Draft version March 4, 2026
Typeset using LaTeX **twocolumn** style in AASTeX7.0.1

A Hybrid Ensemble and Deep Learning Framework for Detecting High-Redshift Lyα Blobs in Broadband Surveys

Daniel Godines [1] and Moire K. M. Prescott [1]

[1] *Department of Astronomy, New Mexico State University, PO Box 30001, MSC 4500, Las Cruces, NM 88003-8001, USA*

ABSTRACT

Lyman-alpha Blobs (LABs) are extended, luminous Lyα sources found at high redshifts and are thought to trace sites of active galaxy formation in dense environments. Their detection has traditionally relied on narrowband imaging targeting specific redshifts, limiting the scope of discovery to small comoving volumes and redshift windows. We aim to develop and validate a broadband-based classification framework for identifying LAB candidates in multi-band wide-field imaging surveys, enabling LAB selection over broader sky areas and redshift ranges. We present a multi-stage machine learning pipeline that combines a tree-based ensemble method, anomaly detection, and a shallow convolutional neural network. Morphological cuts are applied to prioritize final targets for spectroscopic follow-up. In this pilot study, the pipeline reduces $\sim$ 2 million cataloged sources in the Boötes field to 110 high-priority LAB candidates, successfully recovering 60% of the confirmed LABs used as benchmarks. By training on a larger sample of broadband-selected LAB candidates and a balanced set of non-LAB sources, our machine learning models successfully identify six of the nine known LABs in the field as high-probability candidates, including five broadband-selected and one narrowband-selected source. This work demonstrates that effective LAB candidate detection is feasible using real-world broadband data and minimal positive training samples. Our method thereby enables scalable blind searches for LABs across wide-field broadband surveys. All code and data products including our complete catalog have been released as part of an open-source Python package, Blob Identification Algorithm (`pyBIA`), providing the community with a complete framework for replication, extension, and spectroscopic follow-up planning.

Keywords: High-redshift galaxies (734) — Lyman-alpha galaxies (978) — Classification (1907) — Neural Networks (1933) — Outlier Detection (1934) — Random Forests (1935)

1. INTRODUCTION

Extended regions of diffuse Lyα emission, known as Lyα blobs (LABs), span hundreds of kiloparsecs and are typically found in overdense environments (C. C. Steidel et al. 2000; A. Dey et al. 2005; J. E. Geach et al. 2005; N. K. Wells et al. 2022). These large-scale structures can host numerous galaxies and are thought to trace sites of active galaxy formation and hierarchical assembly, potentially serving as signposts of proto-cluster environments (J. W. Colbert et al. 2006; S. Cantalupo 2017; B. Kimock et al. 2021).

The mechanisms driving the luminous Lyα emission in LABs remain poorly understood (R. Cen & Z. Zheng 2013; T. Chen & C. Guestrin 2016). Proposed explanations for their high Lyα luminosities ($L_{\mathrm{Ly}\alpha} \sim 10^{42} - 10^{44}$ erg s^{-1}) include a range of physical processes. Photoionization by ultraviolet photons from embedded star-forming galaxies or active galactic nuclei (AGN) remains a leading candidate (C. C. Steidel et al. 2000; P. Palunas et al. 2004; Z. Zheng et al. 2011; R. A. Overzier et al. 2013; Z. Cai et al. 2017). Additional mechanisms include galactic superwinds driven by intense star formation and supernova feedback, which can shock-heat the surrounding circumgalactic medium (CGM) and produce Lyα emission through subsequent hydrogen cooling (Y. Taniguchi & Y. Shioya 2000; R. J. Wilman et al. 2005). Alternatively, theoretical models suggest that gravitational cooling radiation, resulting from the release of potential energy as cold hydrogen gas accretes onto dark matter halos, can also account for the observed luminosities (Z. Haiman et al. 2000). This scenario has been supported by both hydrodynamical simulations (T. Goerdt et al. 2010; J. Rosdahl & J. Blaizot 2012; E. Daddi et al. 2021) and observational studies (K. K. Nilsson et al. 2006; D. J. B. Smith & M. J. Jarvis 2007). However, such cases remain observationally ambiguous, with one of the best initial cooling radiation candidates later reclassified as an AGN-powered system (M. K. M. Prescott et al. 2015; K. N. Sanderson et al. 2021).

Given the complexity of these environments, it is likely that multiple mechanisms contribute to the Lyα emission, either simultaneously or at different evolutionary stages. Interpretation is further complicated by the extended morphologies of LABs, which can also arise from the resonant scattering nature of the Lyα line (M. Dijkstra 2014). Lyα photons

Corresponding author: Daniel Godines
Email: godines@nmsu.edu

undergo multiple scatterings within neutral hydrogen in the CGM, enhancing the apparent spatial extent of the emission and making it more difficult to disentangle the underlying power sources (J. E. Geach et al. 2005; C. C. Steidel et al. 2011; R. Cen & Z. Zheng 2013).

Among the major challenges in studying these objects is their rarity which requires wide-area coverage for meaningful samples. Their low number densities make it difficult to draw statistically significant conclusions about their physical nature and environments (Y. Matsuda et al. 2004; S. Cantalupo 2017; F. Arrigoni Battaia et al. 2018). Systematic searches have traditionally relied on narrowband (NB) imaging techniques (C. C. Steidel et al. 2000; Y. Matsuda et al. 2004; Y. Yang et al. 2009; I. Iwata et al. 2019), which isolate Lyα emission at specific redshifts by targeting narrow spectral windows. These NB surveys have been successful in both blind searches (Y. Yang et al. 2009, 2010; V. Ramakrishnan et al. 2023) and in targeted observations of known overdensities or quasars (P. J. Francis et al. 1996; C. C. Steidel et al. 2000; Y. Matsuda et al. 2004; F. Arrigoni Battaia et al. 2018). While effective, the narrowband approach is observationally expensive and restricted in redshift coverage, typically probing $\Delta z \sim 0.05$ per filter (e.g., K.-S. Lee et al. 2024).

Broadband imaging provides a complementary approach to narrowband surveys. Unlike NB techniques, broadband searches are well-suited for blind, wide-field surveys and can efficiently probe large comoving volumes at significantly lower observational cost, using existing datasets (M. K. M. Prescott 2009; M. K. M. Prescott et al. 2012). At redshifts $z \sim 2-3$, the Lyα line is redshifted into blue filters, making it accessible to ground-based imaging surveys. However, as Lyα contributes only a small fraction of the total broadband flux, broadband-based searches are less sensitive to low equivalent width and low surface brightness emission than narrowband surveys. In addition, the broadband flux includes continuum emission including that from unrelated foreground/background sources, which dilutes the apparent Lyα morphology.

Early success with this method includes the discovery of an LAB in blue broadband (B_W) imaging via visual inspection by A. Dey et al. (2005). A more systematic effort by M. K. M. Prescott et al. (2012) applied a broadband color and morphology-based algorithm to identify LAB candidates, resulting in the discovery of four new LABs. More recently, R. Shimakawa (2022) used broadband imaging to detect extended Lyα emission around quasars at $z \sim 2-3$. While effective, these approaches rely on hand-tuned selection criteria and/or visual inspection, which become increasingly difficult to reproduce as survey volumes grow.

In this work, we develop a machine learning classification pipeline for automated LAB candidate detection in broadband imaging. We adopt a machine learning approach to enable scalable, objective selection in wide-field datasets from photometric and morphological information, rather than relying on survey-specific selection criteria. We also note that while conventional source-detection pipelines such as Source Extractor (`SExtractor`; E. Bertin & S. Arnouts 1996) provide effective initial detection and segmentation, they are often optimized for compact sources and rely on tunable choices (e.g., detection thresholds) that can strongly bias the recovery of low surface brightness diffuse emission, particularly in the crowded environments representative of LABs. We train on the sample of high-confidence LAB candidates identified by M. K. M. Prescott et al. (2012), which remains the largest publicly available broadband-selected LAB sample. Building on this training set, we introduce the Blob Identification Algorithm (`pyBIA`, D. Godines 2025), a scalable framework for identifying LABs in current and future wide-field imaging surveys, including the Legacy Survey of Space and Time (LSST) at the Vera C. Rubin Observatory (Ž. Ivezić et al. 2019). Although we apply `pyBIA` to deep imaging of the Boötes field, the methodology provides a template for separating rare, diffuse, low-surface-brightness extended sources from broadband survey data and is readily adaptable to other domains.

The structure of this paper is as follows: Section 2 describes the imaging data and candidate selection process, along with the machine learning methods and their performance across different source properties. Section 3 presents our candidate recovery results. In Section 4, we discuss the implications of our findings, and Section 5 summarizes our conclusions and outlines future prospects, including the public release of our classification code. Throughout the paper, all magnitudes are reported in the Vega system, and the intensity scaling of each presented broadband image is adjusted independently to optimize the visibility of source morphology.

2. METHODS

2.1. *Overview*

Our classification algorithm builds on the work of M. K. M. Prescott et al. (2012), who used optical B_W and R broadband imaging from the 9.4 deg^2 Boötes field of the NOAO Deep Wide-Field Survey (NDWFS; B. T. Jannuzi & A. Dey 1999) to search for high-redshift LABs. Their multi-step candidate selection process began by removing bright and compact sources (e.g., stars and galaxies) using `SExtractor` with varying detection thresholds. This was followed by unsharp masking and wavelet decomposition to isolate spatially extended sources. This stage relied exclusively on B_W imaging, motivated by the fact that redshifted Lyα emission can produce a detectable flux excess against the dark sky in deep blue broadband images.

This approach yielded $\approx 2,300$ sources, of which 866 were identified as LAB candidates by visual inspection. Among these, 407 exhibited diffuse, spatially extended morphologies, while the remaining 459 were associated with compact galaxy groupings but were retained due to the known association of LABs with overdense regions (B. P. Venemans et al. 2007; M. K. M. Prescott et al. 2008; N. K. Wells et al. 2022). The authors then incorporated color information and selected a final subset of 85 high-priority candidates using area and $B_W - R$ color cuts based on LABd05, the only confirmed LAB in the NDWFS footprint at the time (A. Dey et al. 2005). Spectroscopic follow-up of roughly one-third of this high-priority subset yielded four new LABs, resulting in a success rate of $\sim 25\%$ and a total of five confirmed LABs in the sample.

Using the broadband-selected LAB catalog from M. K. M. Prescott et al. (2012) as labeled training data, we develop a multi-stage machine learning pipeline to identify similar objects in wide-field imaging surveys. The pipeline explicitly balances recall — the ability to recover true LABs — with a low false-positive rate to yield a credible candidate list suitable for spectroscopic follow-up. It begins with a tree-based ensemble classifier trained on hand-engineered features that include discrete photometric values and segmentation-derived morphological properties extracted from the B_W imaging, such as shape, apparent magnitude, and surface brightness metrics that characterize the extended and diffuse nature of LABs. Next, we incorporate broadband color information (B_W and R-band imaging) and apply an Isolation Forest (`iForest`; F. T. Liu et al. 2008) to detect and remove outliers caused by artifacts such as bright halos, bleed trails from nearby stars, and regions of missing data near image edges. We evaluate several feature sets for the `iForest` step and find that Histogram of Oriented Gradients (HOG; N. Dalal & B. Triggs 2005) features, which encode edge orientations and textures across both bands, yield the best performance. In the final stage, we train a convolutional neural network (CNN) using the B_W and R-band images to further refine the candidate list. The CNN is trained on sources retained from earlier stages, enabling it to learn subtle morphological and color patterns that are difficult to capture using hand-engineered features alone.

Together, these machine learning models filter out $\sim 99.5\%$ of all cataloged sources. Final candidates are prioritized based on their morphological and color similarity to the five spectroscopically confirmed LABs, yielding a high-priority list of 110 LAB candidates and an overall filtering rate of $\sim 99.995\%$.

2.2. *Training Data*

To identify LAB candidates, we frame the task as a binary classification problem, treating LABs as the positive class and all other sources as the negative class. We begin by adopting the 866 LAB candidates identified by M. K. M. Prescott et al. (2012) as the positive training set, hereafter referred to as the LAB class. To ensure robust feature extraction, we visually inspected the B_W images for all 866 candidates and removed six sources affected by bright-star artifacts (halos and bleed trails) or incomplete coverage near the image boundaries. This yields a final LAB class of 860 "instances", which in the context of machine learning refers to the single data points or examples to be used during training or evaluation.

It is important to clarify that our training set does not rely solely on spectroscopically confirmed LABs. As the supervised models are trained on LAB candidates, we emphasize that the classifiers learn an empirical selection function for candidate-like sources among the entire survey population, rather than an astrophysically complete LAB definition. While the training set consists of 860 broadband-selected LAB candidates presented by M. K. M. Prescott et al. (2012), the benchmark set consists of the five sources within this sample that were spectroscopically confirmed in M. K. M. Prescott et al. (2013). By training on the larger candidate pool, we allow the models to learn generalizable morphological features from a statistically robust sample size. We utilize the five confirmed LABs to validate model recall and to optimize decision thresholds. While the true nature of the candidates in this larger set remains to be spectroscopically confirmed, their selection was based on objective imaging criteria and refined through visual inspection. We therefore consider their B_W morphologies to be representative of the high-redshift LAB population as currently understood.

The negative class, hereafter referred to as the OTHER class, is constructed by randomly sampling 860 sources from the full NDWFS Boötes catalog. This public NDWFS catalog was generated with `SExtractor` (v2.3.2) and is used here only to provide source IDs and centroid coordinates for extracting the imaging cutouts. We draw the OTHER sample uniformly at random to represent the underlying survey population of detected sources, avoiding additional cuts such as magnitude or morphology matching to ensure the negative class reflects the full range of morphologies and fluxes present in the survey catalog. We match the OTHER class sample size to that of the LAB class to maintain class balance and avoid introducing bias toward the majority class (M. Kuhn et al. 2013). Importantly, to avoid introducing a selection bias, we do not visually vet the negative class. For example, manually excluding only "obvious" non-LAB sources could lead the model to learn overly simplistic distinctions. This randomized sampling ensures the model is exposed to a wide range of sources, including extended, compact, and ambiguous morphologies. While this strategy carries a small risk of inadvertently including true LABs in the

negative class, this risk is negligible given the extremely low comoving number density of LABs ($\sim 10^{-6}\ h_{70}^3\ \mathrm{Mpc}^{-3}$ at $z \sim 2.3{-}6.6$) (P. Palunas et al. 2004; K. K. Nilsson et al. 2006; T. Saito et al. 2006; Y. Matsuda et al. 2004, 2006; M. K. M. Prescott et al. 2008; M. Ouchi et al. 2009; Y. Matsuda et al. 2009; Y. Yang et al. 2009, 2010; Y. Matsuda et al. 2011; D. Erb et al. 2011; M. K. M. Prescott et al. 2013). Nonetheless, in Section 2.3.5, we perform a cross-validation analysis to identify any promising LAB candidates within the negative class.

The resulting training set consists of 1,720 instances, evenly divided between the LAB and OTHER classes, and serves as the foundation for training and evaluating a range of supervised classification models.

2.3. *Single-band Classification*

For the initial classification stage, we evaluate a diverse set of supervised learning models that span a range of complexity and interpretability. These include linear models such as logistic regression (`LogReg`); non-linear, kernel-based models such as support vector classification (`SVC`); and flexible function approximators such as multilayer perceptrons (`MLP`). We also test individual decision trees (`Tree`) and tree-based ensemble methods. This progression from simple to more flexible models follows best practices in astronomical machine learning, beginning with simpler models before progressing to more flexible but less interpretible ones (D. Huppenkothen et al. 2023).

Each model is trained and evaluated using the same feature set and cross-validation procedure to ensure consistency. Simpler models such as `LogReg` and `Tree` offer interpretability and fast training but can struggle with non-linear decision boundaries and high-dimensional data (T. Hastie et al. 2009; A. Herrera-Martin et al. 2025), whereas kernel methods like `SVC` can model more complex boundaries but require careful tuning and do not scale well to large datasets (J. Cervantes et al. 2008; R. Fadely et al. 2012; O. A. Montesinos López et al. 2022). Neural networks offer strong representational power but are sensitive to weight initialization, regularization strategies, and the size of the training set (X. Glorot & Y. Bengio 2010; N. Srivastava et al. 2014; P. Nakkiran et al. 2019).

In contrast, ensemble tree methods are robust to irrelevant or missing features, require little preprocessing, and scale well to imbalanced or noisy datasets (R. Maclin & D. Opitz 2011). These advantages have led to their widespread adoption in astronomical classification problems, including gravitational microlensing (Ł. Wyrzykowski et al. 2015, 2016; D. Godines et al. 2019; M. Crispim Romao & D. Croon 2024; A. Schweitzer et al. 2025; M. Crispim Romao et al. 2025), exoplanet detection (Y. Dzigan & S. Zucker 2012; S. Saha et al. 2018), and variable star classification (J. W. Richards et al. 2011; I. N. Pashchenko et al. 2018; S. B. Soltau & L. C. Botti 2021). We focus on two particular algorithms, the Random Forest (`RF`; L. Breiman 2001) and eXtreme Gradient Boosting (`XGBoost`; T. Chen & C. Guestrin 2016), both based on the Classification and Regression Trees (CART; L. Breiman et al. 1984) algorithm. `RF` builds an ensemble of decision trees in parallel using bootstrap aggregation ("bagging"; L. Breiman 1996), where each tree is trained on a random resampled subset of the data and splits on a random subset of features, thereby decorrelating trees and reducing variance. In contrast, `XGBoost` follows a sequential "boosting" framework (J. H. Friedman 2001; T. Chen & C. Guestrin 2016), where each new tree is trained to correct the residual errors made by the previous ensemble. This process can be interpreted as gradient descent in a function space, with each weak learner (i.e., the individual decision trees) approximating the negative gradient of the loss function relative to the current model output, incrementally improving model performance but with greater risk of overfitting if not properly regularized (e.g., by limiting tree depth and/or number). A key advantage of tree-based ensembles is their ability to output calibrated class probabilities. For a given object, the predicted probability of belonging to a class y is computed as the fraction of trees voting for y, normalized by the total number of trees. Henceforth, we denote the probability of an object being classified as an LAB as $P(y = \mathrm{LAB} \mid \mathbf{X})$, where $\mathbf{X}$ is the vector of input features. To construct $\mathbf{X}$, we first perform image segmentation on the B_W imaging to identify the pixel regions associated with each source.

2.3.1. *Image Segmentation*

The NDWFS data release[2] provides merged catalogs for 27 subfields in the Boötes field, containing $\approx$ 2.4 million detected sources. For each object, we extract its right ascension (α), declination (δ), subfield name, and unique catalog ID. Although the NDWFS catalogs include morphological parameters derived using `SExtractor`, we compute our own catalog to better control source detection sensitivity and extract segmentation-based features using the `Photutils` Python library (L. Bradley et al. 2024).

Image segmentation is a pattern recognition technique that partitions an image into distinct regions, isolating sources for morphological analysis (see N. Pal & S. Pal 1993 and references therein). To minimize contamination from nearby sources while preserving extended emission, we extract 100× 100 pixel (26×26 arcsec) cutouts centered on each cataloged object. This reduces artifacts from bright star halos that can otherwise produce excessively large segmentation patches. We disable source deblending to preserve large-scale, con-

[2] https://astroarchive.noirlab.edu/

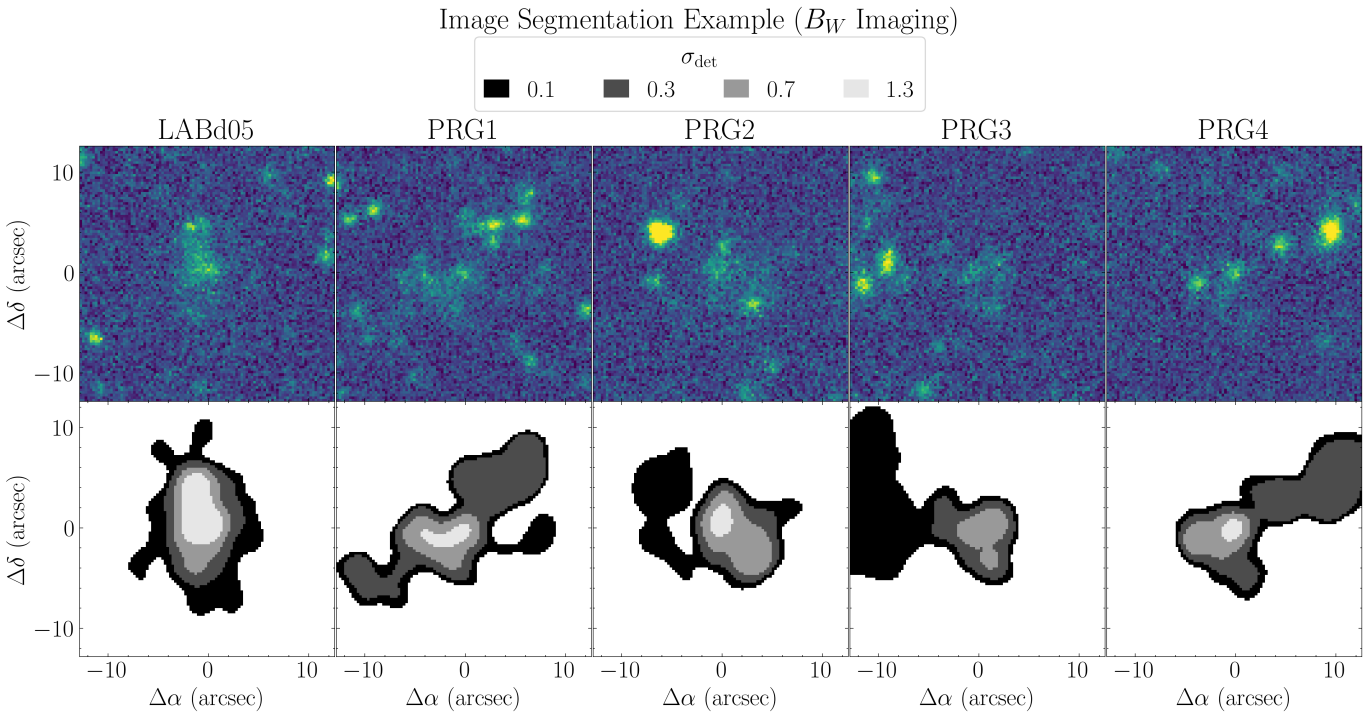

Figure 1. Image segmentation. The upper panels show the B_W imaging of the five confirmed LABs in our training set. The panels below display segmentation masks generated at varying source detection thresholds, showing how the source extent changes with increasing $\sigma_{\rm det}$.

nected structures characteristic of LABs and their group environments.

As the NDWFS imaging of the Boötes field is not background-subtracted, prior to segmentation, we use `Photutils` to estimate the local sky background for each source using a circular annulus with inner and outer radii of 20 and 35 pixels (5.1 and 7.7 arcsec), respectively. The 3σ-clipped median value within this annulus is subtracted from the cutout to ensure that the detection threshold refers to flux above the local sky. We note that while this local subtraction method may result in over-subtraction for the most extended LAB candidates if their diffuse wings extend into the background annulus, the largest LAB in our confirmed sample (LABd05) is within these bounds and was used to guide the size of these annuli. Segmentation is performed by convolving each background-subtracted cutout with a 2D Gaussian kernel with a full width at half maximum (FWHM) of 9 pixels (2.3 arcsec) and a kernel size of 21 × 21 pixels (5.4 × 5.4 arcsec). A detection requires at least 9 connected pixels above the background, where pixels are considered connected if they share an edge or corner. The detection threshold, $\sigma_{\rm det}$, defined as the number of standard deviations above the background, is a key tunable parameter that controls the spatial extent of the segmentation mask. High values of $\sigma_{\rm det}$ restrict detections to compact, bright cores, while lower values allow inclusion of diffuse, low-surface-brightness emission.

We perform segmentation on each image cutout and associate the nearest segmentation patch with the NDWFS catalog position. If no region lies within 10 pixels (2.6 arcsec) of the cataloged coordinates, the source is flagged as a "non-detection". This is an important design choice during catalog generation, as due to the resonant nature of Lyα scattering, the emission peak can be spatially offset from the ionizing source (T. Shibuya et al. 2014; A. Hoag et al. 2019). This behavior is supported by the strong linear polarization of Lyα photons in regions that are offset from photoionizing sources (M. Hayes et al. 2011; M. Beck et al. 2016; M. B. Eide et al. 2018; E. Kim et al. 2020; E. C. Herenz et al. 2020). Therefore, selecting the nearest region allows us to recover prominent emission that may be offset with respect to the cataloged centroid. While associating the nearest segmentation patch can include unrelated emission (particularly at low $\sigma_{\rm det}$), this choice is designed to prioritize recovery of offset Lyα emission near the target position rather than enforce a perfectly isolated segmentation. By employing a low $\sigma_{\rm det}$ and disabling deblending, we increase the likelihood that extended Lyα structure remains connected to the selected region, even if contaminants are included. We catalog the features of a "non-detection" with numerical flags of -999, which invalidates the feature matrix and excludes the instance from training. Because we enforce class balance in training, any OTHER-class non-detections are replaced by additional random draws from the NDWFS catalog until the number of valid OTHER instances matches the number of

valid LAB instances. As such, only LAB non-detections reduce the effective training set size for a given $\sigma_{\rm det}$ threshold, as the OTHER class is always downsampled to match.

Fig. 1 demonstrates the segmentation process for the five spectroscopically confirmed LABs in our training set, highlighting the effect of $\sigma_{\rm det}$ in balancing source recovery and contamination. Besides PRG3, the other four LABs in our training set are successfully segmented in all cases, due to the presence of bright knots near their cataloged positions. At $\sigma_{\rm det} = 1.3$, PRG3 fails to yield valid segmentation regions and is thus flagged as a non-detection. At very low thresholds (e.g., $\sigma_{\rm det} = 0.1$), the segmentation masks encompass more extended emission, but also begin to include unrelated sources in all cases but LABd05. As the largest and brightest LAB in the sample, LABd05 is consistently recovered with a large, clean segmentation mask across all thresholds due to its relatively uniform flux distribution and lack of nearby contaminants. In the analysis that follows, we optimize the value of $\sigma_{\rm det}$ as part of the training process (Section 2.3.3) to balance recovery of extended structure against contamination from neighboring sources.

2.3.2. *Training Features*

For each source we generate a suite of 66 measurable features. The complete list of all source features used for model training is summarized in Table 1. Among the most important features derived from the segmentation patches is the Gini index, originally introduced by C. Gini (1912) to quantify inequality in income distributions. In astronomy, the Gini index has been widely adopted to characterize galaxy morphologies (R. G. Abraham et al. 2003), to identify compact AGN (C. M. Casey et al. 2008), and to analyze lensed galaxy structure in the image plane (M. K. Florian et al. 2016). For a given image segmentation patch, the Gini index quantifies how uniformly the flux is distributed across pixels and is computed following J. M. Lotz et al. (2004),

$$\text{Gini} = \frac{1}{\bar{X} n(n-1)} \sum_{i}^{n} (2i - n - 1) X_i, \tag{1}$$

where X_i are the pixel values sorted in ascending order, $\bar{X}$ is the mean pixel value, and n is the number of pixels in the segmentation mask. A value near zero indicates uniform brightness, while values approaching unity correspond to extreme flux concentration within a small subset of pixels. The Gini index is particularly useful due to its robustness to outliers and confinement to the unit interval.

We compute raw image moments of the form

$$M_{\rm nm} = \sum_{x} \sum_{y} x^n y^m I(x, y), \tag{2}$$

for all (n, m) such that $n + m \leq 3$ (i.e., to third order), where $I(x, y)$ is the pixel intensity at position (x, y). These moments capture the flux spatial distribution. We similarly derive central moments (μ_{nm}) by re-centering the coordinates around the intensity-weighted centroid, making them invariant to translation. The central moments μ_{01} and μ_{10}, corresponding to the first-order intensity-weighted averages in x and y, are excluded because they are zero by definition in the centroid-centered frame. We also compute geometrically centered moments (G_{nm}), which use the geometric center of the image cutout as the origin. These features are sensitive to asymmetries and positional offsets, enabling the model to distinguish both intrinsic and relative morphologies. All moments are calculated to third order, although we omit μ_{00} and G_{00}, as they are equivalent to M_{00}.

To encode higher-order structural information, we compute the seven Hu invariants (M.-K. Hu 1962) using the classical η-normalization of the central moments, where

$$\eta_{nm} = \frac{\mu_{nm}}{\mu_{00}^{\left(1 + \frac{n+m}{2}\right)}}. \tag{3}$$

This normalization makes these moments invariant to scale, rotation, and translation. Because our image data are normalized by exposure time (counts s^{-1}), sources with faint, diffuse emission have small total flux μ_{00}. The η-normalization exponent can strongly amplify such cases, producing Hu values on the order of $10^{17} - 10^{18}$ for higher-order terms. We therefore log-scale the Hu invariants to stabilize their range for model training.

To third order, we also calculate Legendre moments (L_{mn}; M. R. Teague 1980), which project the 2D intensity distribution onto orthonormal Legendre polynomials, yielding coefficients sensitive to large-scale shape characteristics (R. J. Prokop & A. P. Reeves 1992). Additional shape parameters are extracted using `Photutils`, including ellipticity, eccentricity, and orientation, obtained by fitting a multivariate Gaussian to the segmentation region. We also include photometric features, such as apparent magnitude and its associated uncertainty. These are measured independently of the segmentation patch via circular aperture photometry, using a large 15-pixel ($\sim$ 4 arcsec) radius to ensure that the largest known LAB in our spectroscopically-confirmed sample (LABd05) is fully enclosed. The local sky background is estimated using the same annulus parameters described in Section 2.3.1.

These morphological and photometric features collectively provide a high-dimensional description of each source, forming the input vector $\mathbf{X}$ for all classification models introduced at the beginning of Section 2.3.

2.3.3. *Baseline Classification Analysis*

Our first objective is to select the best performing classifier from the six tested models (`Tree`, `LogReg`, `SVC`, `MLP`, RF, and `XGBoost`; see Section 2.3) and determine the optimal

Feature Names	Description
Area	Area of segmentation patch, in pixels
C_{xx}, C_{yy}, C_{xy}	`SExtractor` ellipse parameters
Eccentricity, Ellipticity	Ellipsoidal attributes
Equiv. Radius	Equivalent radius
FWHM	Full Width at Half Maximum
Gini Index	Gini index of the flux distribution
G_{10}, ..., G_{03}	Geometrically centered moments
h_1, ..., h_7	Hu invariant moments
L_{00}, ..., L_{03}	Legendre Moments
Mag, MagErr	Large-aperture B_W magnitude and error
Max Val., Min Val.	Maximum and minimum pixel values
M_{00}, ..., M_{03}	Raw image moments
μ_{20}, ..., μ_{03}	Centroid-centered moments
$\sigma^2(x)$, $\sigma^2(y)$, $\sigma^2(xy)$	Spatial variance & covariance
$\sigma_{\rm major}$, $\sigma_{\rm minor}$	Std. dev. along major and minor axes
λ_1, λ_2	Eigenvalues of the covariance matrix

Table 1. Summary of the 66 image features used to train the supervised learning models in the initial single-band classification stage.

$\sigma_{\rm det}$ for image segmentation. All models are trained with default hyperparameters on the 66 features in Table 1 and are hereafter referred to as the "baseline" classifiers.

To assess the effect of $\sigma_{\rm det}$, we apply the segmentation procedure from Section 2.3.1 to all 1720 training set instances, varying the detection threshold over $0.1 \leq \sigma_{\rm det} \leq 1.5$. At each threshold, segmentation-based features are recomputed and classifiers retrained. Because the OTHER class is drawn randomly from the full NDWFS Boötes catalog, any non-detections in this class are replenished by resampling from the catalog; thus, the effective training set size is determined solely by the LAB non-detection rate at a given $\sigma_{\rm det}$. The top panel of Fig. 2 presents the detection rates for both classes, showing how at $\sigma_{\rm det} = 1.5$, roughly 15% of LABs and $\sim 70\%$ of OTHER sources are missed. The LAB class exhibits much lower non-detection rates overall, with virtually no losses at $\sigma_{\rm det} \lesssim 0.6$, reflecting the relative higher surface brightness and spatial extent of our LAB candidates compared to typical survey sources. While LABs are intrinsically diffuse, low surface brightness objects, the general NDWFS Boötes catalog is heavily dominated by faint sources near the limiting magnitude. As such, the broadband-selected LAB candidates from M. K. M. Prescott et al. (2012) are, on average, relatively brighter with higher surface brightnesses than the majority of sources in the general survey population. At $\sigma_{\rm det} = 1.5$, the training set is reduced by approximately 15% due to LAB non-detections, whereas for $\sigma_{\rm det} \lesssim 1.0$, the LAB non-detection rate remains below 2%.

Model performance is evaluated at each $\sigma_{\rm det}$ using 10-fold cross-validation (CV). In this approach, the training set is divided into 10 equal subsets, or "folds". Each fold is used once as a validation set while the remaining nine folds are used for training. This process is repeated 10 times, ensuring that every instance is used for validation exactly once. Cross-validation provides a reliable estimate of model generalization performance, balancing underfitting (high bias) and overfitting (high variance), making it a robust choice for performance evaluation (S. Arlot & A. Celisse 2009). We adopt the F1 score as our primary evaluation metric, averaged across all folds. This metric is calculated as

$$\mathrm{F1} = 2 \times \frac{\mathrm{Precision} \times \mathrm{Recall}}{\mathrm{Precision} + \mathrm{Recall}}, \tag{4}$$

where "Precision" is the fraction of predicted positives that are correct, and "Recall" is the fraction of actual positives that are recovered. This harmonic mean provides a balanced assessment of classification performance by equally weighting the impact of false positives and false negatives.

The middle panel of Fig. 2 presents the classification performance for all models, and shows that tree-based ensembles consistently outperform the other classifiers, with all models peaking at lower thresholds ($0.3 \lesssim \sigma_{\rm det} \lesssim 0.6$). `XGBoost` achieves the highest F1 score (0.9314) at $\sigma_{\rm det} = 0.38$, followed closely by `RF` (0.9257). The next-best performance is obtained by the `SVC` (~ 0.91), which benefits from its ability to capture non-linear decision boundaries in our high-dimensional, 66-feature space. The `MLP` and `LogReg` models follow closely at ~ 0.90, reflecting the ability of linear and shallow non-linear models to capture much of the class separation but with less flexibility than ensembles or kernel methods. The single decision tree performs the worst (~ 0.89), consistent with its higher variance and lack of boosting or bagging to mitigate overfitting.

Because the OTHER class represents the general survey population, minimizing its non-detection rate is important for completeness. Lower non-detection rates allow for more complete recovery of diffuse and faint emission that would otherwise be missed at moderate thresholds. As shown in Fig. 2, the peak F1 score of `XGBoost` is comparable to its value at slightly lower thresholds. For instance, at $\sigma_{\rm det} = 0.32$, `XGBoost` achieves an F1 score of 0.929, comparable to the score at $\sigma_{\rm det} = 0.38$. At the same time, the OTHER non-detection rate is reduced from 14% to just 0.10%, allowing us to retain 90% of all cataloged sources after segmentation for classification. We therefore adopt `XGBoost` at $\sigma_{\rm det} = 0.32$ as our optimal configuration.

We note that, given our limited training data, instead of assessing performance on a reserved hold-out test set, we adopt 10-fold cross-validation on the full training set as our primary estimate of generalization performance. As a robustness check on potential overfitting, we repeated the same

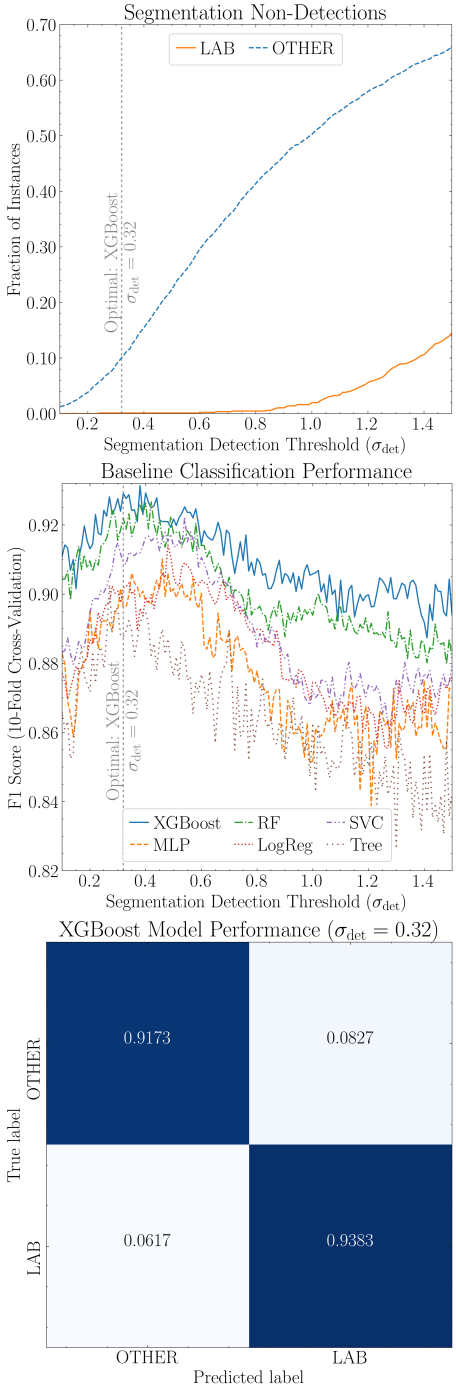

Figure 2. Classification performance. Top: Fraction of non-detections in the LAB and OTHER classes as a function of σ_{det}. Middle: Mean F1 score (10-fold CV) for all baseline classifiers tested. Bottom: Confusion matrix for the best-performing model (`XGBoost`) at the optimal σ_{det}.

baseline analysis presented in the middle panel of Fig. 2 using a stratified hold-out split, reserving 30% of the training data as the test set. To ensure our results are not biased by any single random partition, we repeated this procedure 10 times using different random seeds for the train-test splitting. We find that, across these iterations, the performance exhibits the same behavior observed in Fig. 2, with `XGBoost` outperforming all models, and all models peaking in the $0.3 \lesssim \sigma_{det} \lesssim 0.4$ range. We therefore conclude that our adopted $\sigma_{det} = 0.32$ is not sensitive to the specific validation scheme and provides a stable configuration that also minimizes the non-detection rate.

As only `XGBoost` among our six baseline models can natively handle missing values, all training features are required to be finite. Our feature set is explicitly constructed to avoid undefined values, so we do not apply imputation techniques. In practice, the only invalid (NaN) values arise from log-scaled magnitudes, which occur when flux measurements are negative (e.g., due to oversubtracted backgrounds, extremely faint sources near the survey limit, or other imaging artifacts). At the optimal $\sigma_{det} = 0.32$, all LAB instances have valid magnitudes, but 7.2% of detected sources have invalid magnitudes. Among the remaining $\sim$ 1.97 million cataloged sources with valid magnitudes, 30 objects ($< 0.002\%$) have invalid higher-order Hu moments, likely due to η-normalization instabilities in very faint sources. No other features yield invalid values. Accounting for both non-detections and non-finite features, the catalog is reduced from $\sim$ 2.38 million to $\sim$ 1.97 million sources available for classification ($\sim$ 82.8%), with only one LAB instance removed, yielding a final training set of 1718 objects.

Our results demonstrate that boosting-based ensembles like `XGBoost` provide the strongest baseline performance among the six tested models, and lower detection thresholds yield segmentation masks that better capture the extended, diffuse emission typical of LABs. The lower panel of Fig. 2 presents the confusion matrix for the optimal baseline model, showing > 90% accuracy for both classes and slightly better performance on the LAB class.

2.3.4. *Feature Selection & Model Optimization*

Although ensemble learning algorithms are generally robust to noisy or correlated features (L. Breiman 2001), identifying the most informative features can improve model generalization, simplify interpretation, and reduce training costs (I. Guyon & A. Elisseeff 2003; E. Alpaydin 2020). To rank feature importance, we apply the Boruta SHapley Additive exPlanations (`BorutaSHAP`; E. Keany 2020) framework, which builds on the `Boruta` algorithm (M. B. Kursa & W. R. Rudnicki 2010) but uses SHapley Additive exPlanations (SHAP) values (E. Štrumbelj & I. Kononenko 2014; S. Lundberg & S.-I. Lee 2017) from cooperative game theory (L. Shapley 1953) to quantify each feature's contribution to the model output.

In `BorutaSHAP`, feature relevance is evaluated by comparing each real feature's SHAP-based importance score to

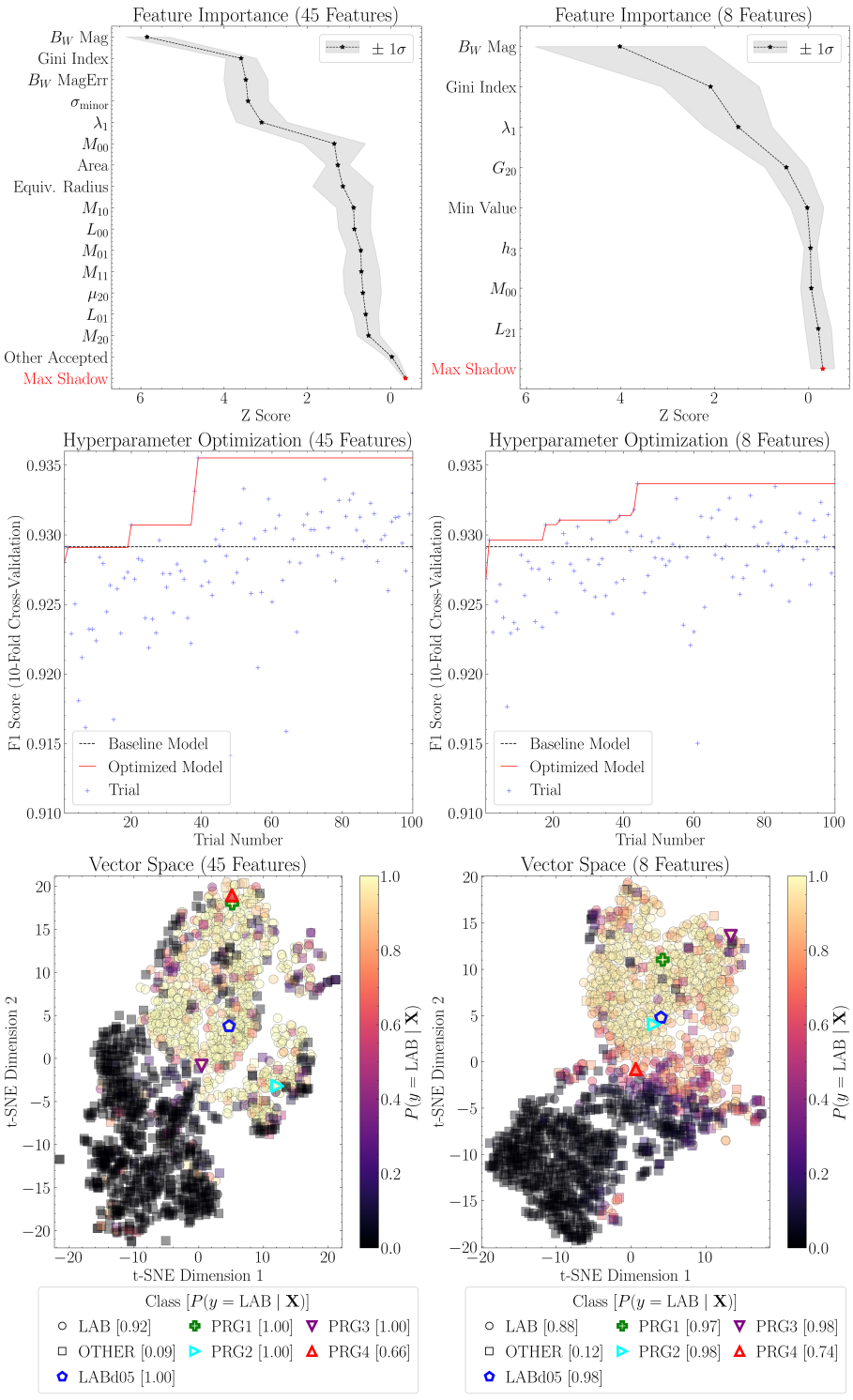

Figure 3. Feature selection and `XGBoost` model optimization. The left column shows results using `BorutaSHAP` with `RF`-based feature ranking, while the right column uses `BorutaSHAP` with `XGBoost`-based ranking. The top panels show the selected features ranked by mean Z-score across trials, with the "Max Shadow" acceptance threshold shown in red. The middle panels displays the results of `Optuna`-based hyperparameter optimization, with baseline model performance at the optimal σ_{det} from Fig. 2 shown for comparison. The lower panels present the 2D t-SNE projections of the reduced feature spaces, where each point is a training instance. LAB-class objects are shown as circles, OTHER-class objects as squares, and the five confirmed LABs are indicated with unique markers. Points are colored by the predicted LAB-class probability, as indicated by the colorbar, with the legend showing class-averaged probabilities for LAB and OTHER classes and individual values for the confirmed LABs.

those of "shadow features", which are copies of the original features that are randomly permuted across the dataset to remove any relationship to the training instance. The model is trained on both real and shadow features, and a feature is considered to score a "hit" in an iteration if its importance exceeds the maximum importance score observed among all shadow features in that iteration. Across iterations, right-tailed binomial tests (with Bonferroni correction) assess the null hypothesis that a feature performs no better than the shadows, leading to its classification as either accepted, rejected, or tentative. The process is repeated for a fixed number of trials, with feature classifications updated after each iteration. The comparison is quantified with a normalized Z-score,

$$Z(\mathbf{S}_i) = \frac{\mathbf{S}_i - \mu_{\text{shadow}}}{\sigma_{\text{shadow}}}, \tag{5}$$

where $\mathbf{S}_i$ is the SHAP-based importance score of feature i, and μ_{shadow} and σ_{shadow} are the mean and standard deviation of the shadow-feature importance distribution from that same iteration.

We apply `BorutaSHAP` using two different underlying models to compute SHAP values, the `RF` and `XGBoost` classifiers. The `RF` distributes importance more evenly across correlated features due to its independent-tree structure, whereas `XGBoost`, which trains trees sequentially, tends to concentrate importance on a smaller subset of features that exert strong early influence on loss reduction (S. M. Lundberg et al. 2018; M. Loecher & Q. Wu 2021). We run `BorutaSHAP` for 100 trials using both `RF` and `XGBoost`-based importance metrics to compute the SHAP values. The `RF`-based run identifies 45 important features, while the `XGBoost`-based run selects only eight. The important features, ranked by the mean Z-scores across all trials, are shown in the top row of Fig. 3, with shaded regions indicating the 1σ spread across trials. The left panel displays the RF-based results, listing the top 15 features and aggregating the remaining 30 into an "Other Accepted" category, while the right panel shows all eight accepted features from the `XGBoost`-based results. The maximum shadow Z-score ("Max Shadow") which defines the acceptance threshold, is shown in red, with features with Z-scores consistently above this value being retained.

In both runs, the most important feature is the B_W magnitude, with mean Z-scores roughly twice as large as the second most important feature. The high importance of the B_W magnitude reflects our classification task, as the model is trained to identify LAB candidates among the full NDWFS Boötes catalog, which is dominated by faint sources near the limiting magnitude of the survey and also includes bright stars. Consequently, the B_W magnitude emerges as a top feature as it is highly informative for rejecting these populations. The Gini index also ranks highly in both cases, followed by moment-based shape descriptors such as σ_{minor} and λ_1, which capture the spatial extent and flux variance along the principal axes of the segmentation region. This suggests that the models distinguish LAB candidates primarily through their apparent brightness as well as their surface brightness distribution, in addition to morphological characteristics that capture the extent of the source in B_W. Other features also contribute meaningfully, including the raw image moment M_{00}, several Legendre and Hu moments, and even the minimum pixel value, highlighting the model's use of complementary flux, shape, and local intensity information. For comparison, in Appendix A we quantify the effects of removing flux features, presenting the performance of an `XGBoost` model trained on morphological features alone.

To compare performance across feature sets, we define two `XGBoost` models: `XGBoost-45`, trained on the 45 features selected by the `RF`-based `BorutaSHAP` run, and `XGBoost-8`, trained on the 8 features selected by the `XGBoost`-based run. The goal of testing both configurations is to determine whether the broader, more redundant feature set offers any performance advantage over the compact set, or whether the smaller set provides superior discriminative power.

Both `XGBoost` models are optimized using Bayesian hyperparameter tuning with the `Optuna` API (T. Akiba et al. 2019), which uses a Tree-structured Parzen Estimator (TPE; S. Watanabe 2023) to guide the search. Bayesian optimization methods probabilistically model the objective function, typically a loss or error metric, to guide the search toward promising hyperparameter regions (J. Snoek et al. 2012). The optimization process iteratively reduces this error by narrowing the search space based on the performance of previously evaluated configurations. In ensemble learning, engine hyperparameters define the model's training configuration, such as the number of decision trees and their depth (i.e., the maximum number of splits per tree). These hyperparameters are set prior to training and are effectively the "settings" at which the model operates. Recent applications of `Optuna` in astronomy include successful cosmological modeling and differentiation (L. W. K. Goh et al. 2024; X. Xiao et al. 2024), lightcurve simulations of compact binary systems (M. Gull et al. 2024), as well as for the optimization of early warning gravitational wave alert systems (A. Tohuvavohu et al. 2024; S. Green & A. Lundgren 2024) and tree-based ensemble model tuning for microlensing detection (M. C. Romao et al. 2025; M. Crispim Romao et al. 2025).

We define the `Optuna` objective function as the 10-fold cross-validated F1 score. The range of hyperparameters explored during the optimization as well as the optimal values for both `XGBoost` models are listed in Table 2. The hyperparameter optimization was run for 100 trials using `Optuna`'s default optimizer. In each trial, the proposed hyperparameter combination is used to train the classifier, and its performance is evaluated. The configuration with the highest

Hyperparameter	Range	Optimal (XGBoost-8/45)
n_estimators	[100, 300]	285 / 156
max_depth	[3, 10]	4 / 6
learning_rate	[10^{-3}, 0.3] (log)	0.108 / 0.247
reg_lambda	[10^{-3}, 2.0] (log)	0.059 / 0.52
reg_alpha	[10^{-3}, 2.0] (log)	1.04 / 0.67
gamma	[0.0, 10.0]	5.56 / 0.018
subsample	[0.5, 1.0]	0.717 / 0.907

Table 2. XGBoost hyperparameter optimization with Optuna. The table presents the explored parameter ranges and optimal values chosen for both XGBoost-8 and XGBoost-45 models.

F1 score is then used to train the final engine. The optimal hyperparameters (Table 2) show that the 8-feature model favors more trees, shallower depth, a lower learning rate, and a substantially larger regularization parameter of gamma = 5.56. This configuration reflects slower but more conservative learning, where only strong, high-confidence splits are permitted in order to suppress overfitting in the limited feature space. In contrast, the 45-feature model converges on fewer, deeper trees with a higher learning rate and a minimal regularization threshold (gamma = 0.018), adopting a more flexible splitting strategy to capture the complexity of the larger feature set.

The second row of Fig. 3 shows the hyperparameter optimization results for XGBoost-45 (left) and XGBoost-8 (right), with the blue points representing the objective value (i.e., the cross-validated F1 score) of the individual trials, with a red line tracking the highest achieved score over time. For comparison, the dashed black line shows the performance of the optimal baseline XGBoost classifier from Fig. 2. Fig. 3 shows how most performance gains are achieved within 50 trials. Although all models achieve comparable performance, XGBoost-45 attained the highest F1 score of 0.936. The tuned XGBoost-8 model, despite using far fewer features, exceeded the performance of the default model trained on substantially more features. This demonstrates that hyperparameter optimization can offset the potential loss of predictive power from dimensionality reduction, enabling a more efficient yet competitive model.

As noted in Section 2.3.3, we adopt a 10-fold cross-validation strategy to make full use of our training data instead of reserving a dedicated hold-out test set. While this approach proved reliable when determining the optimal $\sigma_{\rm det}$, we additionally verified the full BorutaSHAP and Optuna procedure on XGBoost-8 with a stratified 70/30 split to guard against potential overfitting during the optimization steps. We performed feature selection and hyperparameter tuning exclusively on the 70% training partition and evaluated the model afterward on the held-out 30%. We find that the highest-ranked features remain the same, with the optimized model achieving an F1 score of 0.916 on the held-out test set. The resulting test-set metrics are thus consistent with the cross-validation estimates, demonstrating that the optimized model generalizes well and that our primary 10-fold CV performance scores were not artificially inflated by overfitting.

To further assess classification performance, we apply leave-one-out cross-validation (LOOCV) on the entire training set, where each instance is excluded once and treated as a test case. This approach yields low-bias probability estimates and simulates how each source would be classified in a blind search (S. Arlot & A. Celisse 2009). LOOCV is applied to both the LAB and OTHER training classes to evaluate model sensitivity to known LAB candidates and to identify LAB-like objects among the randomly selected negatives.

The lower row of Fig. 3 presents a 2D visualization of the feature space for XGBoost-45 (left) and XGBoost-8 (right). These projections are generated with the t-distributed Stochastic Neighbor Embedding (t-SNE) algorithm, an unsupervised method for nonlinear dimensionality reduction (L. van der Maaten & G. Hinton 2008; G. E. Hinton & S. Roweis 2002). This method maps high-dimensional feature vectors into a lower-dimensional space by converting pairwise distances into conditional probabilities, then matching these to a similar distribution in the low-dimensional space via Kullback-Leibler divergence minimization (L. van der Maaten & G. Hinton 2008; L. van der Maaten 2014). While t-SNE coordinates have no direct physical meaning, points that appear close in the projection correspond to instances with similar feature profiles, allowing patterns and clustering tendencies to be visually assessed.

The t-SNE projections in Fig. 3 reveal distinct LAB (circles) and OTHER (squares) clusters, with confirmed LABs highlighted using unique marker styles and colors. Points are colored by the LOOCV-derived probability of being an LAB, as shown by the colorbar. The legend reports mean predicted probabilities for each class and individual values for confirmed LABs. Intermediate probabilities (e.g., $P(y = \text{LAB} \mid \mathbf{X}) \approx 0.5$) tend to occur where the clusters overlap, marking the effective decision boundaries. All five confirmed LABs are correctly identified by both models with $P(y = \text{LAB} \mid \mathbf{X}) \gtrsim 0.97$, except for PRG4, which is assigned $P(y = \text{LAB} \mid \mathbf{X}) = 0.66$ and $P(y = \text{LAB} \mid \mathbf{X}) = 0.74$ by XGBoost-45 and XGBoost-8, respectively. This is consistent with its position near the cluster boundary in the 8D feature space and near the edge of the LAB cluster in the 45D space. Ambiguous predictions cluster along these boundaries, indicating regions of higher model uncertainty and reduced separability.

Furthermore, the average predicted probabilities shown in the t-SNE plot legends of Fig. 3 are consistent with overall model performance. The LAB class exhibits a mean probability of $P(y = \text{LAB} \mid \mathbf{X}) \approx 0.9$, while the OTHER class

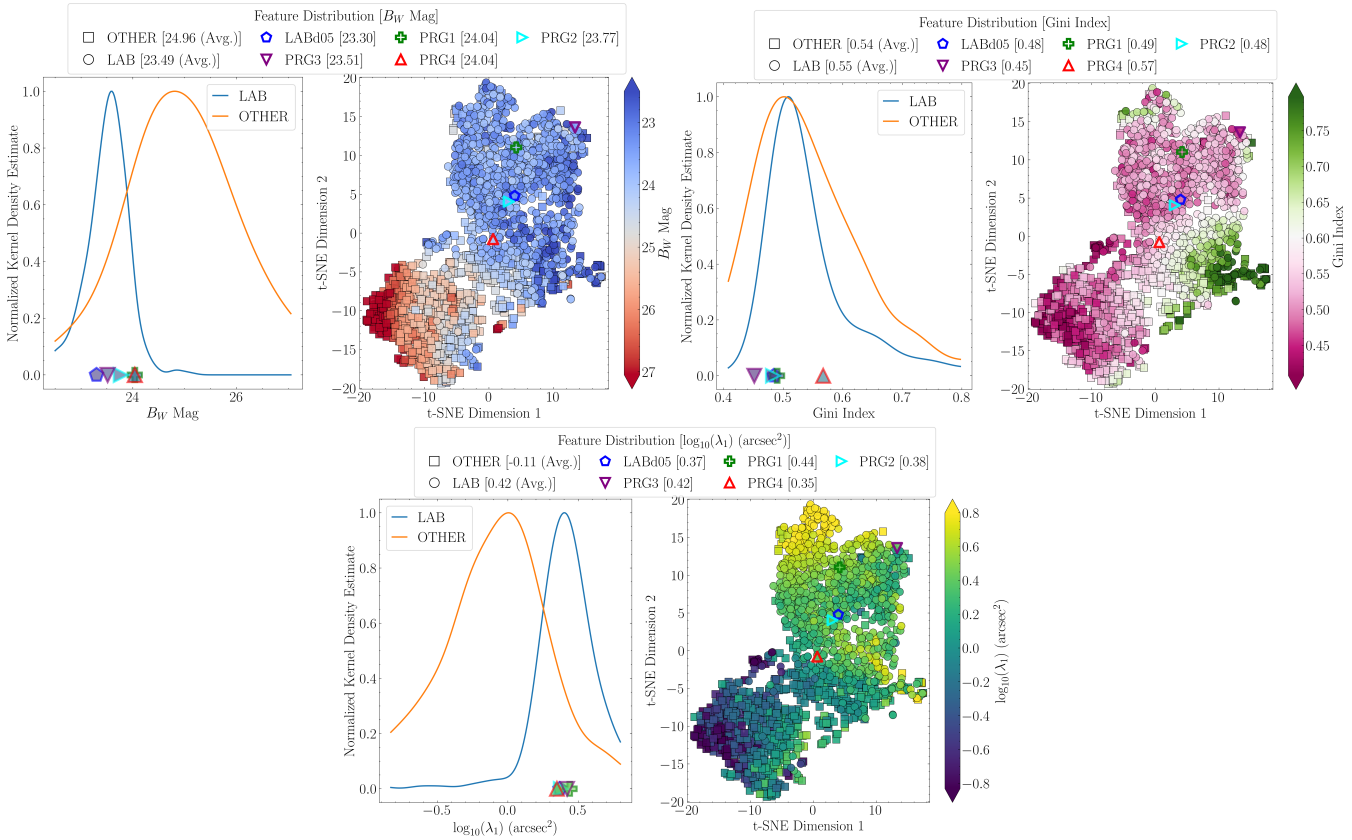

Figure 4. Top three features used in the `XGBoost-8` model. For each feature, the left panel shows the kernel density estimate (KDE) for the LAB and OTHER training classes, while the right panel shows the t-SNE projection of the 8D feature space with points colored by that feature's value (colorbar scale). LAB-class objects are plotted as circles, OTHER-class objects as squares, and the five confirmed LABs (also included in the KDEs) are shown with unique markers. The legend reports the class-averaged feature values for both classes and the individual values of the five confirmed LABs.

averages $P(y = \text{LAB} \mid \mathbf{X}) \approx 0.1$. These values reflect strong recall and effective class separation, and align with the observed filtering rates, indicating that the classifiers are well-calibrated.

To examine model behavior, in Fig. 4 we present the distribution of the top three features used to train `XGBoost-8`. The left panels show kernel density estimates (KDE) of the feature values for the LAB and OTHER training classes. The right-hand panels display the same 2D t-SNE projection as in the lower-right panel of Fig. 3, but with point colors scaled by the respective feature values. In both plots, confirmed LABs are denoted with unique markers (matching those in Fig. 3), and the legends include both the mean feature values for each class and the individual values for the confirmed LABs. These visualizations highlight where morphological traits cluster within the feature space, providing insight into the classifier's decision process and revealing which sources are most challenging to classify. For example, the top left panel highlights the apparent B_W magnitude, showing that LABs tend to be brighter than typical survey objects, although still faint with an average B_W mag of $\sim$ 23. The KDE indicates that LABs are, on average, $\sim$ 1.5 mag brighter than the OTHER class, which includes sources approaching the survey's B_W limiting magnitude of $\sim$ 27. In the corresponding t-SNE projection, a concentration of faint (B_W mag $\sim$ 27) OTHER-class objects appears near the bottom left edge of the feature space. As seen in the probability-scaled t-SNE projection (lower-right panel in Fig. 3), this region is dominated by confidently rejected instances with $P(y = \text{LAB} \mid \mathbf{X}) \lesssim 0.2$. A gradient is visible in which brighter OTHER-class instances (B_W mag $\lesssim$ 24) lie closer to the decision boundary. The brightest sources are present in the middle-right region of the feature space, which contains both LAB and OTHER instances. These represent ambiguous cases, as reflected in Fig. 3, where this region in the feature space contains objects with $P(y = \text{LAB} \mid \mathbf{X}) \approx 0.5$.

The top-right panels of Fig. 4 analyze the Gini index. The KDE shows that both classes cluster around Gini index $\sim$ 0.55, but the LAB class exhibits a narrower peak, indicating more consistent flux distributions. In contrast, the OTHER class spans a wider Gini index range and includes a distinct population of low-Gini index sources ($\lesssim$ 0.45), which occupy the bottom-left region of the feature space and coincide with the faintest sources in the B_W-scaled t-

SNE projection. Similarly, the middle-right region, containing the brightest sources, corresponds to the highest-Gini index values ($\gtrsim 0.7$). A less prominent region of high Gini index sources is also present at the top of the feature space, which also coincides with lower-probability predictions. Thus, while very low-Gini index sources are confidently rejected, the substantial overlap in high-Gini index values between classes contributes to model uncertainty for bright, compact sources. The observed Gini index gradient does not align with the decision boundary but instead reflects the transition from low to high Gini index sources across the 8D feature space.

The lower panels in Fig. 4 show the distribution of λ_1, the first eigenvalue of the covariance matrix, which measures variance along the major axis of the flux distribution and captures the extent of elongation. LABs tend to have significantly higher λ_1 values, consistent with their extended morphologies. The bottom-left of the feature space contains OTHER-class sources with $\lambda_1 \lesssim 0.4$ arcsec^2, representing compact, faint objects with more uniform emission (lower Gini index) that are most distinct from LABs. In contrast, sources with $\lambda_1 \gtrsim 1.5$ arcsec^2 lie near the decision boundary, where classification is more challenging. The most extended sources ($\lambda_1 \gtrsim 6$ arcsec^2) occupy the top of the feature space, a region dominated by LABs but also containing a noteable number of OTHER-class sources. As seen in the probability-scaled t-SNE plot of Fig. 3, these large, non-LAB objects often receive high LAB probabilities, reflecting the model's tendency to associate larger spatial extent with LAB-like morphology. Like B_W magnitude, the λ_1 distribution produces a clear gradient in classifier output, with larger values strongly indicative of LABs and driving higher confidence predictions.

Overall, the t-SNE feature space reveals that faint, compact, low-Gini sources in the bottom-left are confidently rejected; bright, compact, moderately large sources in the middle-right yield intermediate probabilities; the largest sources at the top are mostly LABs but include high-probability false positives; and mid-brightness, intermediate-size objects along the decision boundary are the most ambiguous cases.

2.3.5. *Initial Candidates*

To evaluate which of the two optimized `XGBoost` models is more effective at filtering LAB-like objects, we compute their respective receiver operating characteristic (ROC) curves, showing the true positive rate (recall) as a function of the false positive rate for varying classification thresholds (T. Fawcett 2006). The left panel of Fig. 5 shows the ROC curves for the optimized `XGBoost` models. The area under the curve (AUC) score quantifies overall classifier performance, with a value of 0.5 indicating random classification and 1.0 representing perfect classification. We perform 10-fold cross-validation and compute the associated uncertainties. As a benchmark, we also include the optimal baseline `XGBoost` model from Fig. 2, which trains on all features and undergoes no hyperparameter tuning. All models exhibit high performance, with AUC values of ~ 0.975. Among them, `XGBoost-8` achieves the highest AUC, followed by the baseline model and then `XGBoost-45`, although the scores of all three models are within the error bars.

Despite their comparable AUCs, the models differ substantially in the distribution of probability predictions during inference. Applying the three models to classify the ~ 1.97 million detected sources in NDWFS Boötes sources, `XGBoost-45` outputs ≈ 217 thousand LAB candidates with $P(y = \mathrm{LAB} \mid \mathbf{X}) \geq 0.5$, `XGBoost-8` returns ≈ 221 thousand, and the baseline model yields the fewest candidates (≈ 214 thousand).

The middle panel of Fig. 5 shows the empirical cumulative distribution function (ECDF) of probabilities for the positive predictions of each model. These curves reveal that `XGBoost-8` is the most conservative classifier, assigning high probabilities less frequently than either `XGBoost-45` or the baseline model. For example, only 24.5% of its positive predictions exceed $P(y = \mathrm{LAB} \mid \mathbf{X}) \geq 0.9$, compared to 50.8% for `XGBoost-45` and 56% for the baseline. This trend suggests that models trained with more features, particularly without proper hyperparameter tuning, are more prone to overfitting, assigning high probabilities to a larger number of objects. As a result, such models can produce candidates with $P(y = \mathrm{LAB} \mid \mathbf{X}) \simeq 1.0$, whereas `XGBoost-8` never assigns maximum confidence.

A similar pattern is seen in the training set (right panel of Fig. 5), which shows the ECDFs of leave-one-out cross-validation (LOOCV) probabilities for positively classified LAB training instances. All three models achieve comparable recall rates of ~ 0.94, but the baseline model assigns $P(y = \mathrm{LAB} \mid \mathbf{X}) \geq 0.9$ to 87.2% of its correct LAB classifications, compared to 86.6% for `XGBoost-45` and 73.3% for `XGBoost-8`.

Given that both optimized models correctly recover four of the five confirmed LABs with $P(y = \mathrm{LAB} \mid \mathbf{X}) \gtrsim 0.97$ (see Fig. 3), and that most LABs are recovered with $P(y = \mathrm{LAB} \mid \mathbf{X}) \geq 0.9$, we adopt a threshold of $P(y = \mathrm{LAB} \mid \mathbf{X}) \geq 0.9$ to refine our candidate sample. Although this excludes PRG4, this source has the smallest spatial extent in the B_W imaging and the least diffuse (highest Gini index) morphology of the known LABs, consistent with the compact Lyα emission reported by the spectroscopic follow-up from M. K. M. Prescott et al. (2013). As shown in Fig. 4, PRG4 lies within the decision boundary of `XGBoost-8`. We thus conclude that a 0.9 threshold effectively prioritizes the most extended and morphologically

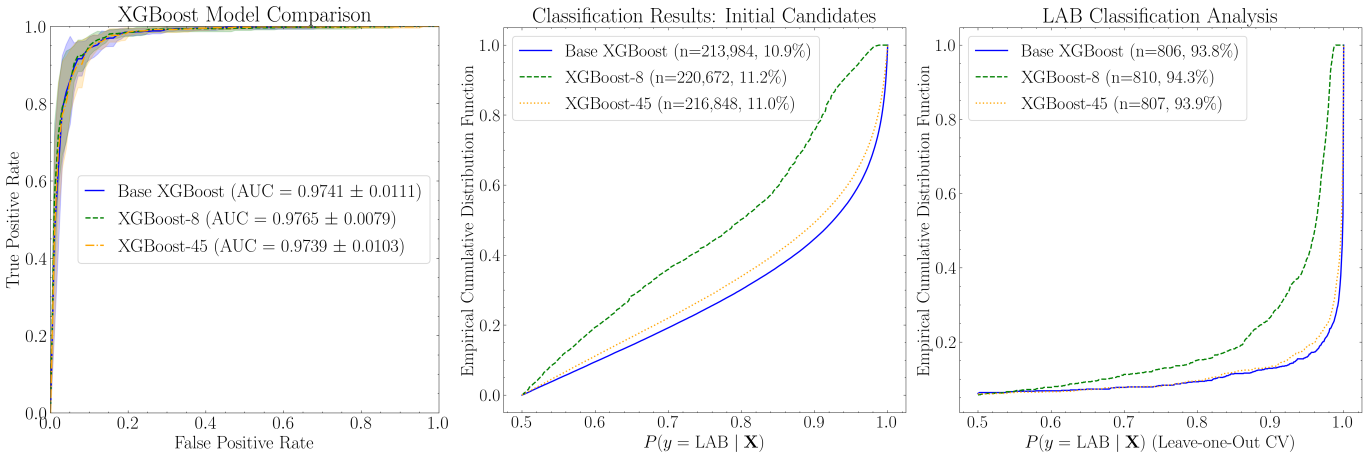

Figure 5. Classification and model performance for the baseline and optimized `XGBoost` models. The left panel shows the receiver operating characteristic (ROC) curves for each model, calculated using 10-fold cross-validation. The legend reports the corresponding area under the curve (AUC) score for each model. The middle panel presents the empirical cumulative distribution functions of predicted probabilities for the positive predictions from the ∼ 2 million classified sources in NDWFS Boötes. The legend displays how many positive predictions were output in each case as a total number and percentage of the ∼ 2 million cataloged objects. The right panel shows the empirical cumulative distribution functions for the positive predictions from the LAB class, calculated using leave-one-out cross-validation. The legend also lists how many instances from the LAB class (as a total number and as a percentage of the class size) were output as positive predictions by each model.

distinct LABs. Applying this threshold, `XGBoost-8` yields 53,518 LAB candidates, a ∼ 76% reduction in candidate sample size.

To reduce contamination while maintaining high recall for extended LABs, we select `XGBoost-8` as our optimal classifier. While all three models perform well, `XGBoost-8` provides the most conservative and reliable probability estimates. Despite being trained on the fewest features, it recovers nearly all known LABs while avoiding overconfident predictions that can increase false positives. Its probability distribution reduces contamination from ambiguous sources, and its selected features effectively capture key morphological traits such as brightness, flux distribution, and spatial extent without overfitting.

We henceforth refer to the resulting 53,518 high-confidence ($P(y = \mathrm{LAB} \mid \mathbf{X}) \geq 0.90$) `XGBoost-8` predictions as the "initial candidate" sample. Because the OTHER class was constructed via random sampling, there is a nonzero probability that some undiscovered LABs were included. As discussed in Section 2.3.4, we conducted LOOCV on the OTHER class, from which 74 were assigned $P(y = \mathrm{LAB} \mid \mathbf{X}) \geq 0.5$. Among these, 18 exceeded the imposed probability prediction threshold of 0.9. We include these 18 in the initial candidate sample for further assessment.

Fig. 6 presents an overview of the `XGBoost-8` classification output. The top panel shows the probability distribution for all cataloged survey objects. The LAB training class (black histogram) occupies the high-probability tail, while most NDWFS Boötes sources are classified with $P(y = \mathrm{LAB} \mid \mathbf{X}) \leq 0.2$. The dashed vertical line marks the $P(y = \mathrm{LAB} \mid \mathbf{X}) \geq 0.9$ threshold. Confirmed LABs, denoted with unique markers, cluster near the high-probability end, validating this threshold as effective for capturing known sources. The lower three panels of Fig. 6 examine the top three features used to train `XGBoost-8` (presented in Fig. 4). These are plotted as binned averages with 1σ scatter. The second panel from the top shows that most low-probability sources are faint ($B_W \gtrsim 24$), whereas LABs and high-probability candidates are brighter ($B_W \lesssim 24$). Sources approaching the survey limit are assigned $P(y = \mathrm{LAB} \mid \mathbf{X}) \leq 0.3$, and those fainter than $B_W \lesssim 25$ dominate the low-probability tail. Even the brightest LABs have $B_W \gtrsim 22$, emphasizing the importance of features beyond brightness alone.

The third panel shows classification output as a function of Gini index, which remains roughly constant until $P(y = \mathrm{LAB} \mid \mathbf{X}) \approx 0.9$ after which it declines sharply from ∼ 0.65 to ∼ 0.5. This trend aligns well with the LAB population, indicating that the Gini index is important for identifying high-confidence LAB candidates. The lower panel presents the first eigenvalue of the covariance matrix (λ_1), which quantifies source extent. Although the correlation is weaker, larger λ_1 values correspond to higher probabilities, consistent with the expectation that spatially extended sources are more likely to be LABs.

2.4. *Outlier Removal*

Although the `XGBoost-8` classifier effectively reduces contamination from survey artifacts, visual inspection of its output reveals a small subset of outliers in the imaging data. These outliers include masked regions due to bleed trails, bright star halos that contaminate pixel distributions,

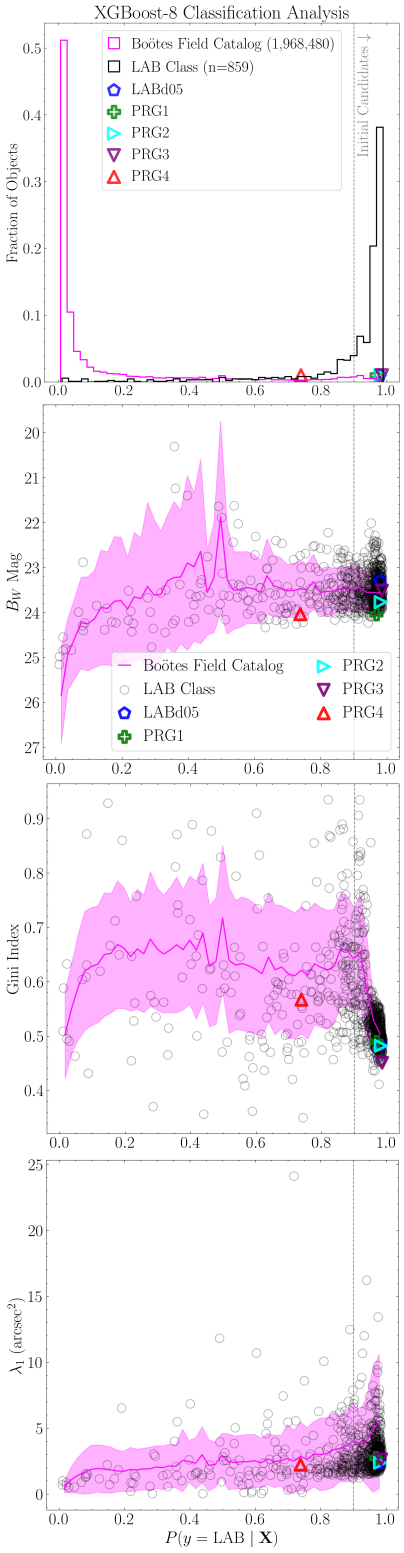

Figure 6. `XGBoost-8` classification analysis. The upper panel shows the probability predictions of all cataloged objects (pink) and the LAB class from LOOCV (black circles). The confirmed LABs are denoted by unique markers. The lower three panels show the influence of the top three training features on the model output.

and image edge effects, as demonstrated in the top panel of Fig. 7. These features often result in non-physical segmentation patches and unrealistic catalog values, such as large areas due to exaggerated patch sizes. To refine the candidate sample prior to training the multi-band CNN, we implement a dedicated detection step to remove sources flagged as outliers in either the B_W or R-bands. We defer this step until after the initial classification since it requires processing large, full-resolution cutouts in multiple filters, making it computationally expensive. Unlike catalog-level cuts (e.g., based on the number of stacked exposures), this approach avoids relying on metadata, which may not always be available. By relying solely on science image data, our pipeline remains generalizable to different datasets and immediately applicable to other multi-band imaging surveys.

We use the `scikit-learn` implementation of the Isolation Forest (`iForest`; F. T. Liu et al. 2008) algorithm, an unsupervised, tree-based ensemble method for anomaly detection that isolates outliers based on the assumption that they occupy sparse regions of the feature space. The model is trained on the "normal" class, hereafter referred to as the inliers. The model consists of N decision trees T_i, where $i = 1, \ldots, N$, each grown through random recursive partitions of a subsample of the data until a maximum depth is reached or no further splitting is possible (i.e., all "leaf" nodes contain a single data point). As the partitions are random, the number of nodes that a data point traverses from a "root to leaf", $h_i(x)$, provides a measure of how inline the data is within the distribution, with outliers generally exhibiting shorter path lengths than inliers.

The `iForest` score is computed as

$$S(x) = 0.5 - 2^{-\frac{\mathbb{E}[h(x)]}{c(n)}} \tag{6}$$

where $\mathbb{E}[h(x)] = (1/N)\sum_{i=1}^{N} h_i(x)$ is the average path length across the forest, and $c(n)$ is the average number of steps needed to isolate a sample in a random tree of size n. Scores range from -0.5 (strong outliers, isolated in few steps) to 0.5 (strong inliers, requiring many partitions), with values near zero representing paths close to the expected average.

The inlier class used to train the `iForest` is the LAB class from the previous model, which was visually inspected in Section 2.2 to ensure good B_W imaging. After an additional R-band inspection, 15 outlier cases are removed, leaving 845 inliers for training. We reserve 100 of these as a test set, and train the `iForest` on the remaining 745. In addition, we visually inspect a subset of the initial candidates to select 1000 sources that we considered to be outliers in either band. Representative B_W-band examples are shown in the top panel of Fig. 7.

To train the classifier, we evaluated five different feature sets — Stats, FFT, Wavelet, LBP, and HOG. Each set is extracted separately from the B_W and R-band imaging and

concatenated to form the final feature matrix. Global intensity properties (Stats) were quantified using ten statistical moments and robust statistics including the mean, standard deviation, median, median absolute deviation, skewness, kurtosis, minimum, maximum, and the 1st and 99th percentiles. We also consider Fourier energy features (FFT), computed from the 2D power spectrum by integrating the spectral power within concentric radial frequency bands. The bands are defined by the Nyquist-normalized frequency edges (0.00, 0.10, 0.25, 0.50, 0.75, 1.00), and the resulting values are expressed as fractions of the total power. A Hann window is applied to reduce spectral leakage, which we find improves performance (F. Harris 1978, see also W. E. Sabin 2008). In addition, we apply a 2D discrete wavelet transform (Wavelet) using the `PyWavelets` Python package (G. Lee et al. 2019). Each image cutout is decomposed with a Daubechies-4 basis and symmetric boundary extension (I. Daubechies 1992), and the energies (sums of squared coefficients) of the approximation and horizontal, vertical, and diagonal detail subbands at each scale are computed and log-transformed. Using the `scikit-image` API (S. van der Walt et al. 2014), we also compute Local Binary Patterns (LBP; T. Ojala et al. 2002) and Histogram of Oriented Gradients (HOG; N. Dalal & B. Triggs 2005) features which capture fine-scale texture and local edge/shape information, respectively. LBP encodes texture by examining a circular neighborhood of P sampling points at radius R around each pixel in the image. Starting from a fixed direction (e.g., clockwise from the top neighbor), each sampling point is assigned a value of 1 if its intensity is greater than or equal to that of the central pixel, and 0 otherwise. These binary values are concatenated in order to form a P-bit binary code, which is then converted to a decimal value. The occurrence frequencies of these codes are accumulated into per-channel histograms, which serve as the texture features. On the other hand, HOG extracts edge and contour information by computing intensity gradients at each pixel, binning their orientations into local cells, and normalizing the cell histograms over larger overlapping blocks. Both the LBP and HOG feature sets are computed using the `scikit-image` default parameters.

To assess sensitivity across spatial scales, we extracted each feature set from square image cutouts ranging from 50×50 to 250×250 pixels ($\sim 13 - 64$ arcsec) and trained a separate `iForest` model at each scale. As there is no well-defined semi-supervised validation metric for optimizing `iForest` hyperparameters, no hyperparameter tuning is performed for this model and the default hyperparameters are used. As HOG features require normalization, to ensure consistency and improve the feature extraction, the images are min-max normalized. From the initial candidate sample, the maximum pixel values are ~ 8 and ~ 15 counts s^{-1} in the B_W and R-bands, respectively; although the 99th percentiles in the respective bands are $\ll 1$ counts s^{-1}. We therefore normalize using a minimum pixel value of 0 and a maximum of 10.

The outlier detection rate (middle panel of Fig. 7) increases with image size for HOG (~ 0.6–0.9) and for LBP (~ 0.2–1), remains ~ 0.8 for FFT, and shows only modest gains for Wavelet ($\sim 0.6 - 0.7$) and Stats ($\sim 0.3 - 0.4$). These trends reflect a scale-dependent context effect with larger cutouts better capturing halos, trails, edges, and background structure that edge and texture-based features like HOG and LBP especially benefit from. The strong performance of HOG and LBP also reflects sensitivity to large uniform regions and sharp boundaries introduced by missing-data patches, which become more evident at larger scales. LBP yields low outlier detection on small cutouts but improves most rapidly with image size as its per-pixel local binary pattern histograms gain many more "neighborhoods", reducing sampling noise and increasing the likelihood of capturing artifact-related patterns. In contrast, both FFT and Wavelet features are averages, while Intensity-based Stats lack spatial detail, so their gains with size are limited.

The inlier detection rate (lower panel of Fig. 7) rises for HOG from 0.97 to 1.00 and remains there for the largest cutouts (≥ 60 arcsec), settles near ~ 0.95 for LBP, and declines for Stats, Wavelet, and FFT from $\sim 0.90 - 0.95$ to $\sim 0.80 - 0.85$ as size increases. The weaker performance of Stats, Wavelet, and FFT likely stems from their limited ability to isolate the signal of the central source once larger regions of sky are included. Intensity-based statistics average over the full aperture and therefore become increasingly sensitive to background variance and neighbors that are not related to the target. Likewise, Wavelet features integrate over spatial position, whereas FFT features summarize global frequency content and average over directions. This can yield lower inlier detection rates as image size grows and more contamination from unrelated large-scale modes and low-frequency background is included. By contrast, HOG and LBP maintain high inlier detection because their gradient and neighborhood-based encoding emphasize local structure. As such, even with larger images, the feature matrix when using these methods still explicitly includes the texture features of the central source. This robustness to scale-dependent noise enables HOG and LBP to preserve near-perfect inlier detection at larger image sizes, unlike the steady decline observed for averaging-based methods.

The gray dashed line in Fig. 7 at 62 arcsec marks the optimal image size for feature extraction. At this size, LBP achieves 100% outlier removal but retains only 95% of inliers, whereas HOG removes 90% of outliers while retaining 100% of inliers. The higher inlier retention with HOG is consistent with its block-normalized gradients, which reduce sensitivity to background level and brightness variation. As

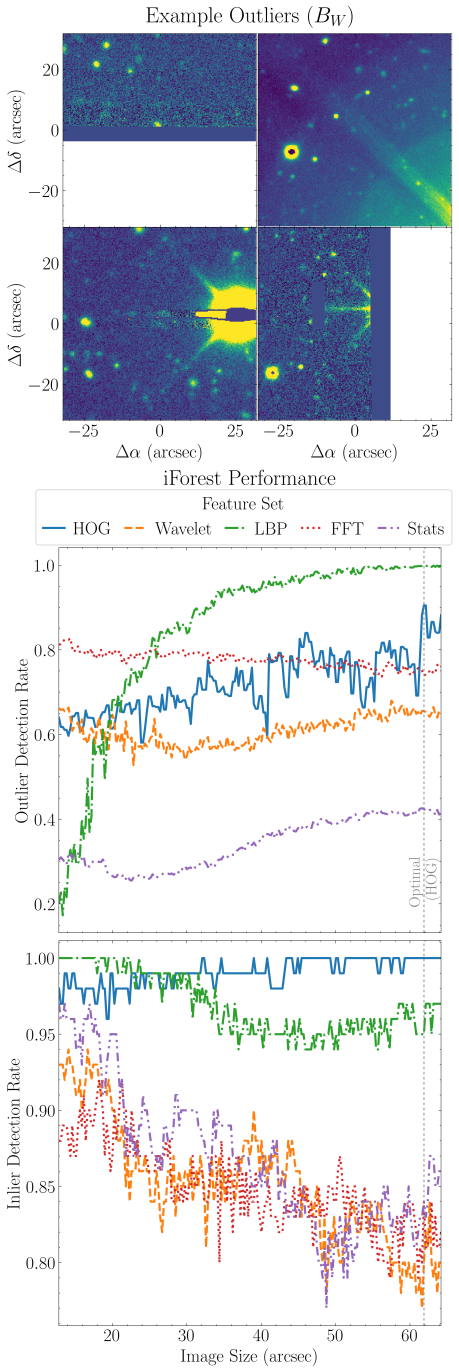

Figure 7. Outlier classification results. The top panel shows examples of outliers in the B_W imaging. The middle panel illustrates the performance of an inlier-trained `iForest` classifier in detecting outliers, as a function of image size. Five different feature sets are used to train the models to assess the optimal features and image size for outlier detection. The bottom panel presents the inlier detection rate from the hold-out test set. The gray dashed line marks the image size at which we achieve optimal performance, balancing both outlier and inlier detection rates.

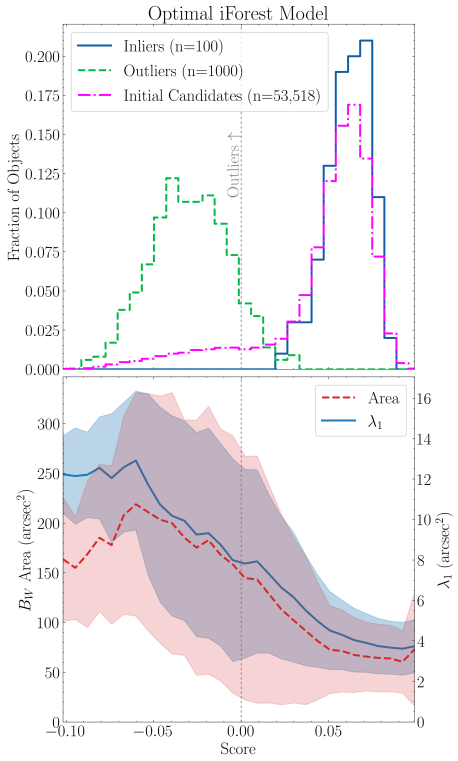

Figure 8. Distribution of `iForest` scores. The top panel shows the distribution of `iForest` scores from the optimal model trained with HOG features, applied to both the test set outliers and inliers (as shown in Fig. 7), alongside the distribution for the initial candidates. The bottom plot presents the bin averages for the segmentation-based area and λ_1 features used to train the `XGBoost` models, as a function of the corresponding `iForest` scores for the initial candidates.

this stage seeks to maximize outlier removal prior to CNN training while preserving the LAB candidate sample, we adopt an `iForest` trained with HOG features extracted from 62×62 arcsec cutouts as the optimal model. This choice removes most contaminants while preserving true candidates.

Using an `iForest` model trained on HOG features at the optimal image size, we classify the 53,518 initial candidates from Section 2.3.5. Of these, 10.63% are flagged as outliers and removed, yielding a refined sample of 47,829 candidates. The top panel of Fig. 8 shows the distribution of the `iForest` scores for the full candidate set (pink), alongside that of the inlier (blue) and outlier (green) test sets. The distribution of the initial candidates is bimodal, with one mode aligning with the inliers and another falling below the score threshold of < 0, where the outlier population lies. Visual inspection of a subset of flagged cases confirms that this step successfully identifies outliers in the B_W and R-band imaging.

The bottom panel of Fig. 8 shows bin-averaged `iForest` scores for the initial candidates versus segmentation area and the first eigenvalue of the spatial covariance matrix (λ_1), plotted with dual y-axes. Although area and λ_1 quantify different

morphological properties, they are strongly correlated. We include both to illustrate that the trained `iForest` model is sensitive to structural anomalies that arise during our segmentation procedure. This demonstrates how the area of the segmentation patch can be used to identify some anomalous features caused by bright halos, which can yield unrealistically large B_W areas of $\gtrsim$ 150 arcsec^2, corresponding to segmentation masks that span roughly a quarter of the entire $\sim 26 \times 26$ arcsec image cutouts. Meanwhile, λ_1, which was also identified as an important feature by our `XGBoost-8` model, serves as an effective proxy for spatial extent. The observed trend suggests that a λ_1 cut of $\gtrsim$ 8 arcsec^2 could independently filter out some outliers. As shown in Fig. 6, the most spatially extended sources, which are on average those with the highest `XGBoost-8` scores, tend to fall below this threshold. This further supports the `iForest`'s ability to identify structurally abnormal sources when trained on HOG features.

2.5. *Multi-band Classifier: Convolutional Neural Networks*

Convolutional neural networks (CNNs) have become widely used in astronomy due to their ability to automatically learn complex spatial features from multi-band imaging data, eliminating the need for handcrafted feature design (I. Goodfellow et al. 2016). CNNs have been successfully applied to a range of astronomical problems, including galaxy morphology classification (C. J. Lintott et al. 2008; K. W. Willett et al. 2013; S. Dieleman et al. 2015), exoplanet detection (J. P. Terry et al. 2022), and stellar parameter inference (S. Fabbro et al. 2017). A key strength of CNNs lies in their robustness to small shifts in position, shape, and scale (K. Fukushima 1980; Y. Lecun et al. 1998), enabled by convolutional kernels that capture patterns at multiple spatial scales. Pooling operations, such as "max pooling", downsample the feature maps by summarizing information in local regions (e.g., by taking the maximum value). This reduces computational cost, increases translational invariance, and helps the model retain dominant features while minimizing sensitivity to noise (Y.-L. Boureau et al. 2010). These properties make CNNs particularly well-suited to detecting irregular emission features. This is especially important in the context of finding Lyα blobs given that Lyα escape processes often produce diffuse, spatially extended morphologies lacking sharp boundaries (e.g., J. M. Mas-Hesse et al. 2003; M. Dijkstra et al. 2006; C.-A. Faucher-Giguère et al. 2010).

CNNs typically require large training sets to generalize effectively. For example, the original AlexNet (A. Krizhevsky et al. 2012) was trained on over a million images from the ImageNet dataset (J. Deng et al. 2009), which includes images of various object categories ranging from animals to everyday objects. Astrophysical CNNs such as GAMORNET (A. Ghosh et al. 2020) and Toothless (A. K. Aniyan & K. Thorat 2017) were trained on $10^4 - 10^5$ simulated galaxy images. Prior work has shown that even shallow CNNs often require $\gtrsim 10^3$ samples per class to achieve accuracies of 70 – 90% (L. Brigato & L. Iocchi 2020).

In contrast, our LAB training set consisting of real sources is orders of magnitude smaller and requires careful architectural design, regularization, and data augmentation to avoid overfitting. We start with the 85 LAB priority candidates identified by M. K. M. Prescott et al. (2012), which as described in Section 2.1, were selected from an initial pool of 865 sources based on both spatial extent and blue color excess relative to a known LAB in the same field. Since our classification approach relies on multi-band (B_W and R) imaging, it is essential that the training set be selected using the same filters. The initial 865 candidates were selected based on B_W imaging alone and do not reflect the full multi-band selection criteria relevant to our task. In contrast, the 85 priority candidates incorporate both size and color information consistent with high-redshift Lyα emission, making them suitable for CNN training. The five spectroscopically confirmed LABs are reserved for final model testing, leaving the remaining 80 for training and validation. Of these, we retain only those assigned a classification probability of $P(y = \text{LAB} \mid \mathbf{X}) \geq 0.9$ by the `XGBoost-8` model, yielding a training set of 70 LAB candidates. This is done to be consistent with the OTHER class, which is randomly sampled from the $\approx 48,000$ initial candidates that were classified with $P(y = \text{LAB} \mid \mathbf{X}) \geq 0.9$ by `XGBoost-8` and output as inliers by the `iForest` model.

This sample of 70 positive instances is clearly insufficient for robust CNN training. To address this limitation, we employ several strategies, including cross-validation, training for fewer epochs, reducing model complexity by using smaller input image sizes, and applying regularization techniques such as "dropout", which randomly disables a fraction of neurons during training to prevent overfitting and promote the learning of more generalizable features. Furthermore, we perform extensive data augmentation, which increases the effective training set size and improves generalization. Augmented samples are generated using random combinations of the following four types of transformations: horizontal/vertical shifts (±10 pixels), horizontal/vertical flips, rotations (0° – 360°), and zooming (±10%). Images are initially extracted at larger dimensions to avoid edge artifacts introduced during transformation and subsequent cropping. Although online augmentation (i.e., transformations sampled on the fly each training epoch) can increase diversity, in our limited-data and few-epoch setting we adopt offline augmentation (i.e., a fixed, precomputed set) so the augmented set stays constant across epochs and can be inspected, with the aim of stabilizing the optimization and fold-level evaluation.

Layer	AlexNet	Output	This Work	Modified Output
Input Image	(R, G, B)	$(227 \times 227 \times 3)$	(B_W, R, B_W/R)	$(70 \times 70 \times 3)$
Conv2D	Normal init., ReLu; k=11, s=4, p=`valid`	$(55 \times 55 \times 96)$	He normal init., ReLu; k=11, s=4, p=`same`	$(18 \times 18 \times 96)$
Pooling 1	Max Pooling; k=3, s=2, p=`valid`	$(27 \times 27 \times 96)$	Max Pooling; k=3, s=2, p=`same`	$(9 \times 9 \times 96)$
Regularizer	Local Response Normalization	...	None	...
Conv2D	Normal init., ReLu; k=5, s=1, p=2	$(27 \times 27 \times 256)$	He normal init., ReLu; k=5, s=1, p=`same`	$(9 \times 9 \times 256)$
Pooling 2	Max Pooling; k=3, s=2, p=`valid`	$(13 \times 13 \times 256)$	Max Pooling; k=3, s=2, p=`same`	$(5 \times 5 \times 256)$
Regularizer	Local Response Normalization	...	None	...
Conv2D	Normal init., ReLu; k=3, s=1, p=1	$(13 \times 13 \times 384)$	He normal init., ReLu; k=3, s=1, p=`same`	$(5 \times 5 \times 384)$
Conv2D	Normal init., ReLu; k=3, s=1, p=1	$(13 \times 13 \times 384)$	He normal init., ReLu; k=3, s=1, p=`same`	$(5 \times 5 \times 384)$
Conv2D	Normal init., ReLu; k=3, s=1, p=1	$(13 \times 13 \times 256)$	He normal init., ReLu; k=3, s=1, p=`same`	$(5 \times 5 \times 256)$
Pooling 3	Max Pooling; k=3, s=2, p=`valid`	$(6 \times 6 \times 256)$	Max Pooling; k=3, s=2, p=`same`	$(3 \times 3 \times 256)$
Flatten	Flatten	9216	Flatten	2304
FC	Normal init., ReLu	4096	Glorot uniform init., tanh	4096
Dropout	0.5	4096	0.5	4096
FC	Normal init., ReLu	4096	Glorot uniform init., tanh	4096
Dropout	0.5	4096	0.5	4096
Output	Softmax	1000	Sigmoid	1
Total Parameters		~ 60M		~ 30M

Table 3. Comparison of the AlexNet CNN and the modified AlexNet model we adopt in this work.

Nonetheless, online augmentation remains a promising avenue for future exploration.

2.5.1. *Model Architecture & Optimization*

Our CNN follows an AlexNet-style architecture, building directly on prior astronomical implementations such as GAMORNET and Toothless (A. K. Aniyan & K. Thorat 2017; A. Ghosh et al. 2020). AlexNet-style CNNs have been shown to achieve classification accuracies of $\gtrsim 0.8$ even with only a few thousand training instances, similar to our case (Z. Zhang et al. 2022; M.-X. Fu et al. 2024).

The network is implemented in `TensorFlow` (M. Abadi et al. 2015) and consists of five convolutional layers (Conv2D) with kernel sizes (k) and strides (s) adopted from the AlexNet framework, each followed by a rectified linear unit (ReLU; V. Nair & G. E. Hinton 2010) activation, consistent with both GAMORNET and Toothless. Here k sets the spatial extent of each square convolutional kernel, while s controls how far the kernel moves between applications. The padding parameter (p) controls whether and how the input is padded during convolution to avoid shrinking feature maps unnecessarily. Unlike the original AlexNet, which used a mix of padding options such as `valid` (no padding) and explicit integer padding (fixed zeros around the input), we adopt the `same` padding option throughout for consistency, which automatically pads inputs so that output dimensions are preserved relative to the stride. Max pooling, which downsamples feature maps by taking the maximum value within non-overlapping windows, is applied after the first, second, and final convolutional layers.

The Conv2D layers are followed by three fully connected (FC) layers, in which each input node is connected to every output node. In line with the GAMORNET design, the FC layers use the tanh activation, which we found yielded more stable performance than ReLU in the dense layers of this architecture. The final output layer consists of a single neuron with a sigmoid activation, which yields the probability that the input image corresponds to the positive class. This single-node design is well suited to binary classification and further reduces redundancy, the total parameter count, and computational overhead.

For the loss function, we adopt binary cross-entropy. This choice aligns with Toothless, whereas GAMORNET used the multiclass cross-entropy formulation. The trainable weights of the Conv2D layers, which use the ReLU activation function, are initialized with He normal initialization (K. He et al. 2015). For the fully connected layers that apply the tanh activation, we use Glorot uniform initialization (X. Glorot & Y. Bengio 2010). These schemes are paired with their respective activations to preserve variance stability across layers at initialization, which facilitates faster convergence and lowers the risk of vanishing or exploding gradients.

To adapt the architecture for our LAB classification task, we systematically explored hyperparameters most relevant to generalization, focusing on the number of augmentations applied to the positive class and the input image size. Both factors significantly affected training time and classifier performance. We found that using 24 augmentations per instance and smaller input images of 70×70 pixels (18×18 arcsec) most effectively reduced overfitting, yielding an effec-

tive training set size of 1,344 per class. We also tested additional regularization and normalization layers. Following AlexNet, we applied Local Response Normalization (LRN; A. Krizhevsky et al. 2012) after the first two pooling layers, which normalizes a neuron's activity across nearby feature maps. We also evaluated performance using Batch Normalization (BN; S. Ioffe & C. Szegedy 2015), a more modern approach that normalizes activations across each batch to stabilize training, applied after each Conv2D and FC layer (except the output layer) and before activation. However, omitting normalization yielded better validation performance, likely due to the relatively small training set and shallow network depth, as BN can be less effective with small batch sizes (S. Ioffe & C. Szegedy 2015). For the final model we adopted a batch size of 8 and a learning rate of 10^{-5}, since larger batches converged faster but generalized worse, and higher learning rates caused unstable losses. For optimization we used the Adam optimizer (D. P. Kingma & J. Ba 2014) with the AMSGrad variant enabled (S. J. Reddi et al. 2019), which in our tests provided more stable convergence than standard stochastic gradient descent (SGD), particularly with the small batch sizes used to train the model. The optimizer hyperparameters β_1 and β_2, which govern the exponential moving averages of the gradient and its squared value in Adam, were set to their default values of 0.9 and 0.999, respectively.

In the context of image classification, pixel normalization is important because it standardizes the dynamic range of the input data, improving numerical stability during training and reducing sensitivity to outlier values. Since our pixel values represent counts per second, we fixed the minimum at zero and capped the maximum at 10 counts s^{-1}, consistent with the procedure used in the outlier classification step (Section 2.4). The network input consists of three channels, including the B_W and R-band image cutouts, and a third channel defined as the flux ratio B_W/R, which highlights the blue color excess expected for LABs at $z \sim 2-3$. Although deeper architectures might learn this feature implicitly, we found that explicitly including this difference image significantly improved classification performance and stabilized CNN training, especially given the limited number of training epochs and training set size. The final CNN model is summarized in Table 3, alongside the original AlexNet architecture for comparison.

2.5.2. *Final Model & Candidates*

We explored a range of model configurations by systematically varying the input image size, number of augmentations, batch size, and number of training epochs. Each configuration was evaluated using a combination of validation score, the agreement between training and validation losses, and the ability to filter a random subset of 1,000 objects. We then selected a representative model that balanced convergence

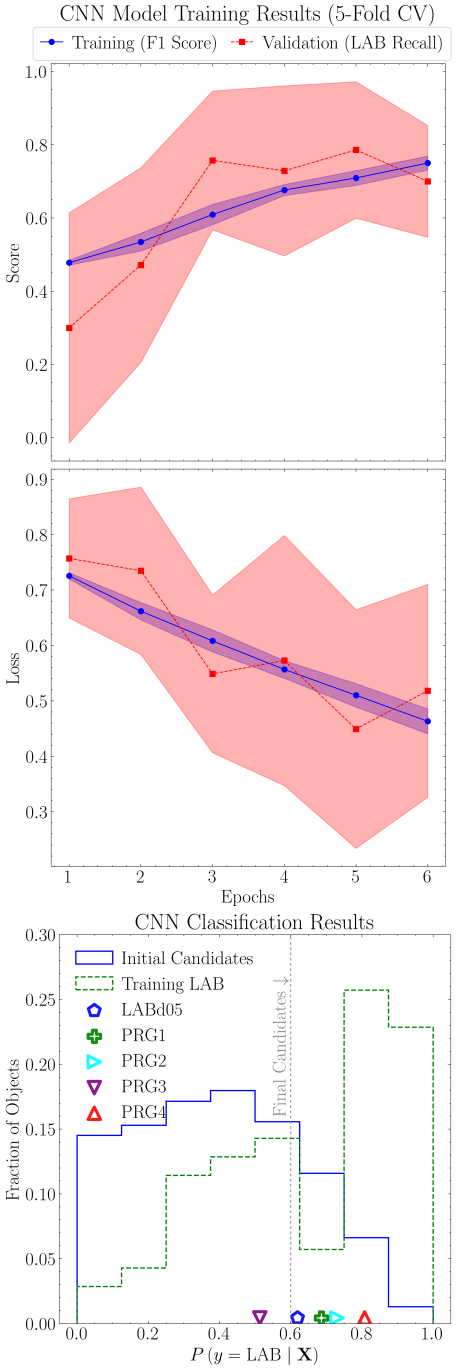

Figure 9. CNN model training performance and classification results. The top panel shows the F1 score for the training set and the binary accuracy for the validation set over six epochs. As the validation set contains only positive instances, the score is equivalent to the LAB recall. Shaded regions indicate the standard deviation across the five cross-validation folds. The middle panel presents the corresponding binary cross-entropy loss. The bottom panel shows the distributions of model predictions for the $\approx 48,000$ initial candidates (blue) and the 70 LAB candidates used for training (green). The five confirmed LABs reserved for final model testing are shown with unique markers.

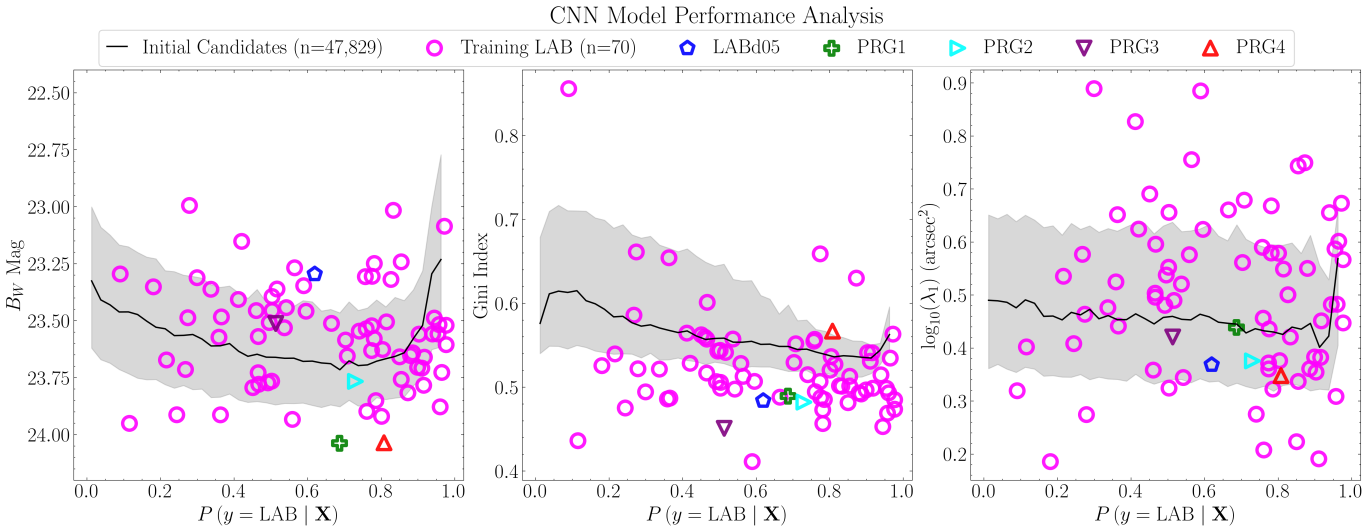

Figure 10. CNN model predictions as a function of the top three B_W-based features (from left to right) used to train `XGBoost-8`. The left panel shows the correlation between the B_W mag of the source and the CNN probability prediction for both the initial candidates (black line with shaded region showing the 25-75th quantile range) as well as the positive LAB class (pink circles). The five confirmed LABs that were reserved for final model testing are shown with unique markers. The middle and right panels display the same CNN predictions as a function of Gini index and the first covariance eigenvalue, respectively.

speed with filtering performance. Training for six epochs proved sufficient, with models trained beyond $\sim$ 10 epochs consistently overfitting. This behavior is expected given the limited size of our training set, as even models such as Toothless, which trained on $\sim 10^4$ sources, also converged within ten epochs. We perform five-fold cross-validation, reserving 14 LAB candidates for validation in each fold and augmenting the remaining 56 for training. As noted in Section 2.5.1, we augment each instance in the positive training class 24 times, yielding a total of 1,344 positive samples to train the CNN with. We then balance classes by randomly sampling 1,344 negatives from the $\approx 48,000$ initial candidates. We do not include negatives in the validation set, since constructing a clean set of 14 negative examples would require manual inspection of the initial candidates and risk introducing bias. Given the rarity of LABs, any contamination in the negative class is expected to be highly diluted and unlikely to affect training. Nonetheless, we avoid randomly selecting 14 negatives for validation as the small sample size would make the validation metric sensitive to even a few mislabeled instances or outliers, artificially inflating or lowering performance estimates. The cross-validation metric is thus limited to positive-class recall. During tuning, we monitored the false-positive rate by applying each candidate model to this independent 1,000-object subset, effectively serving as a negative validation sample.

With our cross-validation approach, the final classifier is an ensemble of five CNN models. Predictions on new data are averaged at inference time to yield the final classification probability. This ensemble strategy improves generalization and reduces sensitivity to the specific training partition, which is particularly important given the small number of LAB candidates. As detailed in Section 2.1, of the 26 priority candidates from M. K. M. Prescott et al. (2012), only five were confirmed as LABs. Assuming a similar success rate, $\sim$ 10 true LABs should be present among our 70 LAB training instances. Cross-validation therefore encourages the model to focus on consistent, shared features across both candidate and confirmed sources, although subtle traits unique to the confirmed LABs may be underrepresented. Future models trained exclusively on confirmed samples could improve identification of such signatures.

The training and validation performance during CNN training is shown in the top two panels of Fig. 9. The shaded regions denote the standard deviation across the five validation folds. As the validation set consists of only positive samples, the binary accuracy reported for the validation data in the top panel is the LAB recall. Across all folds, we observe similar learning behavior. Despite the higher variance in the validation set, caused by the smaller number of instances evaluated, both training and validation results converge to similar final values. In particular, the training F1 score and validation recall increase in parallel with epoch, while the binary cross-entropy losses decrease with epoch in a comparable manner, indicating consistent convergence across folds. The consistency among models trained with distinct subsets of LAB candidates indicates robust generalization across both confirmed LABs and candidate sources.

The lower panel of Fig. 9 shows the probability prediction distribution for the LAB class in the validation set (green), based on the output for each hold-out fold. We also present the classification results for the $\approx 48,000$ ini-

tial candidates (blue) and for the five confirmed LABs in the independent test set (unique markers). All five confirmed LABs are correctly classified with CNN probabilities of $P(y = \mathrm{LAB} \mid \mathbf{X}) > 0.5$. Interestingly, we note that PRG4, which achieves the highest CNN probability ($P(y = \mathrm{LAB} \mid \mathbf{X}) \approx 0.8$), had previously received the lowest probability in the B_W-based `XGBoost-8` classifier and was consequently removed earlier in the pipeline.

From the initial candidates, 17,048 are classified as LABs, corresponding to a filtering rate of $\sim 65\%$. The mean probability prediction for all initial candidates is $P(y = \mathrm{LAB} \mid \mathbf{X}) = 0.401$, in close agreement with the observed LAB fraction of ~ 0.35, indicating that the classifier is reasonably well calibrated. For the LAB training class, we obtain an average probability prediction of $P(y = \mathrm{LAB} \mid \mathbf{X}) = 0.645$, which agrees with the observed recall, as 48/70 (68.57%) of LABs are correctly classified during validation. Taken together, these results show that the CNN effectively identifies true LABs and LAB candidates while maintaining strong filtering capabilities, despite being trained on only 1,344 negative instances and 56 unique LABs per fold. In addition, when applied to the subset of 1,000 objects used for model tuning, the final CNN classifies 65.9% of these instances as OTHER, consistent with the ∼65% rejection rate in the full catalog. This suggests that the tuning procedure effectively constrained the false-positive rate while maintaining model calibration. These results demonstrate that relatively shallow CNNs such as AlexNet can perform well in low-sample-size astrophysical classification tasks.

Based on the classification output of the known LABs in our data, we proceed by imposing a probability prediction threshold of $P(y = \mathrm{LAB} \mid \mathbf{X}) \geq 0.6$ to filter out low-confidence candidates. This threshold is marked as a gray dashed line in the lower panel of Fig. 9, which reduces the candidate sample to 10,922, hereafter referred to as the "final candidates". While this threshold excludes PRG3, which was classified with $P(y = \mathrm{LAB} \mid \mathbf{X}) = 0.513$, we note that among the spectroscopically confirmed sample from (M. K. M. Prescott et al. 2013), this was among the sources that showed the weakest emission. We thus adopt this threshold to prioritize spectroscopic-follow resources on higher-confidence candidates.

In Fig. 10, we present the CNN probability predictions as a function of the three most important B_W-based features identified by the `XGBoost-8` model (see Fig. 3). These include the B_W magnitude (left panel), the Gini index (middle panel), and the first eigenvalue of the covariance matrix (λ_1, right panel). The shaded regions mark the interquartile range (IQR) of probability predictions, while LAB training objects and confirmed LABs are overplotted as scatter points. For B_W magnitude, the median value increases from ~ 23.25 at $P(y = \mathrm{LAB} \mid \mathbf{X}) \approx 0$ to ~ 23.75 at $P(y = \mathrm{LAB} \mid \mathbf{X}) \gtrsim 0.8$, suggesting that fainter sources generally receive higher CNN probabilities. At the highest probabilities ($\gtrsim 0.9$), the median magnitude decreases to B_W mag ~ 23.0. These bins, however, are sparsely populated. This trend may reflect small-number statistics rather than a systematic effect, although it could also indicate enhanced CNN confidence for the brightest LABs in the training set.

The middle panel shows a negative correlation with the Gini index, where the highest-probability LAB candidates exhibit Gini values $\lesssim 0.55$, consistent with the model learning to identify diffuse, low surface-brightness morphologies. In contrast, the right panel reveals no strong correlation between CNN probability and λ_1 across most of the probability range ($P(y = \mathrm{LAB} \mid \mathbf{X}) \lesssim 0.9$). At the higher probabilities of $\gtrsim 0.9$, λ_1 does show a modest increase from ~ 3 to ~ 4 arcsec^2, but this regime is sparsely populated. Overall, the trends suggest that the most confidently rejected candidates tend to be compact, high-Gini sources, while the most confidently accepted candidates are systems with lower Gini indices of ~ 0.55.

3. RESULTS

3.1. *Priority Candidates*

Although up to this stage our pipeline has reduced the initial catalog by 99.13%, the resulting candidate list is still too large for practical spectroscopic follow-up. To optimize the allocation of spectroscopic resources and maximize the probability of identifying high-z LABs, we prioritize the final candidate sample using morphological and color-based metrics. Following the framework of M. K. M. Prescott et al. (2012), candidates are ranked by their spatial extent in B_W and their $B_W - R$ color. As an additional discriminator, we incorporate the Gini index, which our earlier analysis (Section 2.3.5) identified as a strong indicator of the diffuse morphologies characteristic of high-z LABs. These criteria help further reduce the number of sources requiring follow-up while ensuring that the remaining candidates are those most consistent with previously confirmed LABs. Because of the high observational cost of spectroscopic follow-up, these cuts are applied only at the final stage of our pipeline. In future work, models trained exclusively on confirmed LABs may eliminate the need for such prioritization.

Fig. 11 illustrates this multi-step prioritization, from top to bottom. In each panel, the histograms show feature distributions of the filtered final candidates, with the red curve (right-hand y-axis) indicating the corresponding bin-averaged CNN probability predictions and their standard deviation spread. Green dashed lines mark the thresholds imposed by morphological cuts, with the overlaid confirmed LABs guiding the selection criteria. A shaded gray region highlights the excluded regions of the feature space.

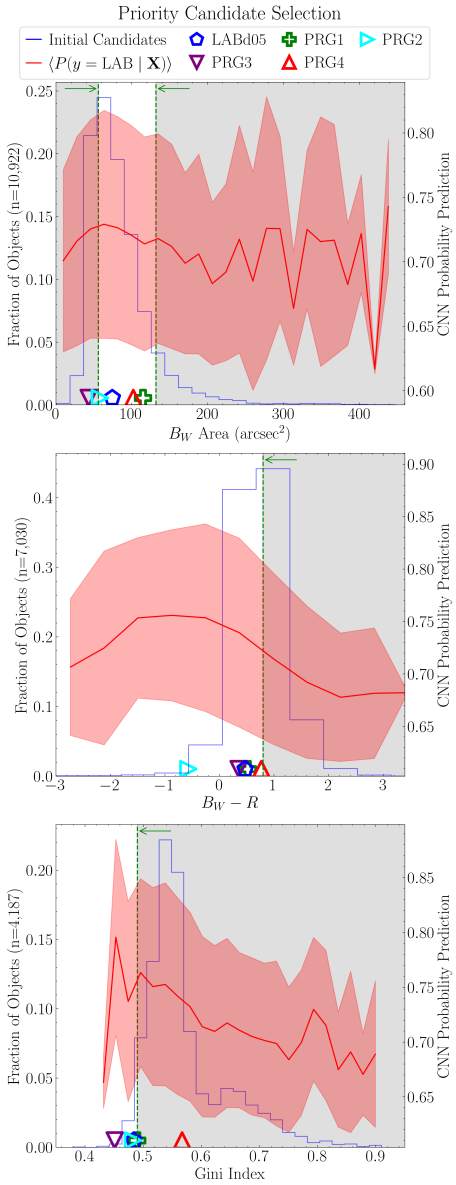

Figure 11. Priority candidate selection. The final LAB candidates are prioritized according to their characteristics with respect to the five known LABs in our sample. Sources are prioritized if they have B_W-based areas between 60 and 132 arcsec^2 (top), followed by a color selection that selects bluer objects with $B_W - R \leq 0.8$ (middle). A final Gini index cut is imposed to select the most diffuse-like candidates with Gini index ≤ 0.49 (bottom).

The top panel shows the initial area-based selection, where we plot the distribution of segmentation areas measured in B_W for the 10,922 final candidates identified by the CNN. The highest probabilities are assigned to sources with areas of $\sim 20 - 100$ arcsec^2, with all five confirmed LABs falling around this region, motivating the use of an area-based cut around it. As the source areas increase to ~ 200 arcsec^2, the bin-averaged probabilities decline. Beyond ~ 200 arcsec^2, the probabilities average out at ~ 0.7 with high variance, as these large-area bins are sparsely populated. As shown in Fig. 8, these bins are dominated by artifacts such as saturated stars and edge effects, which artificially inflate segmentation areas. Visual inspection of the 237 candidates above this threshold confirms that nearly all are outliers. This illustrates a key limitation in that CNN classification remains sensitive to image artifacts. Despite the application of an outlier-removal step, about 10% of artifacts are expected in the final sample (see Fig. 7) which can be output with relatively high probabilities by the CNN model. We therefore impose a refined area cut, selecting only sources with segmentation areas between $60 - 132$ arcsec^2. This reduces the candidate pool to 7,030 objects while preserving all confirmed LABs except PRG3, which had already been excluded due to its low CNN probability (Fig. 9).

The middle panel shows the $B_W - R$ color distribution for the area-selected candidates. The CNN assigns higher probabilities to sources with $B_W - R \lesssim 2$, consistent with expectations that Lyα emission at $z \sim 2 - 3$ produces a blue color excess. Although the predicted probabilities decline for objects with $B_W - R < -1.2$, we note that these bins are sparsely populated, containing only 26 of the 7,030 objects. We therefore refine the sample by selecting candidates with $B_W - R \leq 0.8$, a threshold that includes all five confirmed LABs, including the reddest (PRG4). This cut reduces the candidate pool to 4,187 sources.

The bottom panel presents the Gini index distribution for the area and color-filtered candidates. Consistent with Fig. 10, we find that objects with lower Gini index are generally assigned higher CNN probability predictions, with maximum probabilities occurring near Gini index ~ 0.45. While a sharp decline in probability is observed for Gini index < 0.45, we note that these bins include only 4 of the 4,187 candidates and are not statistically well-sampled. Three of the confirmed LABs fall below Gini index ~ 0.49, motivating a cut at Gini index ≤ 0.49 to preferentially select diffuse, low-concentrating morphologies characteristic of our known LABs. This cut excludes PRG4, which the CNN had assigned relatively high confidence but which had already been removed by `XGBoost-8` due to its low probability prediction. Applying this final Gini index cut yields 155 high-priority candidates, a $\sim 98.5\%$ reduction in the final candidate sample.

To remove duplicate entries, which is especially important as LABs can exhibit group-like morphologies, we eliminate candidates within 10 arcseconds of each other, removing 30 duplicate sources (20%), including four duplicates of PRG2, which were located within $\sim 3 - 5$ arcsec of PRG2's cataloged position. The imaging in Fig. 1 shows the group-like nature of this LAB. We visually inspect the remaining 125 candidates, identifying 15 objects (12%) that we consider outliers as per the criteria discussed in Section 2.4, but that

were not flagged by the `iForest` nor rejected by the CNN. This is consistent with the $\sim$ 90% outlier detection rate the `iForest` achieves with HOG features at the optimal image size (Fig. 7). The remaining 110 candidates represent the final output of our classification pipeline, and are presented in Appendix C. The algorithm design and results are summarized in Table 4.

Fig. 12 presents the multi-band imaging for a subset of the objects that made it through the classification pipeline. Below each image we include a segmentation patch to visualize the spatial extent of the object, extracted at the optimal threshold of $\sigma_{\rm det} = 0.32$ (see Fig. 2). The upper two rows display eight new candidates selected from the full set of 110, ordered by CNN probability prediction (top left to bottom right). The top-left panel shows the top candidate with the highest-confidence probability prediction, while the remaining seven are sampled across the rest of the sorted distribution, thus providing a representative overview spanning the full range of output probabilities. Above each panel, we report the candidate index and its associated probability score. For comparison, in the lower row below the solid line we also include the three known LABs in our training set that made it through the pipeline (PRG1, PRG2, and LABd05), also ordered by CNN probability from left to right. The multi-band broadband imaging reveals that the new candidates exhibit comparable spatial extents, diffuse morphologies, and colors to the known LABs.

Among the 110 high-priority candidates, one object (New Candidate 48, middle-left panel in Fig. 12) was previously targeted for spectroscopic follow-up by M. K. M. Prescott et al. (2013) using the MMT Observatory's Blue Channel spectrograph. This source is the 48th most confident priority candidate output by the CNN. The spectrum reveals extended blue continuum emission but no detectable lines, leaving its redshift and physical nature uncertain. As noted by M. K. M. Prescott et al. (2013), the source may lie within the "redshift desert" ($1.2 \lesssim z \lesssim 1.6$), where Ly$\alpha$ is redshifted below the atmospheric cutoff ($\lambda_{\rm obs} \lesssim 3100$ Å). The absence of emission lines typically seen in low-redshift star-forming galaxies, such as [OII], [OIII], and Hα, is consistent with this scenario. Its diffuse morphology and strong resemblance to confirmed LABs in our training set suggests that it is a strong LAB candidate, potentially at $z < 1.6$. Including this source and the three previously confirmed LABs (shown in the lower panels of Fig. 12), our final sample comprises 113 objects that passed all stages of our pipeline. Of the four with available spectroscopy, three are confirmed LABs, yielding a rough confirmation rate of 75% among the spectroscopically observed sources. Notably, the fourth candidate with spectroscopic follow-up (New Candidate 48) is a strong candidate possibly lying outside the spectral coverage of the available data. This highlights the effectiveness of our pipeline in selecting high-confidence LAB candidates for follow-up observations.

Although our classification engines achieve a 100% success rate among the confirmed LAB sample, the imposed probability cuts removes the two least prominent cases from the final prioritized sample. Given our goal of enabling efficient spectroscopic follow-up, this prioritization is necessary to reduce the number of candidates to a manageable subset. However, we acknowledge that basing prioritization on known LAB properties introduces selection biases. This selection limits our ability to identify compact, redder or less extended LABs.

To enable investigation of all our candidates, we publicly release the full survey catalog containing all $\sim$ 2.4 million sources from the NDWFS Boötes field, which includes the B_W-based morphological features, $B_W - R$ colors, and classification results from all three machine learning models. In addition to the full classification catalog, we also provide a sub-catalog consisting of the high-priority candidates identified in this work, including the imaging cutouts to facilitate rapid inspection and follow-up planning. This release is accompanied by the trained machine learning models and our open-source Python codebase, the Blob Identification Algorithm (`pyBIA`, D. Godines 2025). The `pyBIA` framework is modular and designed to operate on multi-band imaging data, allowing users to reproduce our full analysis while adapting the pipeline to other surveys, filter sets, or related science cases. In addition, we provide a dedicated documentation page which includes application examples including all of the code used in this work, along with the intermediate data products generated throughout the pipeline, ensuring that every step of the analysis can be replicated and extended.

3.2. *Narrowband-Selected LABs*

The goal of our search is to identify the most prominent LABs, i.e., the sources that show diffuse morphologies and blue colors even in broadband imaging. However, it is instructive to evaluate how sensitive our classification pipeline is to less prominent LABs. To do so, we make use of a NB search conducted by Y. Yang et al. (2009), who discovered four LABs in the Boötes field. These four LABs (hereafter referred to as Yang+09 Blobs 1 – 4) were not included in our training set because they were selected based on NB imaging specifically tuned to Lyα emission at $z \sim 2.3$. As such, these do not align with our training set design and our objective of developing a detection algorithm optimized for identifying spatially extended, diffuse, and blue-dominated LABs in broadband imaging.

However, to evaluate how sensitive our classification pipeline is to LABs that might be less prominent in broadband imaging, we passed the Yang+09 LABs through our pipeline. This evaluation provides an independent test of

Step	Description	No. of Candidates
1	Generate a catalog of objects from single-band (B_W) imaging, extracting fluxes and morphological parameters via image segmentation.	2,377,342
2	Remove non-detections (no segmentation object was identified) and sources with NaN B_W magnitudes.	1,968,480
3	Input the extracted parameters into a trained `XGBoost` classifier and retain candidates with probability predictions $P(y = \mathrm{LAB} \mid \mathbf{X}) \geq 0.9$.	53,518
4	Compute multi-band (B_W and R) Histogram of Oriented Gradients (HOG) features and input them into a trained `iForest` model for outlier removal.	47,829
5	Classify the remaining candidates, including the LAB candidate training sample, using a multi-channel convolutional neural network (CNN), trained on B_W and R-band imaging.	17,048
6	Prioritize candidates with CNN probability predictions $P(y = \mathrm{LAB} \mid \mathbf{X}) \geq 0.6$.	10,922
7	Select CNN candidates resembling known LABs by requiring B_W segmentation areas of $60-132$ arcsec^2, blue colors ($B_W - R \leq 0.8$), and Gini indices ≤ 0.49, with thresholds set by the confirmed LAB sample.	155
8	Remove duplicate objects located within 10 arcseconds of another candidate.	125
9	**Visual inspection** to remove remaining objects with outliers in either band.	110

Table 4. Classification pipeline summary of the processed image data from the NDWFS Boötes field. Our algorithm is composed of several machine learning models operating in sequence, followed by physically motivated morphological cuts according to the characteristics of the known LABs in our sample.

whether such NB-selected LABs exhibit similar broadband signatures and would be recovered by our classifiers. The top two rows in Fig. 13 shows B_W imaging (top panels) and segmentation masks at the optimal detection threshold (bottom panels) for the four Yang+09 sources, ordered by decreasing CNN probability prediction from top left to bottom right: Yang+09 Blob 4 (top left), Yang+09 Blob 2 (top right), Yang+09 Blob 1 (bottom left), and Yang+09 Blob 3 (bottom right). The probability predictions assigned by the `XGBoost-8` and CNN models are displayed above each source. We note that all four sources are classified as inliers by the `iForest` model, and the B_W and R-band show none of the aforementioned artifacts described in Section 2.4.

Among these four LABs, Yang+09 Blob 4 exhibits a clumpy, extended morphology in B_W and is output with relatively high confidence by both models, receiving $P(y = \mathrm{LAB} \mid \mathbf{X}) = 0.901$ from `XGBoost-8` and $P(y = \mathrm{LAB} \mid \mathbf{X}) = 0.672$ from the CNN. This source thus made it through our `XGBoost-8` selection threshold of $P(y = \mathrm{LAB} \mid \mathbf{X}) \geq 0.9$ and the imposed CNN selection threshold of $P(y = \mathrm{LAB} \mid \mathbf{X}) \geq 0.6$, and was included in our initial and final candidates. This source, however, was excluded from our priority candidate list due to its high Gini index of 0.587, which exceeds the Gini index cutoff of ≤ 0.49 used in the final prioritization step. With an area in the B_W of 86.9 arcsec^2 and a $B_W - R$ color of 0.35, this LAB would have otherwise made it through our entire pipeline. This highlights the bias introduced during the final prioritization illustrated in Fig. 11, motivating our release of the full candidate catalog so that future studies can examine the complete set of candidates at every stage of the pipeline.

Yang+09 Blob 1 also receives a positive classification from `XGBoost-8` with a probability prediction of $P(y = \mathrm{LAB} \mid \mathbf{X}) = 0.539$, but is rejected by the CNN with $P(y = \mathrm{LAB} \mid \mathbf{X}) = 0.426$. Its lower extent and redder color likely contribute to the reduced probability predictions. Yang+09 Blob 3 was rejected by both the `XGBoost-8` and CNN models, receiving probability predictions of < 0.1 in both cases, consistent with its compact (B_W area of 37.68 arcsec^2), centrally concentrated morphology, and redder broadband color ($B_W - R = 1.45$). Similarly, Yang+09 Blob 2 was rejected with low probability prediction by `XGBoost-8`, although was positively classified by the CNN with a probability prediction of $P(y = \mathrm{LAB} \mid \mathbf{X}) = 0.649$ despite its compact nature and redder color of $B_W - R = 1.13$. The segmentation patch in the lower panel shows that the `XGBoost-8` model cleanly captured the central source without contaminants, suggesting that the higher probability output from the CNN may instead reflect nearby objects that it incorporated into its prediction.

The third row of Fig. 13 places these four NB-selected LABs in context with the five confirmed LABs in our broadband-based training set. The bottom-left panel shows B_W area versus Gini index, while the bottom-right shows $B_W - R$ color versus CNN probability prediction. The scatter points in both plots are scaled by the probability predictions that were output by the `XGBoost-8` model. From the bottom-left panel, it is evident that the two Yang+09 LABs output with the lowest `XGBoost-8` probability prediction (Yang+09 Blobs 2 and 3) occupy regions of high Gini index ($\gtrsim 0.6$) and small B_W area ($\lesssim 40$ arcsec^2). This is in contrast with Yang+09 Blobs 1 and 4 which were positively output by `XGBoost-8` and which are characterized by lower Gini indices, although Yang+09 Blob 1 had a significantly smaller B_W area of 25.89 arcsec^2 (~ 3 times smaller), consistent with its lower prediction than Yang+09 Blob 4 — highlighting the

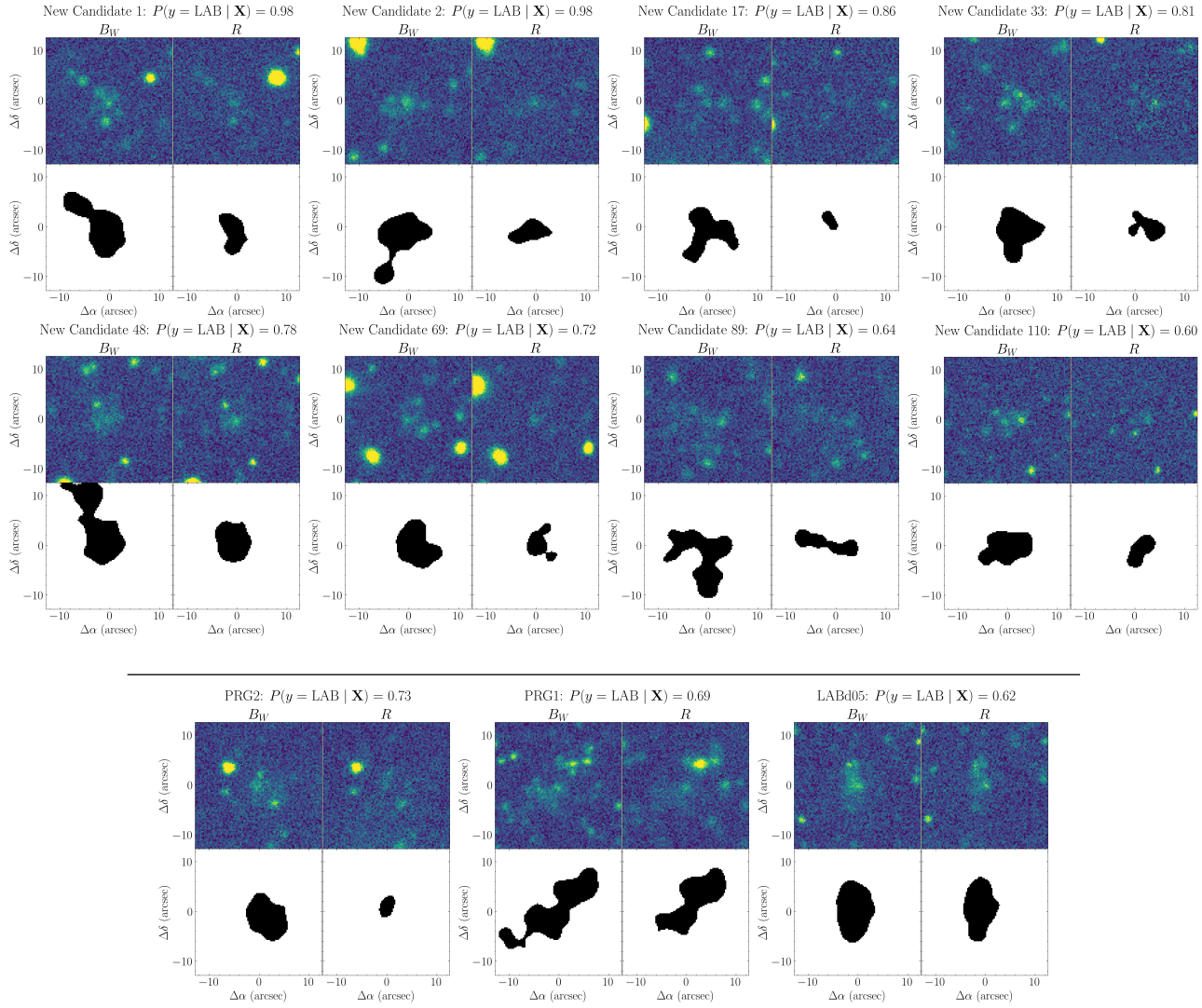

Figure 12. Broadband imaging of eight new candidates discovered by our pipeline (top two rows), and the three known LABs in our sample that were recovered (bottom row). From left to right these subsets are ranked by CNN probability prediction, which is included above each panel. Image segmentation patches are included below the corresponding image data to visualize the source extent at the optimal B_W detection threshold of $\sigma_{\text{det}} = 0.32$.

importance of these features at the earlier stages of classification.

In general, Yang+09 Blob 4 displays a broadband morphology closer to that of the known LABs from M. K. M. Prescott et al. (2012, 2013), with a large B_W area comparable to PRG2 and LABd05 and similar color to LABd05 and PRG1. However, its Gini index value remains relatively high and comparable to PRG4, which was the LAB in our training sample that exhibited the least prominent broadband morphology and notably larger Gini index. This comparison reveals that our broadband selection tends to favor LABs with lower Gini index values and larger apparent spatial extents, indicative of more diffuse and clumpy-like morphologies. LABs that are smaller and more concentrated in broadband images, which NB-based surveys can still efficiently identify, are less likely to be detected by our broadband-based framework.

Despite these selection effects, the recovery of Yang+09 Blob 4 demonstrates that our machine learning classifiers, trained entirely on broadband-selected candidates, are capable of recovering the most prominent narrowband-selected sources. Given the results of this analysis, in total, the models successfully recover six of the nine known LABs in the Boötes field with high probability scores, validating the effectiveness of our approach.

4. DISCUSSION

This work demonstrates that a machine learning-based classification pipeline can successfully identify high-redshift LAB candidates using multi-band broadband imaging, even when constrained by a limited number of positive training

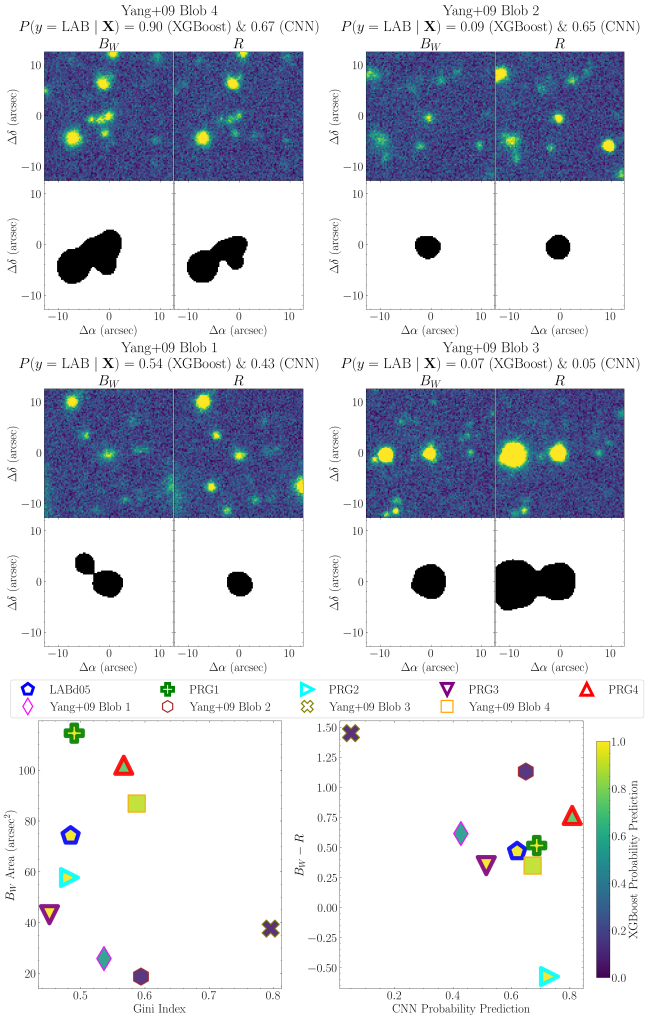

Figure 13. Broadband imaging and morphological properties of the four NB-selected LABs discovered by Y. Yang et al. (2009). From top-left to bottom-right, these are ordered by decreasing CNN probability prediction. The lower-left and lower-right panels compares these LABs to the known broadband-based LABs in our training set. The scatter points are scaled by the `XGBoost-8` probability prediction, and are shown as a function of Gini index, area, color, and CNN prediction. The Yang+09 Blobs are displayed with thinner marker lines than the broadband-selected LABs in our sample.

instances. Unlike prior CNN-based galaxy classification efforts such as Toothless (A. K. Aniyan & K. Thorat 2017) and GAMORNET (A. Ghosh et al. 2020), which relied on large ($\sim 10^4 - 10^5$) simulated training sets, our framework is built on a significantly smaller, real-world dataset. Our multi-stage algorithm first employs a tree-based ensemble model trained on single-band morphological features, followed by a multi-band CNN trained to distinguish the initial candidate set. This is a design choice intended to perform an initial candidate selection using more interpretable models, which work well with limited data and are more computationally efficient. Applying well-informed probability prediction cuts, this step removes the majority of false-positives, after which deep-learning methods such as CNNs further refine the candidate sample. We show that in cases like ours where the positive class is rare, an effective strategy is to randomly sample the negative class from the general survey population and apply rigorous cross-validation strategies such as LOOCV. This approach reduces bias and increases the likelihood of any true positives mistakenly included in the negative class being recovered. We emphasize that the primary goal of this work is to define a scalable, survey-ready framework for identifying broadband-prominent LAB candidates in wide-field surveys. The resulting candidate set is consistent with the broadband selection function of M. K. M. Prescott et al. (2012), as supported by our feature importance analysis which highlighted the importance of spatial extent and concentration-sensitive morphology.

To assess the robustness of our method, we evaluate our pipeline against the total population of nine known LABs in the Boötes field (the five broadband-selected LABs from M. K. M. Prescott et al. 2013 and the four narrowband-selected sources from Y. Yang et al. 2009). The machine learning models successfully classify six of these nine objects (67%) in the early classification stages with high probability scores (Sections 2.3.5 – 2.5.2). This recovery rate suggests that the models successfully learned the general morphological features of LABs, despite training on LAB candidates. The entire pipeline, including the final prioritization, reduces the NDWFS Boötes catalog from $\sim$ 2 million detected sources to 110 high-priority targets (a $\sim$ 99.995% reduction), while retaining three of the five confirmed LABs in our training set in the final prioritized list. The selection is driven by three key morphological characteristics: B_W magnitudes of $\sim 23 - 24$, diffuse surface brightness distributions (Gini index $\lesssim 0.55$), and large spatial extents in the B_W ($\lambda_1 \gtrsim 3$ arcsec^2), with the single-band `XGBoost-8` and the multi-band CNN models learning these trends independently (Fig. 4 and Fig. 10). In practice, this means the pipeline is most sensitive to LABs that look "blob-like" in broadband data (i.e., clumpy, extended, and diffuse).

The explicit prioritization step we apply at the end of our pipeline (area, color, and Gini-based cuts; Fig. 11) refines the candidate sample to those most similar to the confirmed LABs in our training set. Through this final selection we aim to increase the likelihood of a successful spectroscopic follow-up campaign, but it also introduces a selection bias that skews away from compact, centrally concentrated, and/or redder systems. This bias is evident via the NB-selected Yang+09 Blob 4 (Section 3.2; Fig. 13), which is confidently detected as an LAB candidate by all of our machine learning models, but is ultimately removed during the prioritization step due to its relatively high Gini index of $\sim$ 0.59. Our prioritized list of LAB candidates is thus

representative of the diffuse and extended end of the LAB population, and by design, incomplete for compact or AGN-dominated morphologies.

The initial classification stage with the `XGBoost-8` model is also conservative by design (Fig. 5), assigning no predictions with $P(y = \mathrm{LAB} \mid \mathbf{X}) = 1.0$ and very few with $P(y = \mathrm{LAB} \mid \mathbf{X}) \gtrsim 0.98$. This model also yielded better calibrated scores than models trained on larger feature sets, with the mean probability predictions from the LOOCV analysis of ~ 0.9 for the LAB class and ~ 0.1 for the OTHER, consistent with the observed filtering rates. The CNN model, trained with heavy data augmentation, filters out 65% of the remaining candidates, in close agreement with the average probability prediction of $P(y = \mathrm{LAB} \mid \mathbf{X}) \approx 0.4$ among all initial candidates. Its validation and loss scores track the training curves (Fig. 9), indicating limited overfitting despite the small positive class. While data augmentation and a shallow architecture were important design choices, the inclusion of an explicit B_W/R color difference channel was imperative for achieving training/validation convergence. This underscores how domain knowledge can be applied to enhance feature extraction in deep learning algorithms.

Contamination can effect the LAB candidate selection and complicate CNN training especially when dealing with limited training sets. Such contaminations include imaging artifacts such as bleed trails, halos, and edge effects that can yield large segmentation extents and low surface brightness structures. Although our initial single-band engine trained on morphological features may not always filter these out, in Fig. 8 we demonstrate how segmentation-based area cuts can be applied to remove outliers with unreasonably large extents (B_W area $\gtrsim 150$ arcsec^2). More robust, but computationally expensive methods such as HOG and LBP features, can also work well at removing such outliers from imaging datasets. In this work we demonstrate their utility when used to train an `iForest` model, which, when trained on the inlier class, can work well at eliminating the majority of outliers present in the dataset while maintaining inlier detection rates near unity (Fig. 7). To achieve such performance we find that feature extraction at larger image sizes is important, in contrast to the smaller images that work best when training CNNs.

Narrowband surveys targeting Lyα directly, such as the one conducted by Y. Yang et al. (2009) in the Boötes field, can effectively select compact or centrally concentrated halos with indistinguishable broadband morphology. Although our approach enables wide-field sky coverage, it is naturally much less sensitive to such cases, favoring the most extended and diffuse objects that are more prominent in optical broadband imaging. Fig. 13 illustrates well the cases we are most sensitive to, as among the four LABs discovered by Y. Yang et al. (2009), our machine learning models confidently detect only one, Yang+09 Blob 4, the only case with bluer color, large area, and relatively low Gini index in the B_W. Thus, while our broadband-based selection delivers a wide-area list of morphologically distinguishable LABs over a broad comoving volume, and at the fraction of the observational cost of NB programs, our algorithm is biased away from redder LABs that appear smaller and/or more concentrated in the broadband.

The selection of low-Gini, extended morphologies among our priority candidate list is consistent with criteria set by previous successful follow-up campaigns (e.g., M. K. M. Prescott et al. 2013), and is qualitatively consistent with scenarios in which Lyα is distributed by resonant scattering and/or powered by sources in overdense environments. However, the systematic exclusion of high-Gini index cases means that our prioritized list likely under-represents AGN-dominated systems with bright compact cores embedded in lower surface brightness halos. Spectroscopic confirmation of our candidates will nonetheless provide lower limits to the fraction of sky covered by broadband-prominent LABs at the depths, seeing, and filters used in this work.

In future work we will apply our `pyBIA` framework to detect LAB candidates in the broadband Hyper-Suprime Cam Subaru Strategic Program (HSC SSP; G. Hill et al. 2021) data of the COSMOS field (N. Scoville et al. 2007), which will be trained exclusively on confirmed LABs from the NB-based One-hundred-deg^2 DECam Imaging in Narrowbands (ODIN) program (K.-S. Lee et al. 2024). Training sets comprised of true LABs will allow our machine learning models to learn the broader range of morphological characteristics that our current prioritization is not sensitive to, enabling effective detection of LABs dominated by a variety of powering mechanisms at a broad redshift range. Spectroscopic data from HETDEX Public Source Catalog 1 (E. Mentuch Cooper et al. 2023) in the COSMOS field will allow for immediate confirmation of LAB candidates, which will enable space density estimates and redshift distribution analysis. Catalog data from the COSMOS2020 Farmer Catalog (J. R. Weaver et al. 2022; J. R. Weaver et al. 2023) will enhance this analysis, allowing for the systematic study of galaxies within the vicinity of the confirmed LABs. Complementary catalogs will further allow us to enhance our predictions by combining machine learning scores with environment metrics (e.g., star-formation rates, overdensities).

Future improvements to the classification pipeline includes incorporating additional bands, such as the six that LSST will provide (*ugrizy*), to construct color-excess channels tailored to Lyα at broader redshift ranges and further improve filtering efficiency. While we demonstrate the utility of shallow CNN architectures such as AlexNet in the context of limited datasets (70 positive samples), deeper networks and those incorporating residual connections have proven effective in a wide range of astrophysical tasks (X.-P. Zhu et al. 2019;

T. Cecconello et al. 2024) and are worth exploring in this setting. Transfer-learning approaches have also shown substantial success (S. Ackermann et al. 2018; R. Vilalta 2018; J. Schneider et al. 2023), enabling rare-object classification even with limited training data. Transformer-based architectures are also proving powerful for astronomical imaging and classification (J. Yao-Yu Lin et al. 2021; Y. Yang & X. Li 2024; J. Cao et al. 2024), yielding state-of-art performance in a variety of settings.

5. CONCLUSION

LABs are rare, spatially extended emission-line nebulae that trace regions of intense galaxy formation and large-scale structure at high redshift. Their detection in wide-field imaging surveys remains observationally challenging due to their diffuse morphologies and low surface brightness. This work presents a multi-stage machine learning framework capable of identifying LAB candidates directly from broadband imaging, achieving both high recovery of known LABs and a strong reduction in false positives, despite the limited number of confirmed training instances. Our main findings are as follows:

1. We demonstrate that a scalable broadband selection strategy is feasible. Our pipeline consists of a single-band-based `XGBoost` classifier trained on segmentation and photometric features, a HOG-based `iForest` outlier removal step applied to B_W and R-band image cutouts, and a shallow CNN trained on B_W, R, and B_W/R images, reducing the NDWFS Boötes catalog by $\sim 99.5\%$. Final candidates are prioritized for spectroscopic follow-up based on morphological similarity to confirmed LABs, resulting in 110 priority candidates (Table 4, Fig. 15).

2. We demonstrate that tree-based ensemble classifiers consistently outperform other popular supervised learning algorithms, with boosting frameworks providing superior performance (Fig. 2). Through feature selection and hyperparameter optimization, we further show that training with a reduced feature set can produce conservative classifiers that improve overall performance while reducing false positives (Fig. 3, Fig. 5).

3. Explicitly adding a third image channel (B_W/R) stabilizes training and improves performance for a shallow AlexNet-style CNN trained with heavy data augmentation over just six epochs, demonstrating that incorporating astrophysical priors and domain knowledge into the input can enhance classification (Table 3, Fig. 9).

4. Outlier sources from imaging artifacts (e.g., bleed trails and halo/edge effects) can be effectively removed using an `iForest` model trained on HOG or LBP features, while maintaining inlier detection rates near unity (Fig. 7, Fig. 8).

5. The machine learning models successfully recover ($P(y = \mathrm{LAB} \mid \mathbf{X}) \geq 0.5$) all five confirmed LABs in the training set, with three passing the selection thresholds and final prioritization. The selection function is intentionally biased toward large, diffuse, and blue systems with low Gini coefficients and high λ_1 values, i.e., LABs that remain prominent in broadband imaging (Fig. 4, Fig. 10, Fig. 11).

6. Applied to the four narrowband-selected LABs from Y. Yang et al. (2009) in the same field, the machine learning models recover the most broadband-prominent case with high confidence, while down-weighting compact, high-Gini sources (Fig. 13).

The present pipeline demonstrates that broadband imaging alone can deliver a morphologically targeted LAB catalog across wide fields. Despite the limitations of broadband-only analysis, our selection function allows for efficient spectroscopic follow-up of LAB candidates, which in turn will enhance our understanding of the demographics and power sources of these rare objects. To support such efforts, we have made our Boötes LAB candidate catalog publicly available together with the software developed in this work, `pyBIA` (Blob Identification Algorithm), which is published as an open-source Python package and distributed via `pip` (D. Godines 2025). We also provide all trained machine learning models and the scripts required to reproduce every aspect of this work, along with the complete NDWFS Boötes classification catalog (including training features and model scores), and the high-priority candidate list with imaging cutouts for immediate use in spectroscopic follow-up planning. Our open-source API offers a full framework for feature engineering and machine learning model training, enabling replication of this analysis and extension to other datasets, with the long-term goal of establishing a robust, general-purpose LAB detection framework for next-generation extragalactic surveys such as the LSST.

6. ACKNOWLEDGMENTS

We thank John Cutsinger and Huiping Cao for valuable discussions. This material is based upon work supported by the National Science Foundation under grants No. AAG-1813016 and No. AAG-2407612.

This work utilized resources from the New Mexico State University High Performance Computing Group, which is directly supported by the National Science Foundation (OAC-2019000), the Student Technology Advisory Committee, and New Mexico State University and benefits from inclusion in

various grants (DoD ARO-W911NF1810454; NSF EPSCoR OIA-1757207; Partnership for the Advancement of Cancer Research, supported in part by NCI grants U54 CA132383 (NMSU).

APPENDIX

A. CLASSIFICATION USING ONLY MORPHOLOGICAL DESCRIPTORS

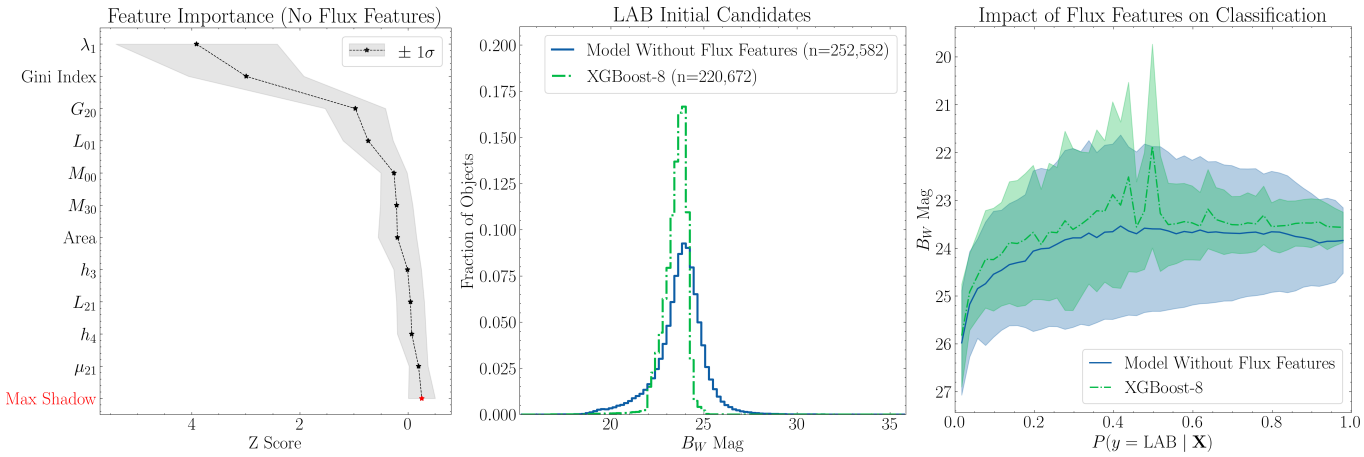

Figure 14. Model trained on morphological features only. The left panel presents the feature importance results when flux features (B_W Mag, B_W MagErr, Min Value, and Max Value) are removed from the feature matrix. The middle panel shows the B_W Mag distributions of the positive candidates output by the model, while the right panel shows the bin-averaged B_W Mag as a function of probability prediction for the full NDWFS Boötes catalog. The same results from our adopted `XGBoost-8` model are also shown for comparison.

The `XGBoost-8` classifier adopted for our initial filter incorporates both photometric and segmentation-derived morphological features, with the B_W magnitude ranking as the most important feature (Fig. 3). To isolate the effects of excluding photometric information, we trained an additional `XGBoost` model after removing flux features from the feature matrix (i.e., B_W Mag and MagErr, as well as the Min and Max pixel values within the segmentation patch). The remaining set of 62 features thus consists exclusively of morphological descriptors. This model underwent the same feature selection and hyperparameter optimization routine used for `XGBoost-8` (see Section 2.3.4).

The feature selection results for this morphology-only model are presented in the left panel of Fig. 14. Eleven features were selected with high importance, with λ_1 and the Gini Index identified as the two most important. Notably, these are the same morphological features prioritized by `XGBoost-8` after the B_W magnitude. This demonstrates that the extended and diffuse morphological descriptors in our feature set are strongly discriminative for identifying the LAB candidates. Even in the absence of explicit flux information, the model achieves a 10-fold cross-validated F1 score of 0.9034, compared to 0.9336 for `XGBoost-8`.

When applied to the full catalog of $\sim$ 2 million sources, the morphology-only model flags 252,582 LAB candidates, whereas `XGBoost-8` flags 220,672. The candidate magnitude distribution is shown in the middle panel of Fig. 14 alongside the candidates output by `XGBoost-8`. The distribution from the morphology-only model shifts toward the fainter end and exhibits prominent bright and faint wings, with sources extending toward the limiting magnitude of the survey ($B_W \sim 27$). In contrast, `XGBoost-8` yields a narrower distribution with nearly all candidates contained within $22 \lesssim B_W \lesssim 25$.

The right panel of Fig. 14 presents the bin-averaged B_W magnitude as a function of probability prediction for both models. This illustrates that, on average, the exclusion of photometric features yields candidates that are $\sim$ 0.3 mag fainter, consistent with a reduced capacity to filter out the dominant faint-end of the survey population. However, the similar classification behavior reveals that the morphology-only model can still effectively flag the majority of extreme faint-end sources as negatives, assigning the lowest probabilities to objects near the survey limit. These results confirm that while the morphology-only model successfully identifies the target population, the inclusion of explicit photometric information in `XGBoost-8` provides a more efficient selection boundary against survey contaminants, especially those at the faint end of the survey population.

B. CODE QUICK-START GUIDE

This appendix summarizes the `pyBIA` codebase (D. Godines 2025) through simplified examples of the core modules. The listings show how imaging data are used to generate source catalogs and to train machine learning models for classification and anomaly detection. `pyBIA` is distributed on PyPI (`pip install pyBIA`), with online documentation and a full API reference.[3]

B.1. *Morphological Catalog Generation*

The `pyBIA.catalog` module constructs photometric and segmentation-based morphological catalogs from single-band imaging. Catalog generation can be run in either targeted mode, where source centroids are provided, or detection mode, where sources are identified automatically when no coordinates are supplied. Listing 1 demonstrates initializing a `Catalog` with a 2D image (`data`) and uncertainty map (`error`), enabling background subtraction (`bkg=None`), and computing the features in Table 1.

```python
from pyBIA import catalog

# Initialize the Catalog object
cat = catalog.Catalog(
    data=data,                  # 2D image (single band)
    error=error,                # 2D uncertainty map (same shape as data)
    bkg=None,                   # Background estimation; set to 0.0 for no subtraction
    exptime=30.0,               # Exposure time (s)
    x=[50, 612], y=[400, 80],   # Source centroid(s) (pix); set to None to detect source(s)
    zp=26.23, nsig=0.35,        # Photometric zero point and segmentation threshold
    obj_name=['Obj_A', 'Obj_B'] # Object name(s)
)

# Compute features and save catalog
cat.create(save_file=True, filename='my_catalog.csv')
```

Listing 1. Example usage of `pyBIA.catalog` for catalog construction from single-band imaging.

B.2. *Supervised Learning*

Listing 2 demonstrates the `pyBIA.ensemble_model` interface for supervised classifier training. The `Classifier` supports optional missing-data imputation and can be run with default hyperparameters or with automated tuning. When enabled, tuning applies `BorutaSHAP` feature selection followed by `Optuna` hyperparameter optimization using a cross-validated objective metric.

```python
from pyBIA import ensemble_model

# Initialize the Classifier with features (data_x) and labels (data_y)
model = ensemble_model.Classifier(
    data_x=data_x,              # Feature matrix of shape (n_samples, n_features)
    data_y=data_y,              # 1D array of labels aligned to data_x
    clf='xgb',                  # The classification model to train
    impute=True,                # Whether to impute missing feature values
    optimize=True,              # Enables automated feature selection & hyperparameter tuning
    n_iter=50,                  # Number of Optuna trials; set to 0 to skip
    scoring_metric='f1',        # Optuna objective (e.g., 'f1', 'precision', 'roc_auc')
    opt_cv=10,                  # Number of CV folds to use during Optuna optimization
    boruta_trials=100,          # Number of BorutaSHAP trials; set to 0 to skip
    boruta_model='rf'           # Base estimator for BorutaSHAP feature ranking
)

# Run the optimization/training pipeline and save
model.create()
model.save(dirname='optimized_xgboost')
```

Listing 2. Training a supervised classifier with optional imputation and automated feature selection and hyperparameter tuning.

[3] https://pybia.readthedocs.io/

B.3. *Unsupervised Anomaly Detection*

Listing 3 demonstrates use of the `pyBIA.outlier_detection` module for unsupervised anomaly detection with an Isolation Forest. The `Classifier` optionally normalizes input cutouts and imputes missing feature values, computes a user-selected set of image descriptors (see Section 2.4), and fits an Isolation Forest on the resulting feature matrix. The trained model can then be applied to candidate cutouts to return outlier labels and anomaly scores.

```python
from pyBIA import outlier_detection

# Initialize the Classifier with single-class data
model = outlier_detection.Classifier(
    data=data,                  # Single-class image cutouts, shape: (N, H, W, C)
    img_num_channels=1,         # Set to C for multi-channel inputs
    feat_set='hog',             # Feature set to use
    clf='iforest',              # The unsupervised engine to train
    normalize=True,             # Whether to min-max normalize prior to feature extraction
    min_pixel=0, max_pixel=10,  # Min and max pixel values for normalization
    impute=True                 # Whether to impute missing feature values
)

# Run the training pipeline and save
model.create()
model.save(dirname='outlier_model')
```

Listing 3. Training an Isolation Forest for unsupervised anomaly detection with optional normalization and imputation.

B.4. *CNN Classification*

Listing 4 provides an overview of the `pyBIA.cnn_model` module for image-based binary classification using convolutional neural networks. The `Classifier` accepts positive and negative-class cutouts (single or multi-channel), and optionally applies per-channel min-max normalization and data augmentation. When validation data is provided, the model can be trained using cross-validation, yielding an ensemble of fold-specific networks.

```python
from pyBIA import cnn_model

# Initialize the Classifier with optional validation data
model = cnn_model.Classifier(
    positive_class=pos_class,   # Positive class cutouts, shape: (N, H, W) or (N, H, W, C)
    negative_class=neg_class,   # Negative class cutouts
    val_positive=val_pos,       # Positive validation cutouts
    val_negative=val_neg,       # Negative validation cutouts
    img_num_channels=1,         # Set to C for multi-channel inputs
    clf='alexnet',              # Model to train
    normalize=True,             # Whether to apply per-channel min-max normalization
    min_pixel=0,                # Minimum pixel for normalization
    max_pixel=100,              # Maximum pixel for normalization (or list for multi-channel)
    augment_data=True,          # Whether to augment the training data
    batch_positive=10,          # Number of augmentations per positive instance
    batch_negative=0,           # Number of augmentations per negative instance
    epochs=10, batch_size=16,   # Training epochs and mini-batch size
    optimizer='sgd', lr=1e-4,   # Optimizer and learning rate
    opt_cv=5,                   # Cross-validation folds; set to None for a single model
    activation_conv='relu',     # Activation function for the Conv2D layers
    activation_dense='relu'     # Activation function for the fully connected (dense) layers
)

# Run the data augmentation/training pipeline and save
model.create()
model.save(dirname='alexnet_model')
```

Listing 4. Training a CNN classifier with optional normalization, augmentation, and cross-validation.

C. LAB CANDIDATES

In this appendix we provide a complete list of the 110 new LAB candidates identified in our analysis and prioritized for a spectroscopic follow-up campaign. Table 5 summarizes the catalog properties, including positions, CNN classification probabilities, and photometry. Fig. 15 shows B_W (top panel) and R-band (lower panel) image cutouts for each candidate, centered on the cataloged position, with each image spanning 100×100 pixels (26×26 arcsec). The intensity scaling of each cutout was adjusted independently to optimize the visibility of source morphology. The ordering of candidates in the figures matches that of the tables. Among these, we note that NDWFSJ143128.2+352656 (number 48) was followed up by M. K. M. Prescott et al. (2013). As discussed in Section 3, the corresponding spectrum for this object shows extended blue continuum but no emission lines, making confirmation difficult. However, it is consistent with this source being in the "redshift desert" ($1.2 \lesssim z \lesssim 1.6$). The lack of [OII], [OIII], and Hα lines and its diffuse, LAB-like morphology suggest it could be a strong LAB candidate at $z < 1.6$.

	Candidate Name (NDWFS Catalog)	RA (J2000 hr)	Dec (J2000 deg)	CNN Proba	B_W Mag (Vega)	$B_W - R$ (Vega)
(1)	NDWFSJ142758.6+354429	14:27:58.591	35:44:28.716	0.98	23.61 ± 0.03	0.39
(2)	NDWFSJ142745.1+352949	14:27:45.129	35:29:49.251	0.98	23.52 ± 0.03	0.16
(3)	NDWFSJ142729.8+341124	14:27:29.884	34:11:24.282	0.94	23.49 ± 0.05	0.31
(4)	NDWFSJ143011.5+345736	14:30:11.531	34:57:36.589	0.92	23.72 ± 0.04	0.57
(5)	NDWFSJ143000.5+354657	14:30:00.559	35:46:57.104	0.90	24.06 ± 0.06	0.43
(6)	NDWFSJ142825.1+345947	14:28:25.130	34:59:47.274	0.90	23.88 ± 0.05	0.23
(7)	NDWFSJ142740.9+352404	14:27:40.792	35:24:01.820	0.89	23.69 ± 0.04	0.65
(8)	NDWFSJ142658.1+340518	14:26:58.220	34:05:18.434	0.89	23.66 ± 0.03	0.58
(9)	NDWFSJ142941.9+344534	14:29:41.890	34:45:34.125	0.89	23.40 ± 0.03	0.00
(10)	NDWFSJ143011.6+340728	14:30:11.625	34:07:27.952	0.89	23.94 ± 0.05	-0.53
(11)	NDWFSJ142821.4+352001	14:28:21.409	35:20:01.680	0.87	23.64 ± 0.04	0.20
(12)	NDWFSJ142852.5+345741	14:28:52.501	34:57:41.458	0.87	23.40 ± 0.04	0.74
(13)	NDWFSJ142558.9+342457	14:25:58.911	34:24:56.755	0.86	24.04 ± 0.05	0.18
(14)	NDWFSJ142925.7+352800	14:29:25.672	35:27:59.469	0.86	23.98 ± 0.05	0.51
(15)	NDWFSJ142812.5+351213	14:28:12.563	35:12:14.079	0.86	23.99 ± 0.05	0.52
(16)	NDWFSJ142700.8+345556	14:27:00.834	34:55:56.673	0.86	23.78 ± 0.05	0.01
(17)	NDWFSJ142903.2+350646	14:29:03.249	35:06:45.987	0.86	23.94 ± 0.05	0.78
(18)	NDWFSJ142746.7+350130	14:27:46.731	35:01:30.891	0.85	23.57 ± 0.04	0.58
(19)	NDWFSJ142833.6+345215	14:28:33.636	34:52:15.506	0.84	23.88 ± 0.05	0.19
(20)	NDWFSJ142729.7+350325	14:27:29.764	35:03:25.667	0.83	24.04 ± 0.07	0.77
(21)	NDWFSJ142803.4+351127	14:28:03.404	35:11:27.566	0.83	24.01 ± 0.05	-0.18
(22)	NDWFSJ142637.4+340855	14:26:37.430	34:08:55.395	0.83	24.00 ± 0.05	0.79
(23)	NDWFSJ143304.3+343357	14:33:04.362	34:33:57.321	0.82	23.81 ± 0.06	-0.19
(24)	NDWFSJ142841.0+344940	14:28:41.036	34:49:40.484	0.82	23.73 ± 0.04	0.44
(25)	NDWFSJ142958.7+352121	14:29:58.736	35:21:21.145	0.82	23.74 ± 0.04	0.54
(26)	NDWFSJ142930.6+353817	14:29:30.629	35:38:17.978	0.82	23.92 ± 0.04	0.68
(27)	NDWFSJ142450.3+341311	14:24:50.341	34:13:11.709	0.82	23.68 ± 0.04	0.58
(28)	NDWFSJ142935.1+344434	14:29:35.116	34:44:34.022	0.82	24.04 ± 0.05	0.45
(29)	NDWFSJ142938.9+350448	14:29:38.982	35:04:48.546	0.82	23.74 ± 0.04	0.25
(30)	NDWFSJ143226.9+335300	14:32:27.010	33:53:00.664	0.81	23.79 ± 0.06	0.06
(31)	NDWFSJ142825.2+352239	14:28:25.213	35:22:39.175	0.81	23.48 ± 0.03	0.63
(32)	NDWFSJ142812.9+345758	14:28:12.892	34:57:58.546	0.81	23.62 ± 0.04	0.74
(33)	NDWFSJ142603.8+343211	14:26:03.872	34:32:11.055	0.81	23.96 ± 0.05	0.67
(34)	NDWFSJ142750.5+345724	14:27:50.489	34:57:24.791	0.81	23.41 ± 0.03	0.43
(35)	NDWFSJ142651.2+334741	14:26:51.260	33:47:40.970	0.80	23.94 ± 0.05	0.27
(36)	NDWFSJ142957.6+345732	14:29:57.640	34:57:33.147	0.79	23.76 ± 0.04	0.56
(37)	NDWFSJ142808.9+353455	14:28:09.003	35:34:55.300	0.79	23.59 ± 0.04	0.29
(38)	NDWFSJ142739.6+351303	14:27:39.607	35:13:02.982	0.79	23.55 ± 0.04	0.68
(39)	NDWFSJ142850.5+345545	14:28:50.541	34:55:45.699	0.79	23.84 ± 0.05	0.50
(40)	NDWFSJ142807.4+353518	14:28:07.483	35:35:19.094	0.79	23.84 ± 0.05	-1.19
(41)	NDWFSJ142502.9+341907	14:25:02.964	34:19:07.082	0.78	23.74 ± 0.04	0.74
(42)	NDWFSJ142518.9+342813	14:25:18.881	34:28:13.696	0.78	23.77 ± 0.04	0.49
(43)	NDWFSJ142929.2+354712	14:29:29.248	35:47:12.612	0.78	23.58 ± 0.04	0.41
(44)	NDWFSJ142747.8+354156	14:27:47.857	35:41:57.004	0.78	24.22 ± 0.07	0.30
(45)	NDWFSJ142529.9+350219	14:25:29.941	35:02:19.888	0.78	23.75 ± 0.05	0.66
(46)	NDWFSJ142743.5+354150	14:27:43.523	35:41:50.234	0.78	24.16 ± 0.06	0.17
(47)	NDWFSJ142642.1+351356	14:26:42.179	35:13:56.693	0.78	23.77 ± 0.05	0.76

Table 5. Catalog properties of the high-priority LAB candidates in the Boötes field. The ordering matches the candidate cutouts in Fig. 15.

	Candidate Name (NDWFS Catalog)	RA (J2000 hr)	Dec (J2000 deg)	CNN Proba	B_W Mag (Vega)	$B_W - R$ (Vega)
(48)	NDWFSJ143128.2+352656	14:31:28.201	35:26:56.851	0.78	23.31 ± 0.03	0.61
(49)	NDWFSJ142851.9+351423	14:28:51.916	35:14:23.746	0.78	24.22 ± 0.08	-0.22
(50)	NDWFSJ142852.6+354725	14:28:52.595	35:47:24.508	0.77	23.85 ± 0.05	0.69
(51)	NDWFSJ142726.5+343735	14:27:26.543	34:37:36.447	0.77	23.90 ± 0.06	0.44
(52)	NDWFSJ143002.6+340905	14:30:02.568	34:09:05.446	0.77	23.93 ± 0.05	0.72
(53)	NDWFSJ143008.9+352218	14:30:08.940	35:22:17.751	0.76	23.74 ± 0.04	0.17
(54)	NDWFSJ142943.6+345707	14:29:43.656	34:57:07.674	0.76	23.73 ± 0.04	0.39
(55)	NDWFSJ142622.2+341423	14:26:22.203	34:14:23.777	0.76	23.96 ± 0.04	0.68
(56)	NDWFSJ142722.3+345222	14:27:22.345	34:52:22.720	0.76	23.86 ± 0.05	0.34
(57)	NDWFSJ142451.6+334800	14:24:51.676	33:48:00.700	0.76	23.98 ± 0.06	0.11
(58)	NDWFSJ143236.7+352726	14:32:36.746	35:27:26.623	0.76	23.99 ± 0.06	0.43
(59)	NDWFSJ142920.2+344837	14:29:20.352	34:48:37.906	0.75	23.74 ± 0.04	0.68
(60)	NDWFSJ142442.8+335753	14:24:42.891	33:57:53.046	0.75	23.98 ± 0.06	0.78
(61)	NDWFSJ142751.3+353042	14:27:51.334	35:30:42.680	0.75	23.93 ± 0.05	0.61
(62)	NDWFSJ143018.9+345643	14:30:18.957	34:56:44.127	0.74	23.77 ± 0.05	0.32
(63)	NDWFSJ142559.1+345406	14:25:59.142	34:54:06.760	0.74	23.97 ± 0.06	0.78
(64)	NDWFSJ142902.7+353916	14:29:02.773	35:39:15.882	0.74	23.99 ± 0.05	0.58
(65)	NDWFSJ142747.6+350112	14:27:47.608	35:01:12.435	0.73	23.81 ± 0.05	0.56
(66)	NDWFSJ142822.7+352250	14:28:22.800	35:22:50.505	0.73	24.21 ± 0.07	0.28
(67)	NDWFSJ142632.8+324551	14:26:32.808	32:45:51.627	0.73	24.09 ± 0.07	0.09
(68)	NDWFSJ142806.1+354751	14:28:06.095	35:47:51.269	0.73	24.08 ± 0.05	-0.24
(69)	NDWFSJ142628.0+350543	14:26:28.065	35:05:42.973	0.72	23.79 ± 0.04	0.63
(70)	NDWFSJ142624.7+341439	14:26:24.784	34:14:40.478	0.71	23.92 ± 0.04	0.34
(71)	NDWFSJ143239.8+351446	14:32:39.805	35:14:46.255	0.70	23.76 ± 0.15	0.64
(72)	NDWFSJ142728.4+352208	14:27:28.515	35:22:08.957	0.70	23.98 ± 0.07	0.23
(73)	NDWFSJ142734.1+343748	14:27:34.160	34:37:48.609	0.70	23.99 ± 0.06	0.23
(74)	NDWFSJ142802.5+351048	14:28:02.573	35:10:48.517	0.69	23.85 ± 0.04	0.66
(75)	NDWFSJ143042.7+345443	14:30:42.804	34:54:43.010	0.69	23.89 ± 0.05	0.59
(76)	NDWFSJ143706.7+335653	14:37:06.723	33:56:53.550	0.68	23.61 ± 0.05	0.75
(77)	NDWFSJ142552.4+331628	14:25:52.420	33:16:28.594	0.68	23.97 ± 0.06	0.20
(78)	NDWFSJ142825.0+352613	14:28:25.054	35:26:13.462	0.68	23.98 ± 0.05	0.51
(79)	NDWFSJ142733.6+352002	14:27:33.669	35:20:02.803	0.67	23.70 ± 0.04	0.63
(80)	NDWFSJ142617.9+341206	14:26:17.909	34:12:06.816	0.67	24.01 ± 0.05	0.22
(81)	NDWFSJ142650.5+342005	14:26:50.592	34:20:05.180	0.67	24.01 ± 0.05	0.67
(82)	NDWFSJ143114.9+354146	14:31:14.982	35:41:46.967	0.66	23.51 ± 0.04	0.54
(83)	NDWFSJ143011.2+333627	14:30:11.288	33:36:27.629	0.66	23.54 ± 0.05	0.61
(84)	NDWFSJ142615.9+323232	14:26:15.955	32:32:31.780	0.66	24.18 ± 0.07	0.74
(85)	NDWFSJ142835.5+344309	14:28:35.485	34:43:10.103	0.66	23.94 ± 0.05	0.33
(86)	NDWFSJ142812.2+332323	14:28:12.290	33:23:24.440	0.66	23.37 ± 0.03	0.53
(87)	NDWFSJ142953.2+345024	14:29:53.234	34:50:24.631	0.65	24.01 ± 0.05	0.51
(88)	NDWFSJ142924.8+351434	14:29:24.861	35:14:34.487	0.65	24.09 ± 0.06	0.52
(89)	NDWFSJ142803.6+352451	14:28:03.680	35:24:51.744	0.64	24.22 ± 0.07	0.61
(90)	NDWFSJ142625.0+333328	14:26:25.069	33:33:28.533	0.64	24.12 ± 0.06	-0.11
(91)	NDWFSJ142842.8+334255	14:28:42.839	33:42:54.476	0.64	23.18 ± 0.03	0.51
(92)	NDWFSJ143145.9+333710	14:31:45.967	33:37:10.291	0.64	23.84 ± 0.08	0.12
(93)	NDWFSJ142811.6+344327	14:28:11.620	34:43:27.775	0.63	23.92 ± 0.05	0.58
(94)	NDWFSJ142935.2+344001	14:29:35.274	34:40:01.092	0.63	23.98 ± 0.05	0.03

Table 5. (Continued)

	Candidate Name (NDWFS Catalog)	RA (J2000 hr)	Dec (J2000 deg)	CNN Proba	B_W Mag (Vega)	$B_W - R$ (Vega)
(95)	NDWFSJ142648.4+344631	14:26:48.449	34:46:31.338	0.63	23.85 ± 0.05	0.77
(96)	NDWFSJ142845.1+345607	14:28:45.093	34:56:07.332	0.63	23.83 ± 0.05	0.70
(97)	NDWFSJ142729.2+351332	14:27:29.225	35:13:33.000	0.63	23.84 ± 0.11	-0.98
(98)	NDWFSJ142814.8+353341	14:28:14.852	35:33:41.228	0.63	23.73 ± 0.04	0.18
(99)	NDWFSJ143119.5+340811	14:31:19.559	34:08:11.473	0.63	23.99 ± 0.06	0.71
(100)	NDWFSJ142750.7+351944	14:27:50.735	35:19:45.002	0.63	24.05 ± 0.06	0.58
(101)	NDWFSJ142806.8+330027	14:28:06.878	33:00:27.686	0.62	23.61 ± 0.04	0.74
(102)	NDWFSJ143015.9+350928	14:30:15.950	35:09:28.266	0.61	22.81 ± 0.00	0.52
(103)	NDWFSJ143056.0+340502	14:30:56.059	34:05:02.189	0.61	23.70 ± 0.05	0.33
(104)	NDWFSJ142838.2+353905	14:28:38.234	35:39:05.930	0.61	24.15 ± 0.06	0.48
(105)	NDWFSJ143045.0+325345	14:30:45.088	32:53:45.641	0.60	23.78 ± 0.07	0.67
(106)	NDWFSJ143056.9+340229	14:30:56.986	34:02:30.004	0.60	24.13 ± 0.08	0.69
(107)	NDWFSJ142812.7+352830	14:28:12.918	35:28:31.299	0.60	23.93 ± 0.05	0.45
(108)	NDWFSJ142855.1+353017	14:28:55.161	35:30:19.059	0.60	23.82 ± 0.05	0.49
(109)	NDWFSJ142757.1+352934	14:27:57.178	35:29:34.329	0.60	24.17 ± 0.06	0.55
(110)	NDWFSJ142505.3+340106	14:25:05.382	34:01:05.998	0.60	23.61 ± 0.04	0.70

Table 5. (Continued)

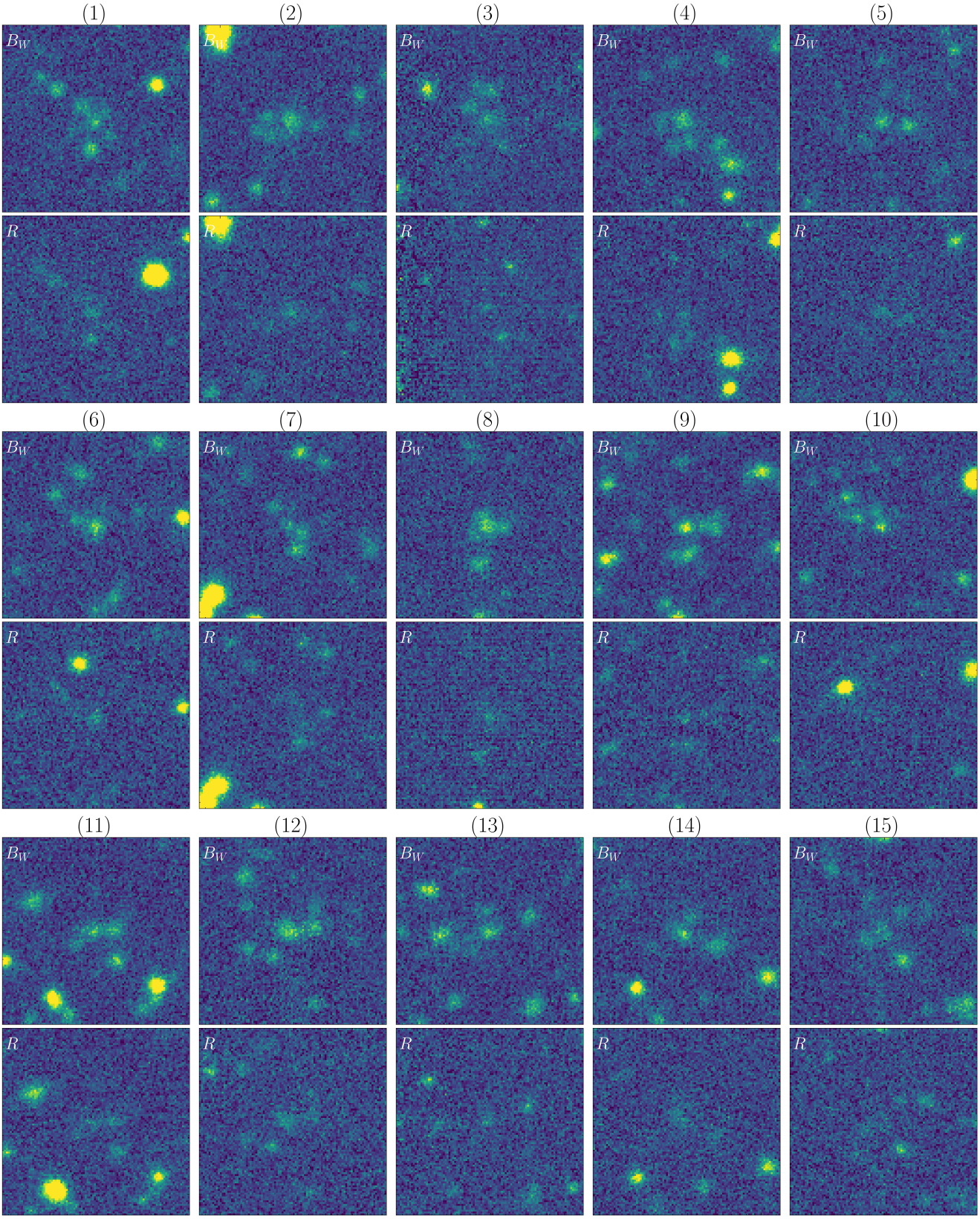

Figure 15. Priority LAB candidates selected by our algorithm, organized according to the CNN probability prediction, from top left to bottom right. The number above each panel corresponds to the ordering in Table 5. The images are 100×100 pixel cutouts (26×26 arcsec), centered on each cataloged source.

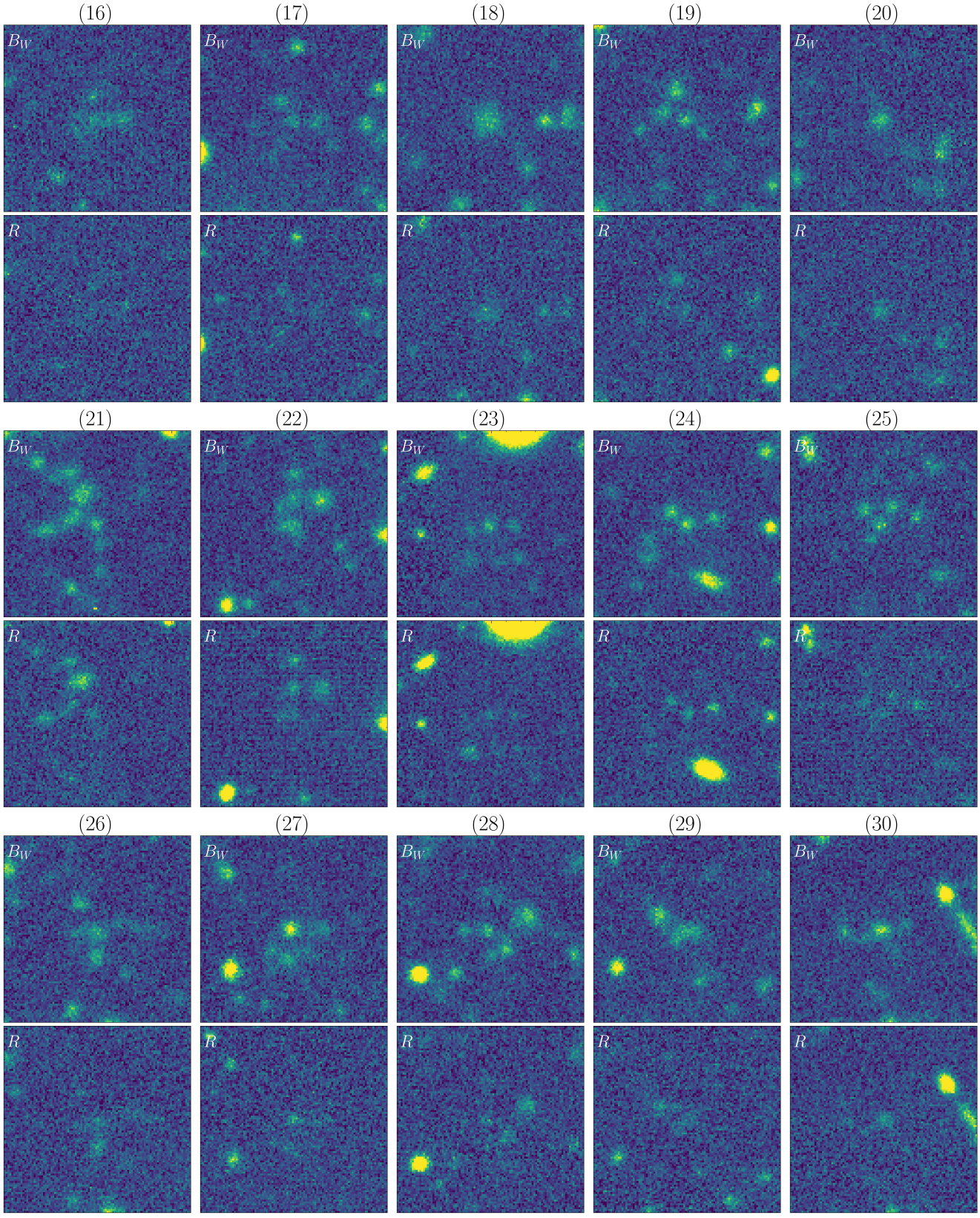

Figure 15. (Continued)

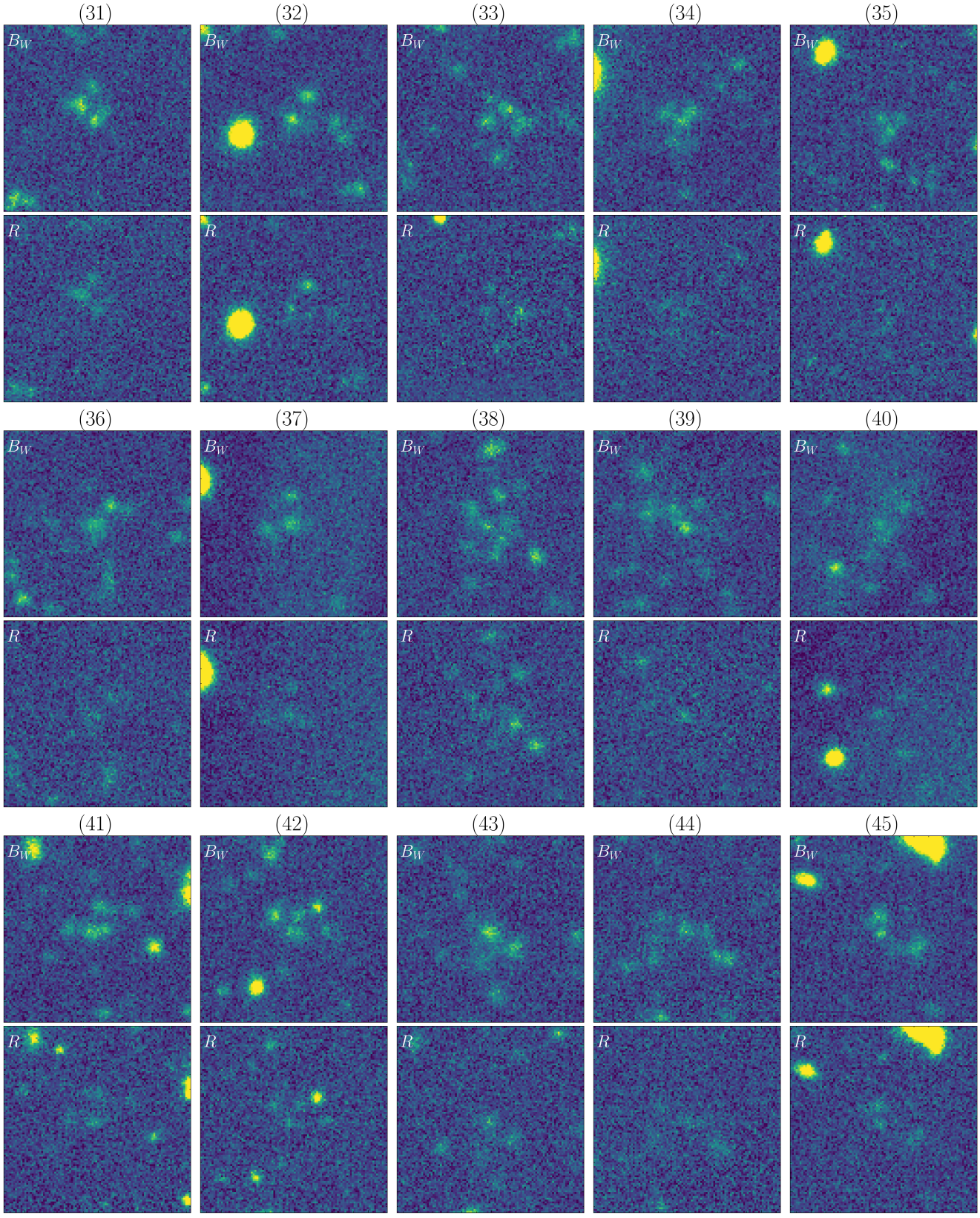

Figure 15. (Continued)

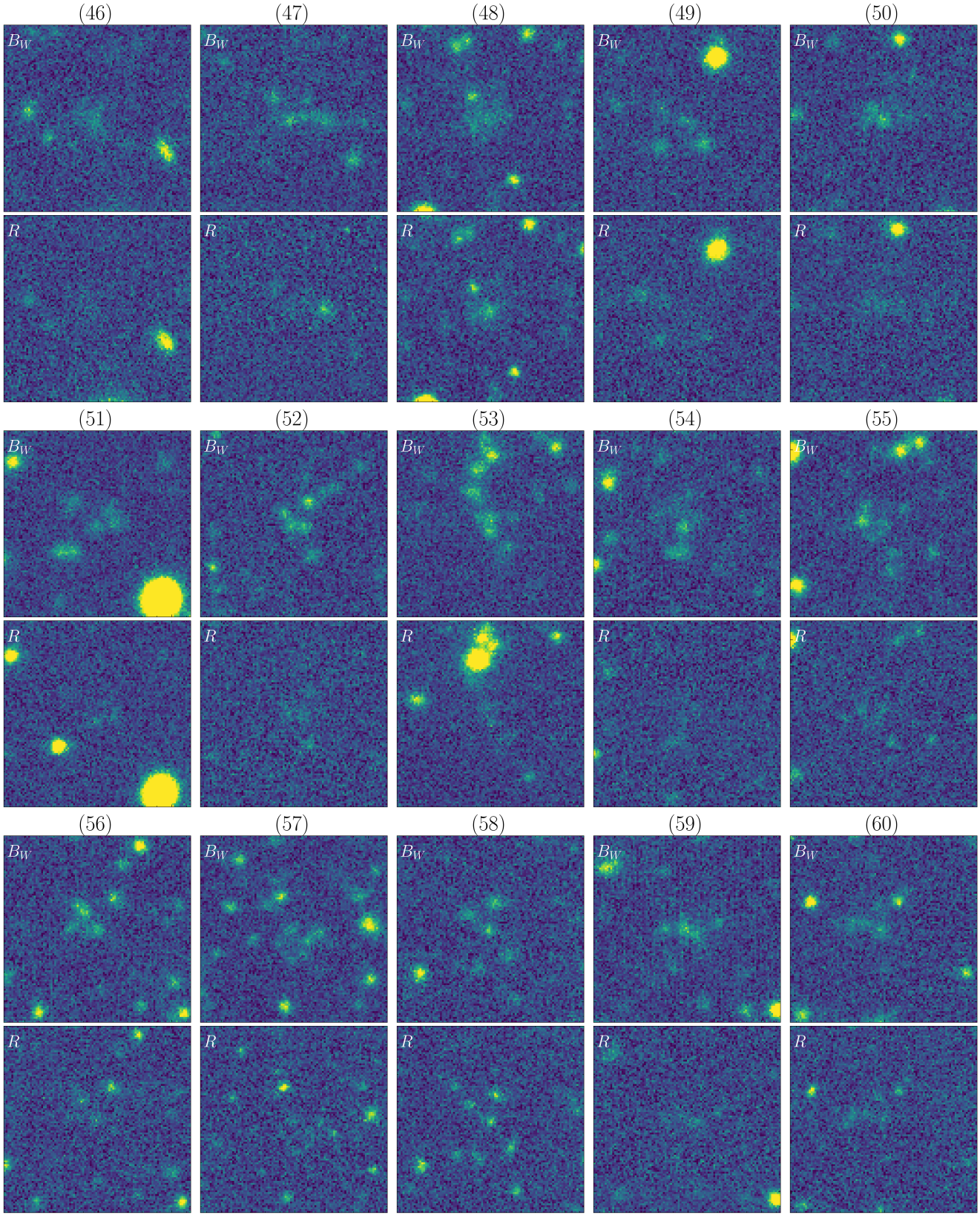

Figure 15. (Continued)

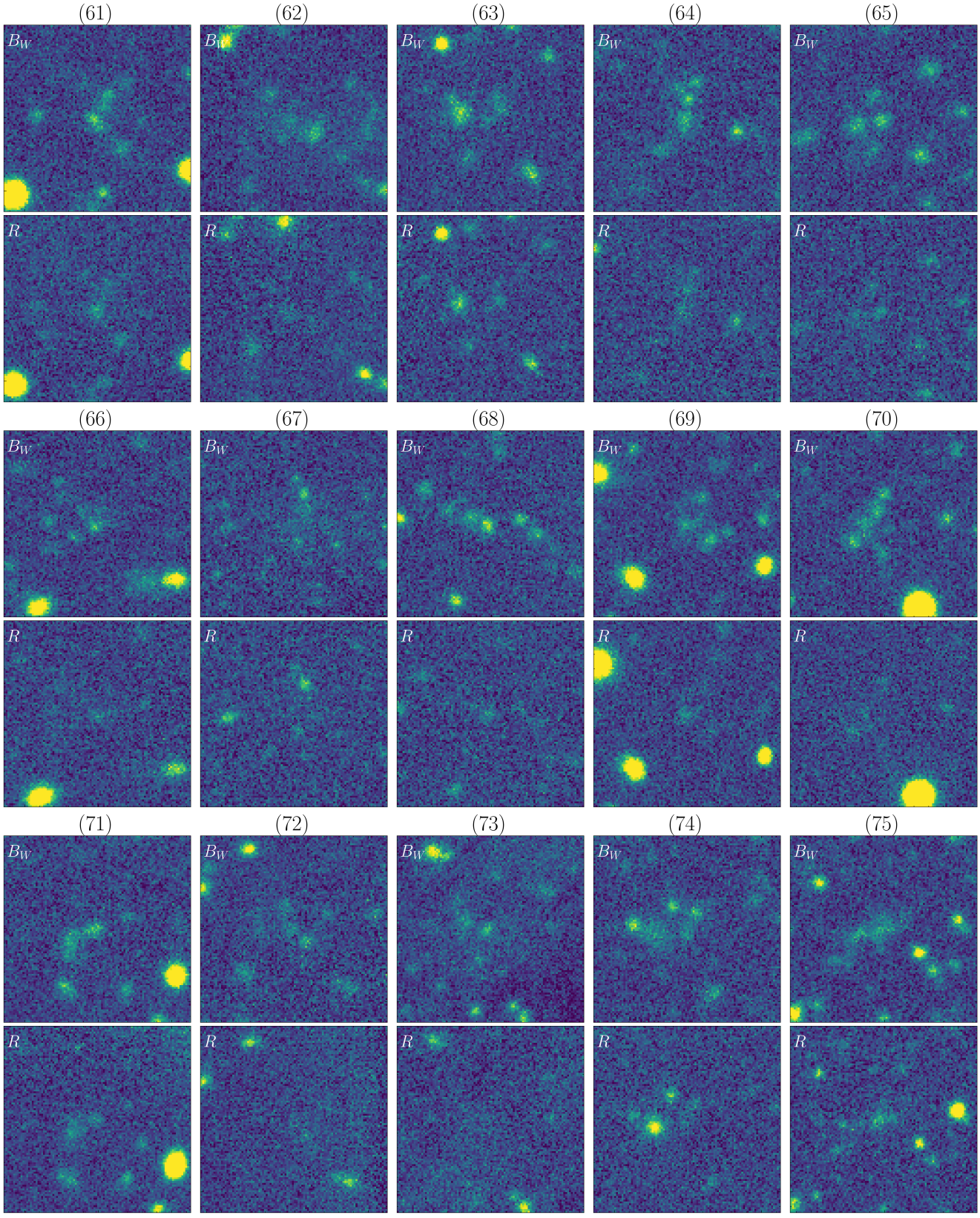

Figure 15. (Continued)

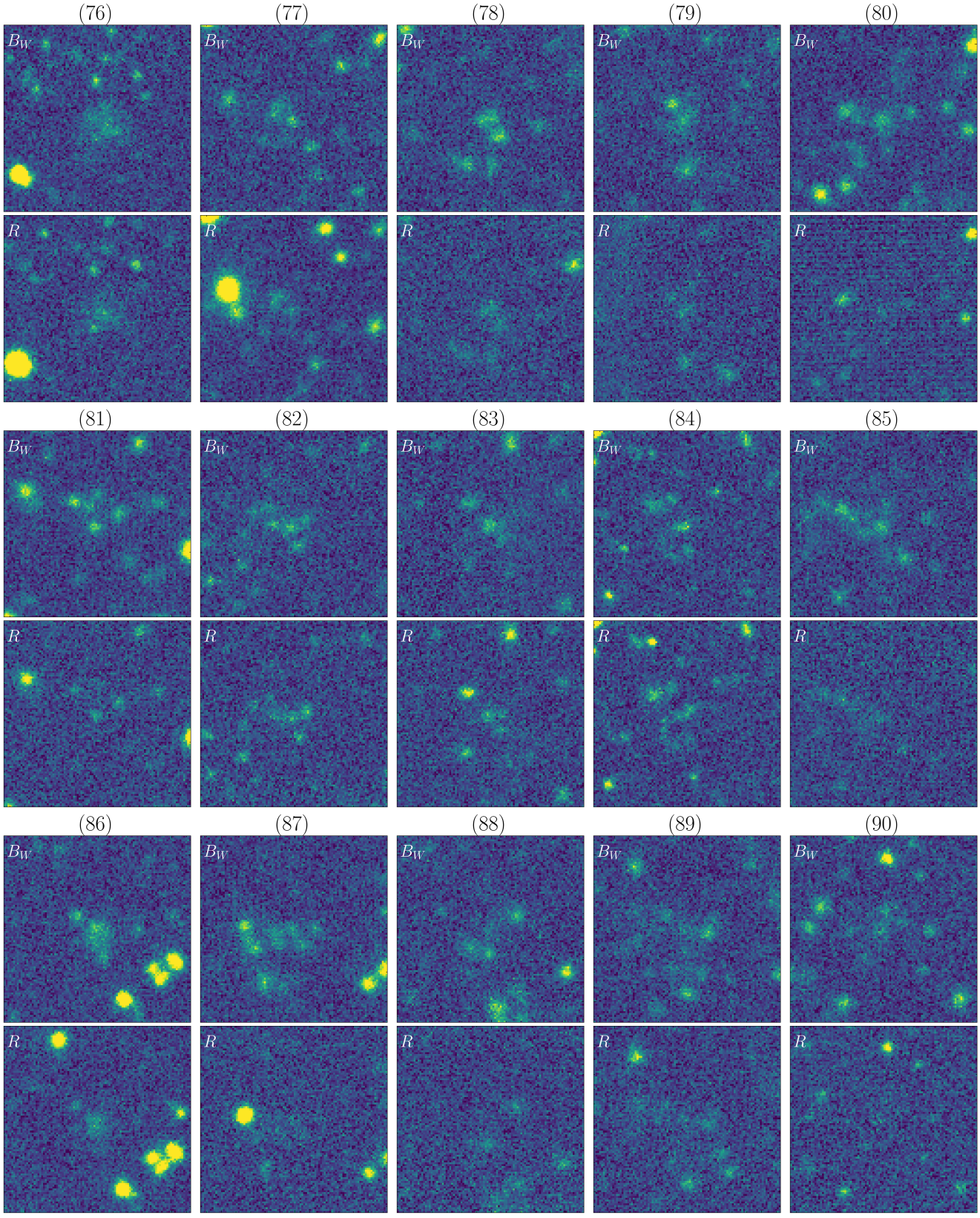

Figure 15. (Continued)

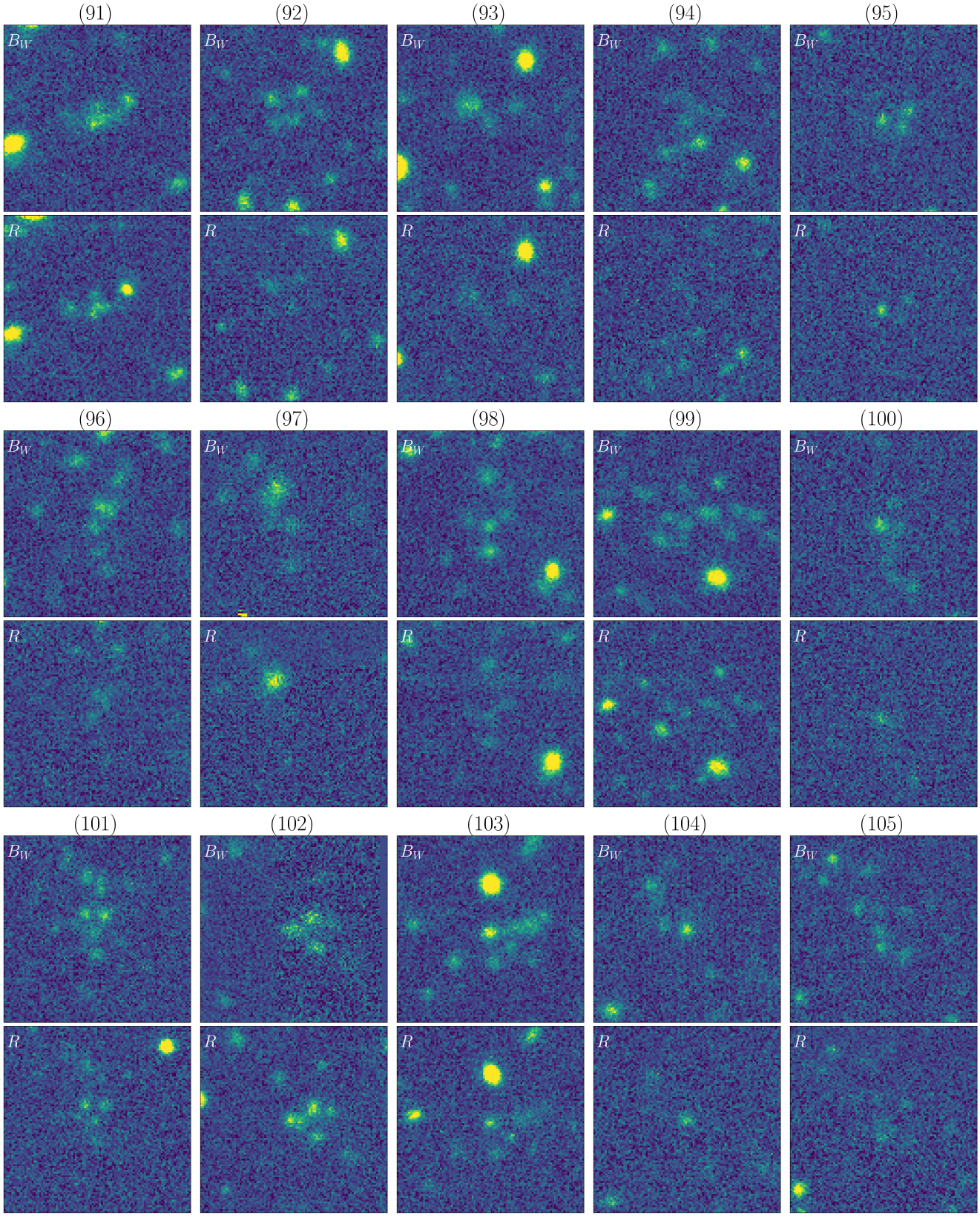

Figure 15. (Continued)

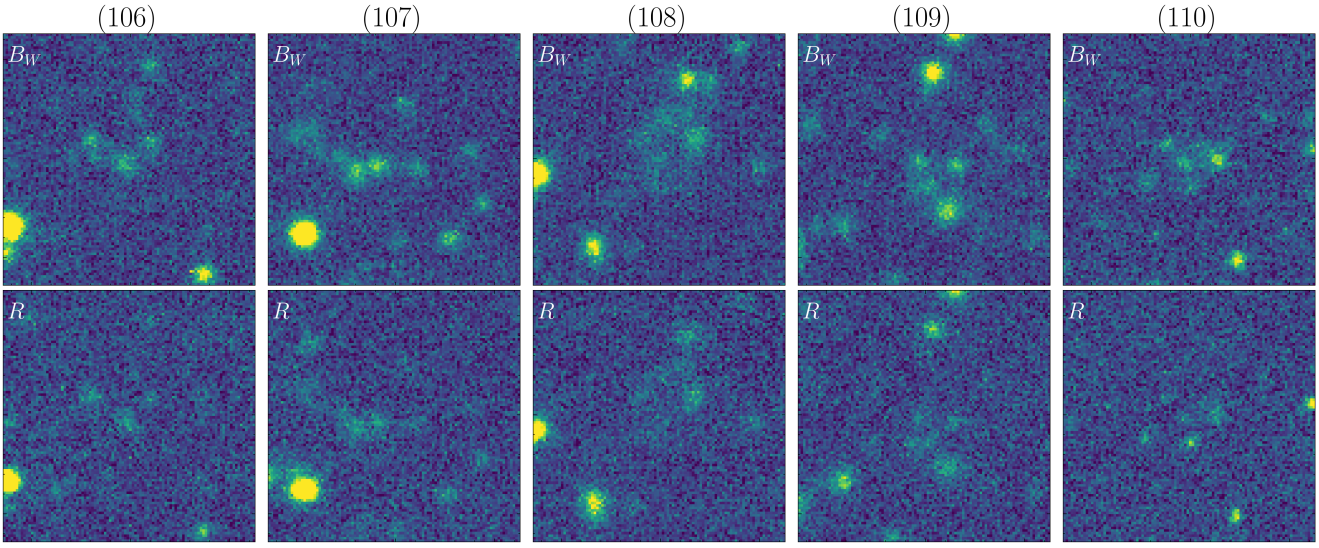

Figure 15. (Continued)

AUTHOR CONTRIBUTIONS

D.G. led the project, including conceptualization, methodology, software development, data curation, analysis, visualization, and preparation of the original draft. M.K.M.P. supervised the project, contributed to the conceptualization and methodology, and provided critical review and editing of the manuscript.

Facility: Mayall (MOSAIC-1 wide-field camera)

Software: `Astropy` (Astropy Collaboration et al. 2013, 2018, 2022), `BorutaSHAP` (E. Keany 2020), `Matplotlib` (J. D. Hunter 2007), `NumPy` (C. R. Harris et al. 2020), `Optuna` (T. Akiba et al. 2019), `Pandas` (W. McKinney et al. 2010), `Photutils` (L. Bradley et al. 2024), `pyWavelets` (G. Lee et al. 2019), `SciPy` (P. Virtanen et al. 2020), `scikit-image` (S. van der Walt et al. 2014), `scikit-learn` (F. Pedregosa et al. 2011), `Tensorflow` (M. Abadi et al. 2015), `xgboost` (T. Chen & C. Guestrin 2016).

REFERENCES

Abadi, M., Agarwal, A., Barham, P., et al. 2015, TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems, https://www.tensorflow.org/

Abraham, R. G., van den Bergh, S., & Nair, P. 2003, The Astrophysical Journal, 588, doi: 10.1086/373919

Ackermann, S., Schawinski, K., Zhang, C., Weigel, A. K., & Turp, M. D. 2018, MNRAS, 479, 415, doi: 10.1093/mnras/sty1398

Akiba, T., Sano, S., Yanase, T., Ohta, T., & Koyama, M. 2019, arXiv e-prints, arXiv:1907.10902, doi: 10.48550/arXiv.1907.10902

Alpaydin, E. 2020, Introduction to machine learning (MIT press)

Aniyan, A. K., & Thorat, K. 2017, The Astrophysical Journal Supplement Series, 230, doi: 10.3847/1538-4365/aa7333

Arlot, S., & Celisse, A. 2009, arXiv e-prints, arXiv:0907.4728, doi: 10.48550/arXiv.0907.4728

Arrigoni Battaia, F., Hennawi, J. F., Prochaska, J. X., et al. 2018, Monthly Notices of the Royal Astronomical Society, 482, 3162, doi: 10.1093/mnras/sty2827

Astropy Collaboration, Robitaille, T. P., Tollerud, E. J., et al. 2013, A&A, 558, A33, doi: 10.1051/0004-6361/201322068

Astropy Collaboration, Price-Whelan, A. M., Sipőcz, B. M., et al. 2018, AJ, 156, 123, doi: 10.3847/1538-3881/aabc4f

Astropy Collaboration, Price-Whelan, A. M., Lim, P. L., et al. 2022, ApJ, 935, 167, doi: 10.3847/1538-4357/ac7c74

Beck, M., Scarlata, C., Hayes, M., Dijkstra, M., & Jones, T. J. 2016, The Astrophysical Journal, 818, 138, doi: 10.3847/0004-637X/818/2/138

Bertin, E., & Arnouts, S. 1996, A&AS, 117, 393, doi: 10.1051/aas:1996164

Boureau, Y.-L., Ponce, J., & LeCun, Y. 2010, in Proceedings of the 27th International Conference on International Conference on Machine Learning, ICML'10 (Madison, WI, USA: Omnipress), 111–118
Bradley, L., Sipőcz, B., Robitaille, T., et al. 2024, astropy/photutils: 2.0.2, 2.0.2 Zenodo, doi: 10.5281/zenodo.13989456
Breiman, L. 1996, Machine Learning, 24, doi: 10.1007/bf00058655
Breiman, L. 2001, Machine learning, 45, 5
Breiman, L., Friedman, J., Olshen, R. A., & Stone, C. J. 1984, Classification and Regression Trees, 1st edn. (Chapman and Hall/CRC), doi: 10.1201/9781315139470
Brigato, L., & Iocchi, L. 2020, arXiv e-prints, arXiv:2003.12843, doi: 10.48550/arXiv.2003.12843
Cai, Z., Fan, X., Yang, Y., et al. 2017, The Astrophysical Journal, 837, 71, doi: 10.3847/1538-4357/aa5d14
Cantalupo, S. 2017, in Astrophysics and Space Science Library, Vol. 430, Gas Accretion onto Galaxies, ed. A. Fox & R. Davé, 195, doi: 10.1007/978-3-319-52512-9_9
Cao, J., Xu, T., Deng, Y., et al. 2024, A&A, 683, A42, doi: 10.1051/0004-6361/202348544
Casey, C. M., Impey, C. D., Trump, J. R., et al. 2008, ApJS, 177, 131, doi: 10.1086/587787
Cecconello, T., Riggi, S., Becciani, U., et al. 2024, arXiv e-prints, arXiv:2411.14078, doi: 10.48550/arXiv.2411.14078
Cen, R., & Zheng, Z. 2013, The Astrophysical Journal, 775, 112, doi: 10.1088/0004-637X/775/2/112
Cervantes, J., Li, X., Yu, W., & Li, K. 2008, Neurocomputing, 71, 611, doi: https://doi.org/10.1016/j.neucom.2007.07.028
Chen, T., & Guestrin, C. 2016, arXiv e-prints, arXiv:1603.02754. https://arxiv.org/abs/1603.02754
Colbert, J. W., Teplitz, H., Francis, P., et al. 2006, The Astrophysical Journal, 637, L89, doi: 10.1086/500647
Crispim Romao, M., & Croon, D. 2024, PhRvD, 109, 123004, doi: 10.1103/PhysRevD.109.123004
Crispim Romao, M., Croon, D., Crossey, B., & Godines, D. 2025, arXiv e-prints, arXiv:2506.20709, doi: 10.48550/arXiv.2506.20709
Daddi, E., Valentino, F., Rich, R. M., et al. 2021, A&A, 649, A78, doi: 10.1051/0004-6361/202038700
Dalal, N., & Triggs, B. 2005, in 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05), Vol. 1, 886–893 vol. 1, doi: 10.1109/CVPR.2005.177
Daubechies, I. 1992, Ten Lectures on Wavelets (Society for Industrial and Applied Mathematics), doi: 10.1137/1.9781611970104
Deng, J., Dong, W., Socher, R., et al. 2009, in 2009 IEEE Conference on Computer Vision and Pattern Recognition, 248–255, doi: 10.1109/CVPR.2009.5206848
Dey, A., Bian, C., Soifer, B. T., et al. 2005, ApJ, 629, 654, doi: 10.1086/430775
Dieleman, S., Willett, K. W., & Dambre, J. 2015, ArXiv, abs/1503.07077. https://api.semanticscholar.org/CorpusID:6247057
Dijkstra, M. 2014, PASA, 31, e040, doi: 10.1017/pasa.2014.33
Dijkstra, M., Haiman, Z., & Spaans, M. 2006, The Astrophysical Journal, 649, 14, doi: 10.1086/506243
Dzigan, Y., & Zucker, S. 2012, Astrophysical Journal Letters, 753, doi: 10.1088/2041-8205/753/1/L1
Eide, M. B., Gronke, M., Dijkstra, M., & Hayes, M. 2018, The Astrophysical Journal, 856, 156, doi: 10.3847/1538-4357/aab5b7
Erb, D., Bogosavljević, M., & Steidel, C. 2011, The Astrophysical Journal Letters, 740, doi: 10.1088/2041-8205/740/1/L31
Fabbro, S., Venn, K. A., O'Briain, T., et al. 2017, Monthly Notices of the Royal Astronomical Society, 475, 2978, doi: 10.1093/mnras/stx3298
Fadely, R., Hogg, D. W., & Willman, B. 2012, ApJ, 760, 15, doi: 10.1088/0004-637X/760/1/15
Faucher-Giguère, C.-A., Kereš, D., Dijkstra, M., Hernquist, L., & Zaldarriaga, M. 2010, ApJ, 725, 633, doi: 10.1088/0004-637X/725/1/633
Fawcett, T. 2006, Pattern Recognition Letters, 27, 861, doi: https://doi.org/10.1016/j.patrec.2005.10.010
Florian, M. K., Li, N., & Gladders, M. D. 2016, The Astrophysical Journal, 832, 168, doi: 10.3847/0004-637X/832/2/168
Francis, P. J., Woodgate, B. E., Warren, S. J., et al. 1996, ApJ, 457, 490, doi: 10.1086/176747
Friedman, J. H. 2001, The Annals of Statistics, 29, 1189 , doi: 10.1214/aos/1013203451
Fu, M.-X., Song, Y., Lv, J.-M., et al. 2024, Chinese Physics C, 48, 095001, doi: 10.1088/1674-1137/ad50ab
Fukushima, K. 1980, Biological Cybernetics, 36, 193. https://api.semanticscholar.org/CorpusID:206775608
Geach, J. E., Matsuda, Y., Smail, I., et al. 2005, Monthly Notices of the Royal Astronomical Society, 363, 1398, doi: 10.1111/j.1365-2966.2005.09538.x
Ghosh, A., Urry, C. M., Wang, Z., et al. 2020, The Astrophysical Journal, 895, doi: 10.3847/1538-4357/ab8a47
Gini, C. 1912, Memorie di Metodologica Statistica (Ed. Pizetti E, Salvemini, T)
Glorot, X., & Bengio, Y. 2010, in Proceedings of Machine Learning Research, Vol. 9, Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics, ed. Y. W. Teh & M. Titterington (Chia Laguna Resort, Sardinia, Italy: PMLR), 249–256. https://proceedings.mlr.press/v9/glorot10a.html
Godines, D. 2025, pyBIA, 1.51 Zenodo, doi: 10.5281/zenodo.17092327

Godines, D., Bachelet, E., Narayan, G., & Street, R. A. 2019, Astronomy and Computing, 28, 100298, doi: 10.1016/j.ascom.2019.100298

Goerdt, T., Dekel, A., Sternberg, A., et al. 2010, Monthly Notices of the Royal Astronomical Society, 407, 613, doi: 10.1111/j.1365-2966.2010.16941.x

Goh, L. W. K., Ocampo, I., Nesseris, S., & Pettorino, V. 2024, A&A, 692, A101, doi: 10.1051/0004-6361/202451099

Goodfellow, I., Bengio, Y., & Courville, A. 2016, Deep Learning (MIT Press)

Green, S., & Lundgren, A. 2024, arXiv e-prints, arXiv:2411.03207, doi: 10.48550/arXiv.2411.03207

Gull, M., Weisz, D. R., El-Badry, K., et al. 2024, arXiv e-prints, arXiv:2410.16393, doi: 10.48550/arXiv.2410.16393

Guyon, I., & Elisseeff, A. 2003, Journal of machine learning research, 3, 1157

Haiman, Z., Spaans, M., & Quataert, E. 2000, The Astrophysical Journal, 537, L5, doi: 10.1086/312754

Harris, C. R., Millman, K. J., van der Walt, S. J., et al. 2020, Nature, 585, 357, doi: 10.1038/s41586-020-2649-2

Harris, F. 1978, Proceedings of the IEEE, 66, 51, doi: 10.1109/PROC.1978.10837

Hastie, T., Tibshirani, R., & Friedman, J. 2009, The Elements of Statistical Learning: Data Mining, Inference, and Prediction, Springer series in statistics (Springer). https://books.google.com/books?id=eBSgoAEACAAJ

Hayes, M., Scarlata, C., & Siana, B. 2011, Nature, 476, 304, doi: 10.1038/nature10320

He, K., Zhang, X., Ren, S., & Sun, J. 2015, arXiv e-prints, arXiv:1502.01852, doi: 10.48550/arXiv.1502.01852

Herenz, E. C., Hayes, M., & Scarlata, C. 2020, A&A, 642, A55, doi: 10.1051/0004-6361/202037464

Herrera-Martin, A., Craiu, R. V., Eadie, G. M., et al. 2025, ApJ, 982, 46, doi: 10.3847/1538-4357/adb623

Hill, G., Lee, H., Macqueen, P., et al. 2021, The Astronomical Journal, 162, doi: 10.3847/1538-3881/ac2c02

Hinton, G. E., & Roweis, S. 2002, in Advances in Neural Information Processing Systems, ed. S. Becker, S. Thrun, & K. Obermayer, Vol. 15 (MIT Press). https://proceedings.neurips.cc/paper_files/paper/2002/file/6150ccc6069bea6b5716254057a194ef-Paper.pdf

Hoag, A., Treu, T., Pentericci, L., et al. 2019, MNRAS, 488, 706, doi: 10.1093/mnras/stz1768

Hu, M.-K. 1962, IRE Transactions on Information Theory, 8, 179, doi: 10.1109/TIT.1962.1057692

Hunter, J. D. 2007, Computing in Science & Engineering, 9, 90, doi: 10.1109/MCSE.2007.55

Huppenkothen, D., Ntampaka, M., Ho, M., et al. 2023, ArXiv, abs/2310.12528. https://api.semanticscholar.org/CorpusID:264306052

Ioffe, S., & Szegedy, C. 2015, arXiv e-prints, arXiv:1502.03167, doi: 10.48550/arXiv.1502.03167

Ivezić, Ž., Kahn, S. M., Tyson, J. A., et al. 2019, ApJ, 873, 111, doi: 10.3847/1538-4357/ab042c

Iwata, I., Inoue, A. K., Micheva, G., Matsuda, Y., & Yamada, T. 2019, Monthly Notices of the Royal Astronomical Society, 488, 5671, doi: 10.1093/mnras/stz2081

Jannuzi, B. T., & Dey, A. 1999, in Astronomical Society of the Pacific Conference Series, Vol. 193, The Hy-Redshift Universe: Galaxy Formation and Evolution at High Redshift, ed. A. J. Bunker & W. J. M. van Breugel, 258

Keany, E. 2020, BorutaShap : A wrapper feature selection method which combines the Boruta feature selection algorithm with Shapley values., 1.1 Zenodo, doi: 10.5281/zenodo.4247618

Kim, E., Yang, Y., Zabludoff, A., et al. 2020, The Astrophysical Journal, 894, 33, doi: 10.3847/1538-4357/ab837f

Kimock, B., Narayanan, D., Smith, A., et al. 2021, ApJ, 909, 119, doi: 10.3847/1538-4357/abbe89

Kingma, D. P., & Ba, J. 2014, CoRR, abs/1412.6980. https://api.semanticscholar.org/CorpusID:6628106

Krizhevsky, A., Sutskever, I., & Hinton, G. E. 2012, in Advances in Neural Information Processing Systems, ed. F. Pereira, C. Burges, L. Bottou, & K. Weinberger, Vol. 25 (Curran Associates, Inc.). https://proceedings.neurips.cc/paper_files/paper/2012/file/c399862d3b9d6b76c8436e924a68c45b-Paper.pdf

Kuhn, M., Johnson, K., et al. 2013, Applied predictive modeling, Vol. 26 (Springer)

Kursa, M. B., & Rudnicki, W. R. 2010, Journal of Statistical Software, 36, doi: 10.18637/jss.v036.i11

Lecun, Y., Bottou, L., Bengio, Y., & Haffner, P. 1998, Proceedings of the IEEE, 86, 2278, doi: 10.1109/5.726791

Lee, G., Gommers, R., Waselewski, F., Wohlfahrt, K., & O'Leary, A. 2019, The Journal of Open Source Software, 4, 1237, doi: 10.21105/joss.01237

Lee, K.-S., Gawiser, E., Park, C., et al. 2024, The Astrophysical Journal, 962, 36, doi: 10.3847/1538-4357/ad165e

Lintott, C. J., Schawinski, K., Slosar, A., et al. 2008, Monthly Notices of the Royal Astronomical Society, 389, 1179, doi: 10.1111/j.1365-2966.2008.13689.x

Liu, F. T., Ting, K. M., & Zhou, Z.-H. 2008, in 2008 Eighth IEEE International Conference on Data Mining, 413–422, doi: 10.1109/ICDM.2008.17

Loecher, M., & Wu, Q. 2021, arXiv e-prints, arXiv:2108.06201, doi: 10.48550/arXiv.2108.06201

Lotz, J. M., Primack, J., & Madau, P. 2004, The Astronomical Journal, 128, 163, doi: 10.1086/421849

Lundberg, S., & Lee, S.-I. 2017, arXiv e-prints, arXiv:1705.07874, doi: 10.48550/arXiv.1705.07874

Lundberg, S. M., Erion, G. G., & Lee, S.-I. 2018, arXiv e-prints, arXiv:1802.03888, doi: 10.48550/arXiv.1802.03888

Maclin, R., & Opitz, D. 2011, arXiv e-prints, arXiv:1106.0257, doi: 10.48550/arXiv.1106.0257

Mas-Hesse, J. M., Kunth, D., Tenorio-Tagle, G., et al. 2003, ApJ, 598, 858, doi: 10.1086/379116

Matsuda, Y., Yamada, T., Hayashino, T., Yamauchi, R., & Nakamura, Y. 2006, The Astrophysical Journal, 640, L123, doi: 10.1086/503362

Matsuda, Y., Yamada, T., Hayashino, T., et al. 2004, The Astronomical Journal, 128, 569, doi: 10.1086/422020

Matsuda, Y., Nakamura, Y., Morimoto, N., et al. 2009, Monthly Notices of the Royal Astronomical Society: Letters, 400, L66, doi: 10.1111/j.1745-3933.2009.00764.x

Matsuda, Y., Yamada, T., Hayashino, T., et al. 2011, Monthly Notices of the Royal Astronomical Society: Letters, 410, L13, doi: 10.1111/j.1745-3933.2010.00969.x

McKinney, W., et al. 2010, in Proceedings of the 9th Python in Science Conference, Vol. 445, Austin, TX, 51–56

Mentuch Cooper, E., Gebhardt, K., Davis, D., et al. 2023, The Astrophysical Journal, 943, 177, doi: 10.3847/1538-4357/aca962

Montesinos López, O. A., Montesinos López, A., & Crossa, J. 2022, Support Vector Machines and Support Vector Regression (Cham: Springer International Publishing), 337–378, doi: 10.1007/978-3-030-89010-0_9

Nair, V., & Hinton, G. E. 2010, in Proceedings of the 27th International Conference on International Conference on Machine Learning, ICML'10 (Madison, WI, USA: Omnipress), 807–814

Nakkiran, P., Kaplun, G., Bansal, Y., et al. 2019, arXiv e-prints, arXiv:1912.02292, doi: 10.48550/arXiv.1912.02292

Nilsson, K. K., Fynbo, J. P. U., Møller, P., Sommer-Larsen, J., & Ledoux, C. 2006, A&A, 452, L23, doi: 10.1051/0004-6361:200600025

Ojala, T., Pietikainen, M., & Maenpaa, T. 2002, IEEE Transactions on Pattern Analysis and Machine Intelligence, 24, 971, doi: 10.1109/TPAMI.2002.1017623

Ouchi, M., Ono, Y., Egami, E., et al. 2009, The Astrophysical Journal, 696, 1164, doi: 10.1088/0004-637X/696/2/1164

Overzier, R. A., Nesvadba, N. P. H., Dijkstra, M., et al. 2013, ApJ, 771, 89, doi: 10.1088/0004-637X/771/2/89

Pal, N., & Pal, S. 1993, Pattern Recognit., 26, 1277, doi: 10.1016/0031-3203(93)90135-J

Palunas, P., Teplitz, H. I., Francis, P. J., Williger, G. M., & Woodgate, B. E. 2004, ApJ, 602, 545, doi: 10.1086/381145

Pashchenko, I. N., Sokolovsky, K. V., & Gavras, P. 2018, Monthly Notices of the Royal Astronomical Society, 475, doi: 10.1093/mnras/stx3222

Pedregosa, F., Varoquaux, G., Gramfort, A., et al. 2011, Journal of Machine Learning Research, 12, 2825

Prescott, M. K. M. 2009, PhD thesis, University of Arizona

Prescott, M. K. M., Dey, A., & Jannuzi, B. T. 2012, ApJ, 748, 125, doi: 10.1088/0004-637X/748/2/125

Prescott, M. K. M., Dey, A., & Jannuzi, B. T. 2013, ApJ, 762, 38, doi: 10.1088/0004-637X/762/1/38

Prescott, M. K. M., Kashikawa, N., Dey, A., & Matsuda, Y. 2008, ApJL, 678, L77, doi: 10.1086/588606

Prescott, M. K. M., Momcheva, I., Brammer, G. B., Fynbo, J. P. U., & Møller, P. 2015, The Astrophysical Journal, 802, 32, doi: 10.1088/0004-637X/802/1/32

Prokop, R. J., & Reeves, A. P. 1992, CVGIP: Graphical Models and Image Processing, 54, 438, doi: https://doi.org/10.1016/1049-9652(92)90027-U

Ramakrishnan, V., Moon, B., Im, S. H., et al. 2023, The Astrophysical Journal, 951, 119, doi: 10.3847/1538-4357/acd341

Reddi, S. J., Kale, S., & Kumar, S. 2019, arXiv e-prints, arXiv:1904.09237, doi: 10.48550/arXiv.1904.09237

Richards, J. W., Starr, D. L., Butler, N. R., et al. 2011, Astrophysical Journal, 733, doi: 10.1088/0004-637X/733/1/10

Romao, M. C., Croon, D., & Godines, D. 2025, Monthly Notices of the Royal Astronomical Society, staf1477, doi: 10.1093/mnras/staf1477

Rosdahl, J., & Blaizot, J. 2012, Monthly Notices of the Royal Astronomical Society, 423, 344, doi: 10.1111/j.1365-2966.2012.20883.x

Sabin, W. E. 2008, Discrete-signal analysis and design (John Wiley & Sons)

Saha, S., Basak, S., Safonova, M., et al. 2018, Astronomy and Computing, 23, doi: 10.1016/j.ascom.2018.03.003

Saito, T., Shimasaku, K., Okamura, S., et al. 2006, The Astrophysical Journal, 648, 54, doi: 10.1086/505678

Sanderson, K. N., Prescott, M. K. M., Christensen, L., Fynbo, J., & Møller, P. 2021, ApJ, 923, 252, doi: 10.3847/1538-4357/ac3077

Schneider, J., Stenning, D. C., & Elliott, L. T. 2023, Frontiers in Astronomy and Space Sciences, Volume 10 - 2023, doi: 10.3389/fspas.2023.1197358

Schweitzer, A., Street, R., Kruszynska, K., & Godines, D. 2025, Research Notes of the AAS, 9, 232, doi: 10.3847/2515-5172/ae0180

Scoville, N., Abraham, R. G., Aussel, H., et al. 2007, ApJS, 172, 38, doi: 10.1086/516580

Shapley, L. 1953, Edited by Emil Artin and Marston Morse, 343

Shibuya, T., Ouchi, M., Nakajima, K., et al. 2014, ApJ, 785, 64, doi: 10.1088/0004-637X/785/1/64

Shimakawa, R. 2022, Monthly Notices of the Royal Astronomical Society, 514, doi: 10.1093/mnras/stac1575

Smith, D. J. B., & Jarvis, M. J. 2007, Monthly Notices of the Royal Astronomical Society: Letters, 378, L49, doi: 10.1111/j.1745-3933.2007.00318.x

Snoek, J., Larochelle, H., & Adams, R. P. 2012, arXiv e-prints, arXiv:1206.2944, doi: 10.48550/arXiv.1206.2944

Soltau, S. B., & Botti, L. C. 2021, Revista Mexicana de Astronomia y Astrofisica, 57, doi: 10.22201/IA.01851101P.2021.57.01.07

Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., & Salakhutdinov, R. 2014, Journal of Machine Learning Research, 15, 1929. http://jmlr.org/papers/v15/srivastava14a.html

Steidel, C. C., Adelberger, K. L., Shapley, A. E., et al. 2000, The Astrophysical Journal, 532, 170, doi: 10.1086/308568

Steidel, C. C., Bogosavljević, M., Shapley, A. E., et al. 2011, The Astrophysical Journal, 736, 160, doi: 10.1088/0004-637X/736/2/160

Taniguchi, Y., & Shioya, Y. 2000, The Astrophysical Journal, 532, L13, doi: 10.1086/312557

Teague, M. R. 1980, J. Opt. Soc. Am., 70, 920, doi: 10.1364/JOSA.70.000920

Terry, J. P., Hall, C., Abreau, S., & Gleyzer, S. 2022, The Astrophysical Journal, 941, 192, doi: 10.3847/1538-4357/aca477

Tohuvavohu, A., Kennea, J. A., Roberts, C. J., et al. 2024, The Astrophysical Journal Letters, 975, L19, doi: 10.3847/2041-8213/ad87ce

van der Maaten, L. 2014, Journal of Machine Learning Research, 15, 3221. http://jmlr.org/papers/v15/vandermaaten14a.html

van der Maaten, L., & Hinton, G. 2008, Journal of Machine Learning Research, 9, 2579. http://jmlr.org/papers/v9/vandermaaten08a.html

van der Walt, S., Schönberger, J. L., Nunez-Iglesias, J., et al. 2014, PeerJ, 2, e453, doi: 10.7717/peerj.453

Venemans, B. P., Röttgering, H. J., Miley, G. K., et al. 2007, Astronomy and Astrophysics, 461, doi: 10.1051/0004-6361:20053941

Vilalta, R. 2018, in Journal of Physics Conference Series, Vol. 1085, Journal of Physics Conference Series, 052014, doi: 10.1088/1742-6596/1085/5/052014

Virtanen, P., Gommers, R., Oliphant, T. E., et al. 2020, Nature Methods, 17, 261, doi: 10.1038/s41592-019-0686-2

Štrumbelj, E., & Kononenko, I. 2014, Knowl. Inf. Syst., 41, 647–665, doi: 10.1007/s10115-013-0679-x

Watanabe, S. 2023, arXiv preprint arXiv:2304.11127

Weaver, J. R., Kauffmann, O. B., Ilbert, O., et al. 2022, ApJS, 258, 11, doi: 10.3847/1538-4365/ac3078

Weaver, J. R., Zalesky, L., Kokorev, V., et al. 2023, The Astrophysical Journal Supplement Series, 269, 20, doi: 10.3847/1538-4365/acf850

Wells, N. K., Prescott, M. K. M., & Finlator, K. M. 2022, ApJ, 941, 180, doi: 10.3847/1538-4357/aca3ac

Willett, K. W., Lintott, C. J., Bamford, S. P., et al. 2013, Monthly Notices of the Royal Astronomical Society, 435, 2835, doi: 10.1093/mnras/stt1458

Wilman, R. J., Gerssen, J., Bower, R. G., et al. 2005, Nature, 436, 227, doi: 10.1038/nature03718

Wyrzykowski, Ł., Rynkiewicz, A. E., Skowron, J., et al. 2015, ApJS, 216, 12, doi: 10.1088/0067-0049/216/1/12

Wyrzykowski, Ł., Kostrzewa-Rutkowska, Z., Skowron, J., et al. 2016, MNRAS, 458, 3012, doi: 10.1093/mnras/stw426

Xiao, X., Ding, J., Luo, X. L., et al. 2024, arXiv e-prints, arXiv:2411.11280, doi: 10.48550/arXiv.2411.11280

Yang, Y., & Li, X. 2024, Universe, 10, doi: 10.3390/universe10050214

Yang, Y., Zabludoff, A., Eisenstein, D., & Davé, R. 2010, The Astrophysical Journal, 719, 1654, doi: 10.1088/0004-637X/719/2/1654

Yang, Y., Zabludoff, A., Tremonti, C., Eisenstein, D., & Davé, R. 2009, The Astrophysical Journal, 693, 1579, doi: 10.1088/0004-637X/693/2/1579

Yao-Yu Lin, J., Liao, S.-M., Huang, H.-J., Kuo, W.-T., & Hsuan-Min Ou, O. 2021, arXiv e-prints, arXiv:2110.01024, doi: 10.48550/arXiv.2110.01024

Zhang, Z., Zou, Z., Li, N., & Chen, Y. 2022, Research in Astronomy and Astrophysics, 22, 055002, doi: 10.1088/1674-4527/ac5732

Zheng, Z., Cen, R., Weinberg, D., Trac, H., & Miralda-Escudé, J. 2011, The Astrophysical Journal, 739, 62, doi: 10.1088/0004-637X/739/2/62

Zhu, X.-P., Dai, J.-M., Bian, C.-J., et al. 2019, Ap&SS, 364, 55, doi: 10.1007/s10509-019-3540-1