye: AI Detecting the Universe's Hidden Events
submitted as partial fulfillment for the award of
BACHELOR OF TECHNOLOGY
DEGREE

SESSION 2025-26

in

Name of discipline

By

Abhishek Kumar 2400290109001

Ananya Pandey 2300290100048

Radhika Saxena 2300290100195

Ananya Pandey 2400290109004

Under the supervision of

Dr. Neeti Pahuja

KIET Group of Institutions, Ghaziabad

Affiliated to

Dr. A.P.J. Abdul Kalam Technical University, Lucknow
(Formerly UPTU)
Academic Year 2025-26

Index

1. Abstract
2. Introduction.....
2.1 Motivation & Gaps.....
2.2 Contribution.....
2.3 Technology and Field.....
3. Literature Survey.....
4. Methodology / Planning of Work.....
5. Facilities Required.....
6. References (IEEE).....

1. Abstract (not to exceed 1 page)

Modern wide-field sky surveys produce very large volumes of image data and transient alerts (e.g., LSST expected >10 million alerts/night). Manual inspection is infeasible. This project develops an end-to-end, production-oriented pipeline for automated detection and ranking of rare astronomical transients (supernovae, kilonovae, microlensing events, variable stars) by combining optimal image subtraction (ZOGY), robust source extraction, feature engineering, and machine learning (CNN-based real-vs-bogus classifier plus autoencoder/isolation-forest anomaly detection). The pipeline includes astrometric and photometric calibration, PSF modeling, stamp-level CNN inference, and an interactive visualization dashboard (Streamlit) for alert triage. Expected deliverables are a working prototype processing archival ZTF FITS images, pre-trained RB and anomaly models, documentation, and an alert-ranking dashboard. Performance targets: RB precision >95% and processing latency suitable for rapid follow-up.

2. Introduction

2.1 Brief Project Overview

Time-domain astronomy studies objects whose brightness or position change on timescales from seconds to months. Detecting and classifying astronomical transients promptly enables critical follow-up (spectroscopy, multi-wavelength observations) and scientific discoveries. This project implements a software pipeline that ingests FITS images (science + reference), performs astrometric/photometric calibration, applies optimal image subtraction (ZOGY), extracts candidates (SEP/Photutils), computes morphological/photometric/astrometric features, and classifies candidates using machine learning.

2.2 Motivation

Large surveys (ZTF, Pan-STARRS, ATLAS, LSST) generate extremely high alert rates; manual vetting is impossible. Automated, ML-powered pipelines enable discovery acceleration, reduce false positives, and prioritize candidates for scarce follow-up resources. The project addresses the urgent need for a reproducible, open-source pipeline tailored to archival ZTF-like data but extensible to LSST-scale operations.

2.3 Gaps in Existing Work

- Many existing broker systems (ALeRCE, fink, Lasair) focus on classification at scale but assume large operational infrastructure. A compact, reproducible pipeline focusing on ZOGY-based subtraction + stamp-level CNN + anomaly detection for educational and prototyping use is desirable.
- Anomaly discovery of novel transient types remains challenging; combining autoencoder reconstruction error with isolation-forest scoring provides an avenue to surface unusual candidates missed by supervised RB classifiers.

2.4 Contribution of This Project

1. Implement or wrap a ZOGY-based image subtraction module suitable for ZTF FITS images.
2. Develop a stamp-level CNN real-vs-bogus classifier trained on curated ZTF real & bogus examples.
3. Build an autoencoder + isolation-forest anomaly detection module to highlight novel candidates.
4. Integrate components into an end-to-end pipeline with a Streamlit dashboard for visualization and alert export.
5. Package the system with documentation and reproducible deployment artifacts (Docker).

2.5 Technology Used and Field

Field: Time-domain Astronomy / Astronomical Image Processing.

Primary technologies: Python scientific stack (Astropy, Photutils, NumPy, SciPy, scikit-learn, PyTorch), SEP for source extraction, Streamlit for dashboarding, Docker for reproducible deployment.

3. Literature Survey

Modern astronomical transient-detection systems have evolved significantly over the past two decades, driven by the increasing volume of sky-survey data and the scientific need for rapid identification of rare events. The literature in this domain spans four major areas: **image-differencing algorithms, survey pipelines and alert brokers, machine learning techniques for real–bogus (RB) classification, and anomaly/outlier detection for rare transient discovery.**

3.1 Image Subtraction Techniques

Image subtraction is the foundation of transient detection. Traditional approaches such as the **Alard & Lupton (1998)** method introduced spatially varying convolution kernels to match PSFs between images. This technique became widely used due to its robustness for small-scale surveys.

Later, **Bramich (2008)** proposed a purely numerical kernel solution that avoided assumptions of separable Gaussian basis functions, improving subtraction fidelity in crowded fields.

A revolution came with **Zackay, Ofek & Gal-Yam (2016)** who introduced **ZOGY**, an optimal Fourier-domain subtraction method. ZOGY analytically maximizes the signal-to-noise ratio of detected sources, producing not only a cleaner difference image but also a *score image* indicating statistically significant detections. Due to its optimality, ZOGY forms the backbone of modern large-scale time-domain surveys.

3.2 Survey Pipelines and Real-Time Alert Systems

Large-scale surveys like the **Zwicky Transient Facility (ZTF)** and the upcoming **Vera Rubin Observatory LSST** produce millions of nightly detections. Their pipelines incorporate automated calibration, astrometric fitting, PSF estimation, and image differencing.

ZTF, described by **Bellm et al. (2019)**, represents a benchmark system, capable of delivering real-time transient alerts within minutes. These alerts are disseminated to community brokers.

Multiple **alert brokers**—such as **ALeRCE, fink, and Lasair**—process these detections further. They provide real-time classification, cross-matching with catalogs, and distribution to astronomers. These brokers highlight the need for compact, reproducible pipelines for research and academic purposes, which this project aims to emulate at a smaller scale.

3.3 Machine Learning for Real–Bogus (RB) Classification

Raw image subtraction produces many false positives (artifacts, cosmic rays, CCD defects). To address this, ML-based RB classifiers evaluate image cutouts (stamps) and assign probabilities of a candidate being real.

Early RB efforts used **random forests** with morphological and photometric features. With deep learning's rise, **Leroy et al. (2019)** demonstrated that convolutional neural networks significantly outperform classical methods, achieving over 96% precision on ZTF datasets.

Further advancements by **Duev et al. (2019)** introduced **Bayesian neural networks**, adding uncertainty estimation that helps distinguish ambiguous cases—critical for follow-up scheduling.

These works establish CNN-based RB classifiers as a crucial step in any modern transient-detection pipeline.

3.4 Anomaly and Outlier Detection

While supervised RB classifiers detect known transient classes, discovering *new* or *rare* astronomical phenomena requires unsupervised approaches. Autoencoders, isolation forests, and hybrid models have shown effectiveness in anomaly detection across astrophysics and sensor systems.

Autoencoders learn a low-dimensional representation of “normal” astronomical sources; unusual sources yield higher reconstruction errors. Isolation forests detect anomalies based on feature-space sparsity. When combined, they enable the system to flag previously unknown transient types for human analysis.

Such methods are essential for modern astronomy, where unexpected events—supernova subtypes, kilonovae, tidal disruption events—carry significant scientific value.

3.5 Summary of Literature Gaps Addressed by This Project

Across the literature, challenges remain:

- Broker systems assume large infrastructure; few compact pipelines exist for academic/research prototyping.
- RB classifiers work well but rarely integrate anomaly detection into a unified workflow.
- Many studies focus on operational pipelines but lack open, modular implementations suitable for learning environments.

This project's contribution bridges these gaps by integrating optimal subtraction (ZOGY), CNN-based RB classification, and anomaly detection into a single, reproducible, educational pipeline.

4. Methodology / Planning of Work

Methodology / Planning of Work (should not exceed 1 page)

Planned Steps and Timeline

1. **Data Acquisition & Exploration (Weeks 1–2):** Download public ZTF FITS images and alert catalogs; build small labeled dataset (real transients vs bogus).
2. **Preprocessing & Calibration (Weeks 3–4):** Implement FITS I/O, WCS handling (Astropy), photometric zero-point estimation, bad-pixel masking, cosmic-ray rejection.
3. **Image Subtraction (Weeks 5–7):** Integrate/wrap ZOGY implementation; validate difference images on test fields.
4. **Source Extraction & Feature Engineering (Week 8):** Use SEP/Photutils to detect candidates and compute morphological/photometric/astrometric features.
5. **ML Model Development (Weeks 9–12):** Train stamp-level CNN for RB classification (augmentations, class imbalance handling); train autoencoder and isolation forest for anomaly scoring.
6. **Integration & Dashboard (Weeks 13–15):** Combine pipeline stages; build Streamlit dashboard for candidate visualization and export.
7. **Testing & Evaluation (Weeks 16–18):** Evaluate precision/recall, bogus rejection rate, processing latency; run injected-source experiments for anomaly sensitivity.
8. **Documentation & Packaging (Weeks 19–20):** Prepare README, installation guide, Dockerfile, and Jupyter notebooks.

Milestones

- Prototype pipeline ingesting FITS → ranked alert JSON (Week 12)
- RB classifier with validation precision >95% (Week 14)
- Dashboard + packaging (Week 20)

5. Facilities Required for Proposed Work

Software (minimum versions suggested):

- Python 3.10+
- Astropy 5.1+
- Photutils 1.5+
- SEP (Source Extraction Package)
- NumPy, SciPy, Pandas
- PyTorch 2.0+ (or TensorFlow/Keras)
- scikit-learn 1.3+
- Streamlit
- Docker

Hardware:

- Development machine: 16–32 GB RAM, 4+ CPU cores
- GPU (optional but highly recommended) for CNN training: NVIDIA GPU with CUDA (e.g., GTX 1080Ti / RTX series) or cloud GPU (A100/RTX) for faster training
- Disk: 1TB (for storing FITS cutouts and intermediate products)

6. References (IEEE format)

- [1] B. Zackay, E. O. Ofek, and A. Gal-Yam, "Proper Image Subtraction – Optimal Transient Detection, Astrometry, and Photometry for Difference Imaging," *Publ. Astron. Soc. Pac.*, vol. 128, no. 970, p. 12001, 2016.
- [2] E. C. Bellm *et al.*, "The Zwicky Transient Facility: System Overview, Performance, and First Results," *Publ. Astron. Soc. Pac.*, vol. 131, p. 018002, 2019.
- [3] T. Leroy *et al.*, "Real-bogus classification for the Zwicky Transient Facility using deep learning," *Mon. Not. R. Astron. Soc.*, vol. 490, pp. 4575–4591, 2019.
- [4] D. A. Duev *et al.*, "Real-bogus classification for the ZTF using Bayesian neural networks," *Mon. Not. R. Astron. Soc.*, vol. 489, pp. 3582–3590, 2019.
- [5] F. Förster *et al.*, "ALeRCE: The Automatic Learning for the Rapid Classification of Events Broker," *Astron. J.*, vol. 161, p. 242, 2021.
- [6] K. W. Smith *et al.*, "fink, a new generation of broker for the LSST community," *Astron. Astrophys.*, vol. 651, p. A103, 2020.
- [7] R. Stein *et al.*, "Lasair: Real-time Alert Filtering, Broking and Archive of Transients," *J. Astron. Instrum.*, vol. 10, p. 2050014, 2021.
- [8] W. D. Pence *et al.*, "Definition of the Flexible Image Transport Format (FITS)," *Astron. Astrophys.*, vol. 524, p. A42, 2010.
- [9] D. M. Bramich, "A new algorithm for difference image analysis," *Mon. Not. R. Astron. Soc.*, vol. 386, pp. 887–901, 2008.
- [10] Astropy Collaboration, "The Astropy Project: Building an Open-science Project and Status of the v2.0 Core Package," *Astron. J.*, 2018.