Astronomy & Astrophysics manuscript no. output
September 28, 2025

RGC: a radio AGN classifier based on deep learning

I. A semi-supervised model for the VLA images of bent radio AGNs

M. S. Hossain[1], M. S. H. Shahal[2], K. M. B. Asad[2], P. Saikia[2,3], A. Khan[1,2], F. Akter[4,2], A. A. Ali[1], M. A. Amin[1], A. Momen[1,2], and A. K. M. M. Rahman[1]

[1] Center for Computational & Data Sciences, Independent University, Bangladesh, Dhaka 1229, Bangladesh
e-mail: sazzat@iub.edu.bd
[2] Center for Astronomy, Space Science and Astrophysics, Independent University, Bangladesh, Dhaka 1229, Bangladesh
e-mail: kasad@iub.edu.bd
[3] Center for Astrophysics and Space Science, New York University Abu Dhabi, P.O. Box 129188, Abu Dhabi, UAE
[4] Department of Agricultural and Biosystems Engineering, North Dakota State University, Fargo, ND 58108, USA

Received ; accepted

ABSTRACT

Context. Bent radio active galactic nuclei (AGNs), such as wide-angle-tail (WAT) and narrow-angle-tail (NAT) sources, are powerful tracers of dense environments in galaxy groups and clusters. With the advent of large surveys (LOFAR, SKA), automated classification of these sources has become essential.
Aims. We introduce RGC v1.0, the first semi-supervised deep learning classifier designed to distinguish bent radio AGNs into WAT and NAT morphologies. Alongside the model, we release a curated, ML-ready dataset to facilitate future studies.
Methods. RGC combines Bootstrap Your Own Latent (BYOL) self-supervised pre-training with an E(2)-steerable convolutional neural network (E2CNN) to extract rotation-equivariant features. Pre-training was performed on 20,000 unlabeled sources from Radio Galaxy Zoo, followed by fine-tuning on 639 labeled WAT and NAT galaxies curated from the Sasmal et al. (2022) catalog.
Results. The model achieves an accuracy of **87.5%**, with F1-scores of **0.92** (WAT) and **0.85** (NAT). ROC analysis yields an AUC of **0.88–0.94**, and calibration tests give Expected Calibration Errors of **0.11–0.20**, indicating reliable predictions.
Conclusions. RGC v1.0 is the first semi-supervised classifier for bent radio AGNs, providing both a Python package and dataset for scalable application to forthcoming large-area surveys.

Key words. Techniques: image processing – Methods: data analysis – Methods: statistical – Galaxies: active – Radio continuum: galaxies

1. Introduction

In this paper, we present Version 1.0 of RGC, a radio AGN (active galactic nuclei)[1] classifier named after Radha Gobinda Chandra (1878–1975), a Bangladeshi-Indian amateur astronomer who contributed more than fifty thousand observations to the American Association of Variable Star Observers (Maitra 2021) and reported the observation of 1P/Halley in 1910 in Bangla (Kapoor 2023). RGC uses convolutional neural netowrks (CNN) to classify radio AGN (RAGN) with straight and bent tails created from synchrotron jets. Its performance in classifying these objects into Fanaroff–Riley (FR) types I and II was described by Hossain et al. (2023, hereafter H23). In this work, we present and analyze its performance in classifying bent RAGNs into wide-angle-tail (WAT) and narrow-angle-tail (NAT) sources. A background to the present work is given below.

The observable universe could contain almost two trillion galaxies (Conselice et al. 2016), but only a fraction of them have been observed at different wavelengths. The x-ray space telescope eROSITA is expected to find thousands of galaxy clusters and millions of AGNs (Predehl et al. 2021). Samples of 700 thousand AGNs (Merloni et al. 2024) and 12 thousand galaxy clusters and groups (Bulbul et al. 2024) have already been published. In visible light, the eleventh and twelfth data releases of SDSS-III[2] contained more than 1.3 million galaxies and more than a quarter of a million quasars (Alam et al. 2015). The mid-infrared space telescope WISE (Wide-field Infrared Survey Explorer) has identified almost 750 million radio sources which are part of its AllWISE data release (Kurcz et al. 2016). At mid radio frequencies, the Faint Images of the Radio Sky at Twenty-Centimeters (FIRST) taken using the Karl G. Jansky Very Large Array (VLA) contains more than 800 thousand sources in its April 2003 release (Proctor 2011); the initial catalog of Becker et al. (1995) contained 20 thousand sources. At a lower frequency, the Evolutionary Map of the Universe (EMU) conducted using the Australian Square Kilometre Array Pathfinder (ASKAP) contained more than 200 thousand sources in its pilot survey and it is expected to find many more (Norris et al. 2021). Moreover, at the lowest radio frequencies, LOw Frequency ARray (LOFAR) has detected almost 4.4 million sources as part of its Two-metre Sky Survey LOTSS (Shimwell et al. 2022).

Astronomers are also cross-matching the sources seen at different wavelengths, especially the AGNs with their host galaxies. For example, Best & Heckman (2012) cross-matched FIRST

[1] The radio AGNs we classify here are sometimes called radio galaxies, but we use the more general term 'radio AGN' (RAGN) throughout.

[2] Sloan Digital Sky Survey

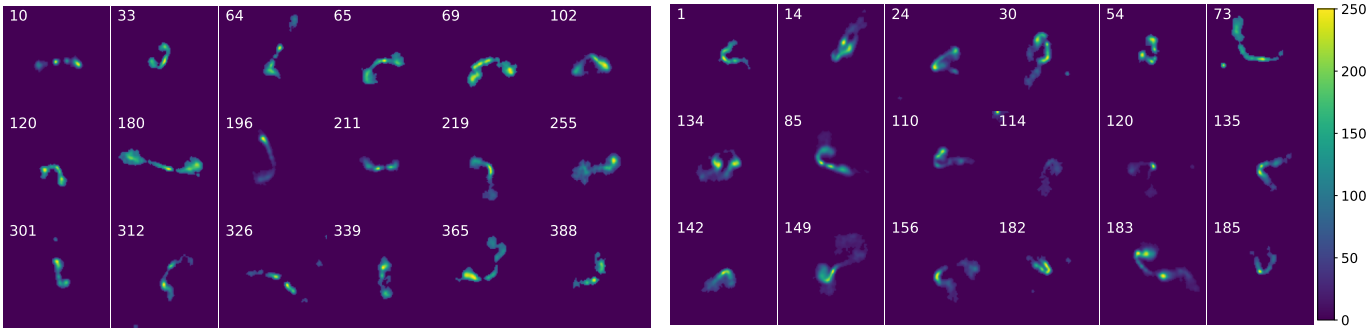


Fig. 1: Some representative bent radio AGNs from the catalog of Sasmal et al. (2022a,b), imaged by VLA as part of FIRST. *Left*: 18 wide-angled tail sources (WATs), and *Right*: 18 narrow-angled tail sources (NATs). The pixel values were originally in Jy, but they have been normalized to a dimensionless range from 0 to 255 for producing PNG images. All these pre-processed images have the same size: 100 pixels (~ 3 arcmin) on either side. In the actual work, we have used images with 150^2 pixels. On each panel, the top-left number indicates the object index from the Sasmal catalog.

and SDSS with NVSS (NRAO[3] VLA Sky Survey) to produce a catalog of more than 18 thousand sources. Traditionally cross-matching has been achieved through visual inspection by astronomers which is not feasible for the latest generation of surveys. Therefore, Banfield et al. (2015) created the citizen science project Radio galaxy Zoo (RGZ) where users were asked to cross-identify radio images from FIRST and ATLAS (Australia Telescope Large Area Survey, a pathfinder of EMU) with their host galaxies imaged by WISE or Spitzer Space Telescope in infrared. Data Release 1 of RGZ contains classifications of 99,146 radio sources from FIRST, with an average reliability of 0.83 based on the weighted consensus levels of the citizen scientists (Wong et al. 2025).

If we focus on the history of 'radio' continuum surveys from the days of Grote Reber until the last decade, we see that the number of detected sources increased by an order of four from the 1940s to the 1970s, and since then we have seen an increase by four more orders of magnitude (Figure 2 of Norris 2017). EMU and LOFAR are expected to find close to 100 million radio sources. The number will increase a lot when the Square Kilometer Array (SKA) begins to operate in two phases, SKA1 and SKA2, and at two different frequency bands, SKA-Low and SKA-Mid. At mid-frequencies, SKA1 All Sky Survey (SASS1) will detect around 500 million galaxies and SASS2 over 3 billion galaxies spanning all redshifts (Norris et al. 2015). The first of the 131,072 two-meter-tall Christmas-tree-shaped antennas of SKA-Low was installed in a Wajarri country in Australia on 7 March 2024 and the first of the 197 fifteen-meter-wide dishes of SKA-Mid was lifted onto a pedestal on 4 July 2024 in the Karoo region of South Africa.[4]

In this context, artificial intelligence (AI) is necessary for localizing astronomical sources in large datasets and classifying them into scientifically meaningful classes. A recent review by Ndung'u et al. (2023) emphasizes on the new paradigm shift that has happened due to the application of machine learning (ML) and deep learning (DL) for the morphological classification of radio AGNs. They have reviewed 32 papers published between 2017 and 2023 that utilized both conventional ML and DL methods. The most frequently used methods were found to be based on shallow and deep convolutional neural networks (CNN), as evident in their Figure 7. The methods were divided into model-centric and data-centric approaches. The model-centric approaches focus on applying novel architectures on existing well-curated datasets which are scarce. Most methods use low-resolution images taken by older telescopes such as, VLA. These are sometimes not suited to the new high-resolution images coming out of the latest surveys from the new generations of telescopes, for example, LOFAR and MeerKAT. The data-centric approaches leverage the availability of more numerous images and focus on transfer learning and semi-supervised learning.

Our data-centric approach is based on semi-supervised learning, where we leverage the availability of unlabeled data for classifying bent radio AGNs. The model used here is the same as that of H23, but we use it for classifying bent AGNs for the first time. The paper is organized as follows. Section 2 gives an overview of bent AGNs and Section 3 gives a detailed description of the data we have prepared for an efficient use in DL. Section 4 presents our semi-supervised model in a more detailed manner than H23. Both Sections 3 and 4 refer to the installable Python package `RGC`[5] throughout, which is released along with this paper. Section 5 describes the performance of the model in classifying the bent AGNs of our dataset, and Section 6 gives a critical discussion of the limitations and prospects of the model. We conclude the paper with Section 7.

2. Bent radio active galactic nuclei

AGNs are the centers of galaxies that produce extremely high luminosities from an extremely small region. They are made of a supermassive black hole (SMBH) surrounded by a thin accretion disk and a thick torus. The gravitational potential energy of the SMBH is the source of their luminosity. Outflows of high-energy particles are found along the poles of the disk or torus. These bipolar jets and their tails emit radio light through the synchrotron mechanism, creating the radio, or radio-loud, AGNs, the so-called RAGNs (Urry & Padovani 1995). If the brightness of the jets decreases from the center toward the edges, the RAGNs are called FRI sources, and, conversely, if the brightness increases toward the edges, they are called FRII sources (Fanaroff & Riley 1974). RAGNs of a hybrid FRI-FRII morphology are also found.

Sometimes the jets and tails of the RAGNs (especially the FRI sources) are found to be bent, in which case they are called bent RAGNs, bent-tailed radio galaxies (BTRG; Lao et al. 2025),

[3] National Radio Astronomy Observatory
[4] According to `https://www.skao.int/news`.
[5] `https://pypi.org/project/rgc`.

Table 1: The surveys conducted by array radio telescopes (ART) that have found most of the bent RAGNs so far.

ART	Survey	Location	Sensitivity	Resolution	RAGNs	WATs	NATs	Bent RAGN reference
VLA	FIRST	USA	150	5″ (1400)	717	430	287	Sasmal et al. (2022a)
					4876	4424	652	Lao et al. (2025)
LOFAR	LOTSS	Netherlands	83	6″ (144)	35			Golden-Marx et al. (2021)
					55	45	10	Pal & Kumari (2023)
ATCA	ATLAS	Australia	20	12″ (1400)	45			Dehghan et al. (2014)
GMRT	TGSS	India	3500	25″ (150)	264	203	61	Bhukta et al. (2022)
MeerKAT	MIGHTEE	South Africa	6	5″ (1250)	359			Vardoulaki et al. (2025)

Notes. Sensitivity is given as 'median rms' in μJy beam^{-1}. Resolution is in arcsec, with the frequency given within brackets in MHz.

or simply tailed radio galaxies (Bhukta et al. 2022). Some examples are given in Figure 1. If the bending angle, or opening angle (BA / OA) of a bent RAGN is less than 90°, forming a shape similar to 'V', they are called narrow-angle-tail sources, NATs, or head-tail sources (HT; Rudnick & Owen 1976) because sometimes the two tales are so close that they create a single tail. And if the OA is more than 90°, forming a shape similar to 'C', they are called wide-angle-tail sources, WATs (Owen & Rudnick 1976; O'Dea & Baum 2023). An angle of exactly 90° would produce a shape similar to 'L' which could be considered a hybrid WAT-NAT source. Clear identification of a source as either WAT or NAT depends on various observational effects: projection effects related to the orientation of the object on the plane of our sky (Proctor 2011), and the sensitivity and resolution of our telescopes.

WATs show a sudden transition from jets to tails, possibly related to the transition from the interstellar medium (ISM) of a host galaxy to the intracluster medium (ICM) of a cluster (O'Dea & Baum 2023) or the intergalactic medium (IGM) of a dense environment. The reasons for this transition, the distortion of jets and the overall bending of tails provide motivations behind RAGN research: they can give insights into the nature of IGM in dense environments like groups and clusters of galaxies. Vardoulaki et al. (2025, Page 1) lists several reasons behind jet distortions discussed in the literature, including the path of jets in IGM (Miley et al. 1972), buoyancy forces, precession of jets, gravitational interaction with nearby galaxies, and the path of jets through a steep pressure gradient. Bending of tails due to ram pressure in IGM and ICM, have been studied particularly well. For example, the works of Garon et al. (2019), Mingo et al. (2019), and Golden-Marx et al. (2021) have shown that bent RAGNs are found predominantly in denser environments, if not in clusters, up to a redshift of 2.2. Moreover, more bent RAGNs are found in richer and more massive clusters; and the more bent the tails are the more they tend to be located near the centers of clusters (Vardoulaki et al. 2025, Page 2).

A substantial number of bent RAGNs have been found in the surveys conducted by different array radio telescopes (ART), for example, FIRST by VLA, LOTSS by LOFAR, ATLAS by Australia Telescope Compact Array (ATCA), the Tata Institute of Fundamental Research (TIFR) Giant Metrewave Radio Telescope (GMRT) Sky Survey (TGSS), and the MeerKAT International GHz Tiered Extragalactic Explorations (MIGHTEE). The number of bent RAGNs found in each case and the corresponding reference papers are given in Table 1, which clearly shows that FIRST has produced by far the highest number of these sources.

The bent RAGNs of FIRST have attracted a lot of investigations. An early example is Proctor (2011), where 7106 sources with four or more components were visually examined and manually annotated from more than 800,000 sources. The visual inspection was performed on 4′ cutouts, and around 400 sources were identified as either WAT or NAT, although some of them were uncertain. Garon et al. (2019) presented a sample of 4304 RAGNs from RGZ, and 87% of the sample were found to be within 50 Mly (mega-light-year) of an optically identified cluster. The brightest cluster galaxies (BCGs) were found to be more likely to harbor radio-emitting bent tails. However, they did not classify the sample into WATs and NATs.

Sasmal et al. (2022a) created the largest categorized sample of WATs and NATs up to that time. They used the Decemer 2014 data release of FIRST containing almost a million radio sources. In order to find extended RAGNs, they first filtered the sources with an angular size greater than 10″, at least twice the size of the synthesized beam of VLA. Around 95,000 automatically filtered sources were then visually examined, a remarkable achievement. During this inspection, 717 sources were categorized as either WAT or NAT based solely on the angle between the two tails. In order to determine the center of a RAGN from where the angle has to be calculated, they searched for the optical counterparts from SDSS data release 12. Optical counterparts were found for only half of the sources. For the other half, the center was guessed through 'eye estimation.'

Sasmal et al. (2022a) gives two separate tables for WAT (their Table 1) and NAT (their Table 2), and each table has 12 columns: Catalog Number, Name, R.A. (right ascension), Decl. (declination), Ref. (reference survey for finding counterparts), Redshift, F_{1400} (the flux density at 1400 MHz in mJy, taken from NVSS), F_{150} (flux density at 150 MHz, taken from TGSS), α_{150}^{1400} (the spectral index between 150 MHz and 1400 MHz), L (the luminosity in erg/s), FR Type, Other Catalogs (reference for a source in other catalogs). Both WAT and NAT have independent catalog numbering, each starting from 1. The scarcity of available information resulted in the inability to populate all data columns for a significant proportion of the sources.

An even bigger dataset (around 4800 sources) of WATs and NATs from the FRIST images was presented by Lao et al. (2025), and it is heavily skewed toward WATs: only 13% of the sources are NATs, as evident from our Table 1. They found optical counterparts for 86% of their sources using optical data from DESI (Dark Energy Spectroscopic Instrument) Legacy Surveys. Lao et al. (2025) cover a luminosity range of 8 orders of magnitude starting from 10^{20} W Hz^{-1}, up to a redshift of 3.43. Almost 37% of their RAGNs were found to be located in a cluster. Unlike the non-ML automated filtering method of Sasmal et al. (2022a), they have used a DL-based source finder called Radio Galaxies Classification with Mask Transfiner (RGCMT, their Section 2.2), that utilizes transformers. It was trained on

a manually annotated dataset of around 3600 sources, including 400 bent RAGNs. Searching through almost a million FIRST sources, RGCMT could separate around 11,000 bent RAGN candidates, taking more than 17 hours with 3 GPUs (graphics processing units). These sources were visually inspected to produce the final catalog of 4876 bent RAGNs.

Unlike the supervised model of Lao et al. (2025) based on Mask R-CNN, our semi-supervised approach uses both unlabeled and labeled data. We perform the self-supervised pre-training on an unlabeled dataset of 20,000 sources taken from RGZ (described in the next section). And for supervised fine tuning, we have used the catalog of bent RAGNs made by Sasmal et al. (2022a). The number of labeled bent RAGNs in the training dataset of Lao et al. (2025) was around 400, less than the 717 labeled WATs and NATs of Sasmal et al. (2022a). Given the class imbalance due to scarcity of NAT sources, currently there are no semi-supervised binary classifier to categorize bent RAGNs into WATs and NATs. Our work fulfills this need. We also provide a batched dataset of bent RAGNs, labeled as WATs or NATs, ready for use in ML. Our processing and data preparation steps are described next.

3. Data and pre-processing

As mentioned before, both our labeled ($\mathbf{R}_L$) and unlabeled ($\mathbf{R}_U$) datasets are created using FIRST images.[6] Although the angular resolution of FIRST was 5 arcsec at 1400 MHz (Table 1), the final images have a pixel resolution of 1.8 arcsec. Although Figure 1 showed the inner 100^2 pixel cutouts, the images we use in training and testing have a dimension of 150 pixels on either side, giving an angular size of 4.5 arcmin.

3.1. Unlabeled data

We have used the batched dataset of 20,000 sources created for use in ML by Slijepcevic et al. (2022b) from RGZ. As described in their Section 3.2 and in Wong et al. (2025), RGZ data resease 1 (DR1) contains approximately 100,000 sources, 99.2% of which were taken from FIRST. For each FIRST object in this catalog with the largest angular size (θ_{LAS}) between 15 and 270 arcsec, Slijepcevic et al. (2022b) downloaded the corresponding image using the `Python API` of `SkyView`, a virtual observatory that provides access to a wide range of astronomical images from different surveys.[7] They cropped the inner 150^2 pixels from the original 300^2-pixel images, and put all pixels outside a radius of $0.6\theta_{LAS}$ to zero. After applying a $3-\sigma$ amplitude thresholding, each image R_o was normalized to

$$R_n = 255 \times \frac{R_o - min(R_o)}{max(R_o) - min(R_o)} \quad (1)$$

in order to convert them to PNG (Portable Network Graphic) format, where $min(R_o)$ is the minimum value of R_o, and $max(R_o)$ is the maximum. Thereafter all unresolved sources were removed, and the dataset was further reduced by discarding sources that were too dissimilar to their labeled dataset of FRI and FRII sources, the MiraBest dataset (Porter & Scaife 2023) which was also used in H23. The similarity between RGZ DR1 and MiraBest was analyzed using Fréchet inception distance (FID), as described in their Section 4.3. Most importantly, the labeled sources of MiraBest that were present in RGZ were also removed, so that there is no overlap between the labeled and unlabeled data. The final unlabeled dataset contained 20,000 sources organized in 10 batches of training data, and one batch of test data. We downloaded these from their GitHub repository[8] and used them without any further processing. Hereafter, this batched dataset will be referred to as $\mathbf{R}_U$.

Slijepcevic et al. (2024) uses a larger unlabeled dataset of 108,000 sources from RGZ for self-supervised learning, but pre-training on such a large dataset is beyond the scope of our present work.

3.2. Labeled data

Unlike $\mathbf{R}_U$, our labeled dataset $\mathbf{R}_L$ has been pre-processed for efficient use in ML in this work for the first time. As mentioned in Section 2, this dataset is created using the catalog of more than 700 WATs and NATs published by Sasmal et al. (2022b) (hereafter, the Sasmal catalog). However, it was created through visual inspection and, hence, there was no batched dataset ready for use in ML based on the catalog. We describe below how we refined the Sasmal catalog and prepared an ML-friendly batched dataset.

3.2.1. Downloading

We accessed the Sasmal catalog from VizieR[9] using the `astropy` module `astroquery`. It contains two tables: Table 1 lists 430 WATs, while Table 2 lists 287 NATs. The different columns of the table have been described above in Section 2. The catalog was downloaded in `VOTable` format and converted to a `Pandas DataFrame` for further processing. To facilitate automated catalog retrieval, we developed the `catalog_quest` function within our `RGC` package, which directly queries VizieR and returns the data in a structured `DataFrame` format.

The coordinates of the sources were used to download the corresponding FIRST images through the `Python API` of `SkyView` within `astroquery`.[10] The images were retrieved in `FITS` (Flexible Image Transport System) format. To facilitate the automated download of images, we have developed the function `celestial_capture` within `RGC`. This function takes the survey name (e. g., `FIRST`), source coordinates and output file path as input, and downloads the images of the sources to the specified file path. Additionally, we provide the function `celestial_capture_bulk`, which enables batch downloading of all sources in a given catalog. This function takes the catalog `DataFrame`, survey name, and directory path as input and downloads the sources in the catalog to the specified directory.

Among the 717 bent RAGNs of the Sasmal catalog, we could successfully retrieve 703. The remaining 14 sources could not be retrieved due to either server errors during the download, or the downloaded image being blank. Among the 703 images, 281 were NATs and 422 were WATs. We pre-processed these images and created a refined annotated catalog for efficient use in ML which will be called Sasmal-ML hereafter. The pre-processing were performed using two separate software: `PyBDSF` (Mohan & Rafferty 2015) and `Photutils` (Bradley et al. 2025). The labeled dataset $\mathbf{R}_L$ created through `PyBDSF` processing will be

[6] We use the symbol **R** because not only are these 'radio' images, but also the values of the pixels are 'real' numbers.

[7] Developed at the auspices of the High Energy Astrophysics Science Archive Research Center (HEASARC) at the Goddard Space Flight Center (GSFC) of NASA; `https://skyview.gsfc.nasa.gov`.

[8] `https://github.com/inigoval/fixmatch`

[9] `https://vizier.cds.unistra.fr/viz-bin/VizieR`

[10] `astroquery.skyview.SkyView`

called $\mathbf{R}_{L1}$ and the one created through `Photutils` processing will be called $\mathbf{R}_{L2}$.

Because of the way we downloaded the images, the target is always located at the center of a 150^2-pixel image. During the first step of pre-processing, we created a mask that can be applied to the image to remove everything except the central target. This was performed in two steps: first, a mask was generated using `PyBDSF` for $\mathbf{R}_{L1}$ and, then, the mask was refined using `Photutils` for $\mathbf{R}_{L2}$. The two steps are described in the subsequent two subsections.

3.2.2. Generating masks using `PyBDSF`

We start by estimating the background noise level for each image. `PyBDSF` achieves this by segmenting the image into smaller subregions and computing the local mean ($\mu_{\rm local}$) and standard deviation ($\sigma_{\rm local}$) of the pixel intensities within each subregion. As given in the documentation of `PyBDSF`,[11] the background intensity at a pixel (x, y)

$$B(x, y) = \mu_{\rm local} + k \cdot \sigma_{\rm local} \tag{2}$$

where k is a scaling factor accounting for typical noise fluctuations. After background estimation, sources are detected by identifying contiguous regions called 'islands' where the intensity exceeds the local background by a defined threshold. Two types of thresholds are applied. The 'island threshold' $T_{\rm isl}$ is the minimum signal level for considering a group of contiguous pixels as part of an island. A pixel is included in an island if its intensity

$$R(x, y) > B(x, y) + T_{\rm isl} \cdot \sigma_{\rm local} \tag{3}$$

where we used $T_{\rm isl} = 3$ for all cases. The 'peak threshold' $T_{\rm pix}$ is the peak intensity of an island, and only islands with peaks exceeding

$$R(x, y) > B(x, y) + T_{\rm pix} \cdot \sigma_{\rm local} \tag{4}$$

are considered valid detections. We used $T_{\rm pix} = 5$ for all the cases. Once the thresholds are applied, contiguous pixels meeting the criteria are grouped into islands. For each island, a two-dimensional Gaussian model is fitted to extract key parameters, including the centroid (x_0, y_0), peak intensity R_0, and morphological descriptors, for example, major axis $\sigma_{\rm maj}$, minor axis $\sigma_{\rm min}$, and position angle θ. The Gaussian model is

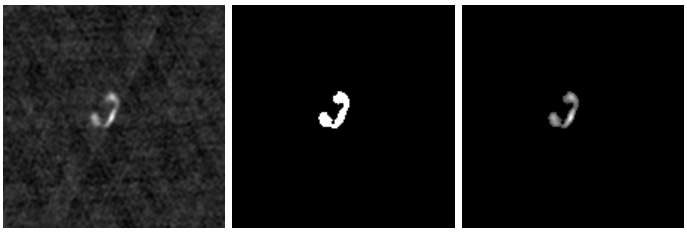
$$R(x, y) = R_0 \exp\left(-\frac{(x' - x_0)^2}{2\sigma_{\rm maj}^2} - \frac{(y' - y_0)^2}{2\sigma_{\rm min}^2}\right), \tag{5}$$

where (x', y') are the coordinates rotated by the angle θ. Once sources are identified, a binary mask is generated to delineate the spatial extent of each source. The mask $M(x, y)$ is created by thresholding the fitted Gaussian model as

$$M(x, y) = \begin{cases} 1, & \text{if } R(x, y) > B(x, y) + T_{\rm pix} \cdot \sigma_{\rm local}, \\ 0, & \text{otherwise.} \end{cases} \tag{6}$$

For sources with extended emission, the mask undergoes dilation by d pixels to ensure complete coverage of the source. The final binary masks are stored in FITS format for further analysis. These steps were performed using the `process_image` function of `PyBDSF` that takes the beamshape, frequency, and the aforementioned threshold parameters as inputs, and provides a `PyBDSF` object as output containing the detected sources. Using the `export_fits` method, we export the mask of the sources in FITS format, applying a dilation of $d = 0$ for most sources, with different values used for a few cases where adjustments were needed. The images for which different dilations had to be applied are identified in Sasmal-ML, the catalog published with this paper.

[11] `https://pybdsf.readthedocs.io/en/latest/`



(a) Original $R_{\rm PNG}$ (b) Mask $M_{\rm PNG}$ (c) Masked $R_{\rm masked}$

Fig. 2: An example of the process of masking an image (described in Section **??**) using `PyBDSF` for removing everything other than the bent RAGN at the center.

For ease of use, we provide the function `generate_mask` in `RGC`, which automates this process. It takes the image file name, beamshape, frequency, threshold values, dilation, and output file name as inputs, and generates the corresponding mask as output. Additionally, we offer `generate_mask_bulk` to process an entire catalog of images.

3.2.3. Refining Masks Using `Photutils`

To further improve the isolation of the sources, we applied a second, independent masking based on `Photutils`. to capture faint, connected emission that may be missed or to prune spurious detections. This refinement proceeds in two steps: (i) generating a mask with `Photutils` from each 150×150 FITS image, and (ii) taking the pixel-wise product of the `PyBDSF` and `Photutils` masks to form a conservative final mask.

For each image, we first compute robust background statistics using `sigma_clipped_stats`, obtaining the median and standard deviation. We then set a detection threshold at

$$T = \text{median} + \alpha\,\sigma,$$

with α tuned per our pipeline (typical values $\alpha \in 3$ depending on noise level). Sources are detected using `detect_sources` with a minimum area constraint (`npixels`) to suppress noise islands. Since the target is known to be at the image center, we identify the label at the central pixel; if the central pixel is background, we search a small neighborhood and select the most significant nearby label. To ensure contiguous coverage of extended emission (e.g., lobes), we optionally apply binary dilation by d pixels. The resulting binary mask $M_{\rm phot}(x, y) \in \{0, 1\}$ is saved as a FITS image with the original header preserved, thus matching the input WCS and dimensions.

- Background estimation: `sigma_clipped_stats` (median, σ), threshold T = median + $\alpha\sigma$. - Source detection: `detect_sources` with `npixels` for a minimum island size. - Target isolation: select central label or dominant label in a small neighborhood. - Refinement: optional morphological dilation by d pixels. - Outputs: $M_{\rm phot}$ and $M_{\rm final}$ stored as FITS, preserving input headers.

3.2.4. Mask combination.

Let $M_{\rm bdsf}(x, y)$ denote the PyBDSF-derived island mask (exported via process_image and export_image with optional dilation), and $M_{\rm phot}(x, y)$ the Photutils mask. We form a conservative, high-confidence final mask by the pixel-wise product

$$M_{\rm final}(x, y) = M_{\rm bdsf}(x, y) \cdot M_{\rm phot}(x, y),$$

which is equivalent to a logical AND, keeping only pixels that are simultaneously supported by both methods. In practice, PyBDSF masks may be stored as $(1, 1, 150, 150)$ arrays; we remove singleton dimensions to $(150, 150)$ before multiplication to ensure shape compatibility. The combined mask is written to FITS with the original header, enabling downstream use without additional registration.

Implementation notes. - Background estimation: sigma_clipped_stats (median, σ), threshold T = median + $\alpha\sigma$. - Source detection: detect_sources with npixels for a minimum island size. - Target isolation: select central label or dominant label in a small neighborhood. - Refinement: Morphological dilation by d pixels. - Outputs: $M_{\rm phot}$ and $M_{\rm final}$ stored as FITS, preserving input headers.

This two-stage masking strategy leverages complementary strengths: PyBDSF provides robust radio source finding with beam- and frequency-aware thresholds, while Photutils offers flexible, image-statistics–driven segmentation. Their intersection yields masks that are precise at the target location and resilient to background fluctuations and artifacts.

3.2.5. Conversion to PNG

The downloaded images and the generated masks were in FITS format, which is not suitable for efficient use in ML. Therefore, we converted the images and masks to PNG, which is widely used for image processing. For this conversion, the images were first normalized following Equation 1. The normalized images and masks were then converted to PNG using Pillow[12].

RGC has the function fits_to_png to convert the FITS files to PNG format automatically. It takes the FITS file name as input and converts the image to PNG format and returns an image in the form of a PIL.Image object. We also provide a function fits_to_png_bulk to convert all FITS files in a given directory to PNG format. The function takes the directory path and output directory path as input and converts all FITS files in the directory to PNG format and saves them in the output directory.

3.2.6. Applying the masks

Background removal from the radio images is achieved by applying the binary masks (values either 0 or 1) generated in the previous step. By multiplying the original image with its corresponding binary mask, the background is suppressed, retaining only the source emissions. This operation can be expressed as

$$R_{\rm masked} = R_{\rm PNG} M_{\rm PNG} \quad (7)$$

where $R_{\rm masked}$ represents the background-subtracted image, $R_{\rm PNG}$ the original normalized image in PNG, and $M_{\rm PNG}$ its mask. Since the mask contains ones for the sources and zeros for the background, this multiplication removes the background while preserving the detected sources. This process is applied to all images in the dataset, resulting in a series of background-subtracted images that highlight the radio sources. Figure 2 illustrates the background removal process for a sample radio image.

RGC provides two functions named mask_image and mask_image_bulk in order to automate this process. The mask_image function takes PIL.Image objects of the image and the mask as inputs, applies the mask on the image, and returns the masked image as a PIL.Image object. The mask_image_bulk function processes an entire dataset by taking the directory paths of images and masks as input, applying the masks to all images, and saving the masked images in the specified output directory.

3.2.7. Sasmal-ML: a bent RAGN catalog and dataset for ML

The 703 images, masked, processed and converted following the aforementioned steps, were visually examined by three authors of this paper: Saikia, Asad and Hossain. The astronomers Saikia and Asad commented on the quality of the images, and Hossain (a student of computer science) made the final decision about keeping or discarding an object based on the comments of the astronomers. The complete annotations are provided in our Sasmal-ML Catalog (SMC) provided in the Github repository of RGC as a Google Spreadsheet and a TSV (tab separated values) file. Here we show 6 rows for the first 3 WATs and the first 3 NATs in Table 2 for demonstrating the structure of the catalog.

SMC contains 703 rows and 15 columns, described in 2. The original Sasmal catalog contained 717 sources, but 14 of its sources could not be downloaded. Among the 703 sources of SMC, 639 were finally selected for our Sasmal-ML Dataset (SMD). The reasons behind discarding 64 sources can be found in SMC. From the Hossain column of SMC, one can see that 31 sources were discarded because of their small (compact) size, 9 were discarded because of the confusion about their class (WAT or NAT), and 24 sources were discarded because of a combination of reasons, for example, low signal-to-noise ratio, low angular extent, and faint extended emission. The remaining 639 sources were given the label 'SMD' in this column. The last two columns of SMC are related to the performance of our DL model trained and tested on SMD, to be discussed in Section 6.

These 639 sources were used to create SMD, our batched dataset for ML. It contains 385 WATs and 254 NATs distributed in 9 training batches, and 1 test batch, as shown in Table 3. Each training batch contains around 39 WATs and 25 NATs, and the test batch contains 38 WATs and 25 NATs. The overall class imbalance in SMD is approximately 66%. SMD is given as a single TAR.GZ (tape archive, GNU zip) file in the Github repository of RGC. SMD contains all the PNG images of our labeled dataset $\mathbf{R}_L$.

4. RGC: our semi-supervised model

RGC combines different existing DL architectures to create a new framework for classifying RAGNs. It was first described by Hossain et al. (2023), or H23, where semi-supervised learning (SSL) methods were used to classify RAGNs into FRI and FRII types (see their Sections 3 and 4 for a detailed background). Here we use a refined version of RGC for classifying RAGNs into WATs and NATs and publish its Version 1.0 as a python package for the first time.

Slijepcevic et al. (2022a,b) showed early examples of using SSL for FR classification. They used the contrastive learning method SimCLR (Chen et al. 2020), the self-supervised method BYOL (Grill et al. 2020), and the semi-supervised method Fix-

[12] https://python-pillow.org/

Table 2: The Sasmal-ML Catalog (SMC), created after a careful examination of the Sasmal catalog, for efficient use in ML. Only the first 3 WATs and the first 3 NATs are shown here for demonstrating the format of the table; the complete catalog of 703 objects can be found in our Github repository as a `TSV` file. Our Sasmal ML-Dataset (SMD) was created following this SMC.

RI	SI	FCG	RA	Dec	Label	Batch	Dilation	$T_{\rm isl}$	$T_{\rm pix}$	Saikia	Asad	Hossain	Attention	Prediction
0	1	...	...	...	WAT		0	3	5	...	...	SMD	...	...
1	2	...	...	...	WAT		0	3	5	...	...	SMD	...	...
2	3	...	...	...	WAT		0	3	5	...	...	SMD	...	...
422	1	...	...	...	NAT		0	3	5	...	...	SMD	...	...
423	2	...	...	...	NAT		0	3	5	...	...	SMD	...	...
424	3	...	...	...	NAT		0	3	5	...	...	discard	...	...

Notes. Description of the 15 columns: (1) RI is RGC Index used in SMD, (2) SI is the original Sasmal Index in Sasmal et al. (2022b), (3) FCG stands for FIRST Component Group or FIRST Catalog Galaxy, which is a common prefix for the IAU-compliant names of these objects, (4–5) RA and Dec are right ascension and declination, (6) Label is the label given in the Sasmal catalog, (7) Batch is the batch number of the object in our SMD, (8–10) Dilation and the next two threshold parameters used in `PyBDSF`, (11–13) Saikia, Asad and Hossain contain the comments of these authors, (14–15) Attention and Prediction contain comments about the quality of the attention and prediction of our current DL model trained and tested on SMD. Cells marked with '...' are omitted here for brevity; complete entries are available in the online catalog.

Table 3: Distribution of WATs and NATs in our Sasmal-ML Dataset (SMD) and the labeled dataset $\mathbf{R}_L$, created following the Sasmal-ML Catalog (SMC) demonstrated in Table 2.

Type	Batch	WATs	NATs	Batch total	Type total
Train	0	39	25	64	576
	1	39	25	64	
	2	39	25	64	
	3	39	25	64	
	4	39	25	64	
	5	38	26	64	
	6	38	26	64	
	7	38	26	64	
	8	38	26	64	
Test	9	38	25	63	63
Total	10	385	254	639	639

Match (Sohn et al. 2020). H23 used SimCLR and BYOL in combination with an $E(2)$-steerable CNN (E2CNN; Weiler & Cesa 2019; Scaife & Porter 2021), and found BYOL to outperform SimCLR. Therefore, in this work, we include only BYOL with E2CNN in RGC 1.0. Both BYOL and E2CNN have been described sufficiently in the relevant papers, and in H23. Hence, below we focus mainly on the experimental setup of RGC 1.0 relevant for this work.

As described in Section 5 of H23, RGC has two stages: an unsupervised (or, equivalently, self-supervised) feature extraction, and a supervised fine-tuning. In the first stage, we pre-train RGC using BYOL on the unlabeled dataset $\mathbf{R}_U$. Here BYOL contains a G-CNN as an encoder, which preserves invariance under different transformations. In the second stage, we fine-tune the pre-trained model on the labeled dataset $\mathbf{R}_L$ (given as SMD) using supervised learning. So the first stage is task-agnostic, and the second stage task-specific where classification is performed as a downstream task. In Figure **??**, we show the first stage on the left, and the second stage on the right.

4.1. Self-supervised pre-training

The first stage of RGC is depicted on the left panel of Figure **??**. It shows how BYOL learns a representation $y(\theta)$ that can be used for downstream tasks. Its online (top row) and target (bottom row) networks are similar, but asymmetric, and they have different weights θ and ξ, respectively. The online network has three stages: an encoder f_θ, a projector g_θ and a predictor q_θ. In our case E2CNN was used as the encoder f_θ. BYOL uses its target network as regression 'targets' for training the online network. The process is shown from the left to the right of this panel. Given an image x, BYOL produces its two views v and v' by applying different augmentations t and t'. From v, the online network creates a representation y_θ and a projection z_θ; the target network produces the corresponding y'_ξ and z'_ξ from v'. Then the online network produces a prediction $q_\theta(z_\theta)$. Subsequently l_2-normalization is applied to the online prediction and the target projection. The difference between these two l_2-normalized vectors is the loss function of BYOL, defined as (Equation 2 of Grill et al. 2020)

$$\mathcal{L}^{\rm BYOL}_{\theta,\xi} = \|\overline{q_\theta}(z_\theta) - \overline{z}'_\xi\|_2^2 \tag{8}$$

which is a mean-squared error (MSE). At each training step, a stochastic optimization is performed to minimize the loss with respect to θ, but not ξ, which is shown by the stop-gradient (`sg`) in the left panel of Fig. **??**.

In case of RGC, the architecture of the encoder f_θ is based on E2CNN, which is designed to preserve invariance under transformations such as rotation and reflection, the Euclidean group $E(2)$. The model consists of multiple layers of group CNNs (G-CNN) and pooling operations, shown on the right panel of Figure **??**. It mimics the architecture of AlexNet (Krizhevsky et al. 2012), which contains five convolutional layers followed by three fully connected (FC) layers. Instead of using three FC layers, we used one in E2CNN.

4.2. Supervised fine-tuning

The second stage of RGC involves fine-tuning the pre-trained encoder on a labeled dataset of bent radio AGNs to classify them into WAT and NAT galaxies. The fine-tuning process involves training the model on the labeled dataset using supervised learning techniques to improve its performance on the classification

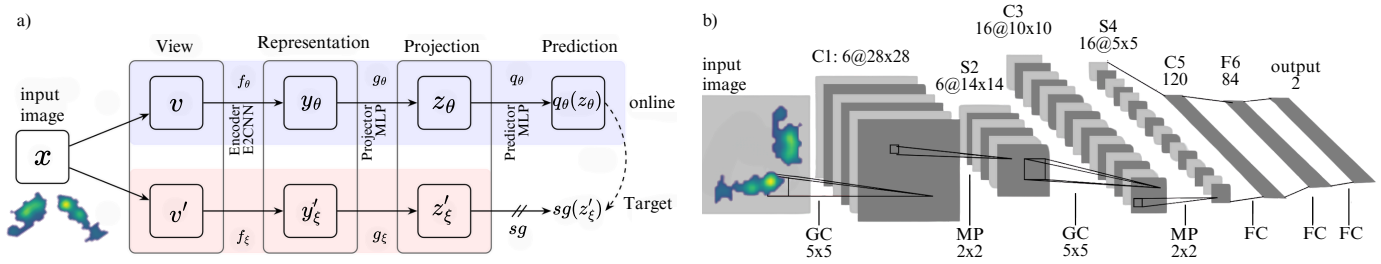


Fig. 3: a)BYOL architecture b) E2CNN architecture.

task. The model is trained to minimize the cross-entropy loss between the predicted and true labels of the input data. The mathematical formulation of the cross-entropy loss is given by:

$$L_{\text{cross-entropy}} = -\sum_{i=1}^{N} y_i \log(p_i), \tag{9}$$

where y_i is the true label of the input data, p_i is the predicted probability of the input data belonging to the true class, and N is the number of classes. The model is trained to minimize the cross-entropy loss by updating the parameters of the encoder using backpropagation. The fine-tuning process enables the model to learn the discriminative features of the input data and improve its performance on the classification task. The fine-tuning process is illustrated in Fig.

4.3. Implementation Details

Table 4: Stage 1: Unsupervised BYOL Training Hyperparameters

Parameter	Value	Notes
Learning Rate	3×10^{-4}	Constant, no scheduler
Batch Size	16	
Epochs	500	No early stopping
Workers	16	Data loading processes
Device	CUDA	GPU training
Optimizer	Adam	No weight decay
Loss Function	Cosine similarity	$\mathcal{L} = 2 - 2 \cdot \cos(\cdot)$

Table 5: Stage 2: Supervised Fine-tuning Hyperparameters

Parameter	Value	Notes
Learning Rate	3×10^{-4}	Initial LR
Weight Decay	1×10^{-6}	Adam optimizer
Batch Size	16	
Epochs	500	Early stopping on validation accuracy
Workers	64	Data loading processes
Device	CUDA	GPU training
Optimizer	Adam	With weight decay
Scheduler	ReduceLROnPlateau	factor=0.9, patience=2, mode=min
Loss Function	NLLLoss + L2	$+0.1\times\|\theta\|_2^2$ regularization

5. Performance of RGC

In this section, we will evaluate the performance of our proposed model on the classification of bent radio AGNs. We have used the following metrics to evaluate the performance of our model: accuracy, precision, recall, and F1-score. In addition to these metrics, we have also used the Receiver Operating Characteristic (ROC) curve, Area Under the Curve (AUC) score, and Expected Calibration Error (ECE) to evaluate the discriminative ability and reliability of our model.

5.1. Confusion matrix

To visualize the output of the classification model in test data, we used a confusion matrix. The confusion matrix is a table that is often used to describe the performance of a classification model on a set of test data for which the true values are known. The confusion matrix shows four values: true positives (TP), true negatives (TN), false positives (FP), and false negatives (FN). The TP indicates the number of positive samples (i.e., WAT galaxies) that were correctly classified as positive, while TN indicates the number of negative samples (i.e., NAT galaxies) that were correctly classified as negative. FP indicates the number of negative samples that were incorrectly classified as positive, while FN indicates the number of positive samples that were incorrectly classified as negative. The confusion matrix for the classification of bent radio AGNs is shown in Fig. 4.

The confusion matrix shows that, without spurious sources, the model correctly classified 35 out of 38 WAT galaxies as WAT and 21 out of 25 NAT galaxies as NAT. Misclassifications include 3 WAT galaxies labeled NAT and 4 NAT galaxies labeled WAT. When spurious sources are included, the model maintains similar performance for WAT galaxies (35/38 correctly classified), while identifying 20 out of 25 NAT galaxies. In this case, 3 WAT galaxies are misclassified as NAT and 5 NAT galaxies as WAT.

5.2. Classification metrics

Based on the output shown in the confusion matrix, we calculated the following classification metrics: accuracy, precision, recall, and F1-score. The accuracy of the model is the proportion of correctly classified samples to the total number of samples. The precision of the model is the proportion of TPs to the sum of TPs and FPs, it measures how many of the samples classified as positive are actually positive. We can calculate the precision using the following formula:

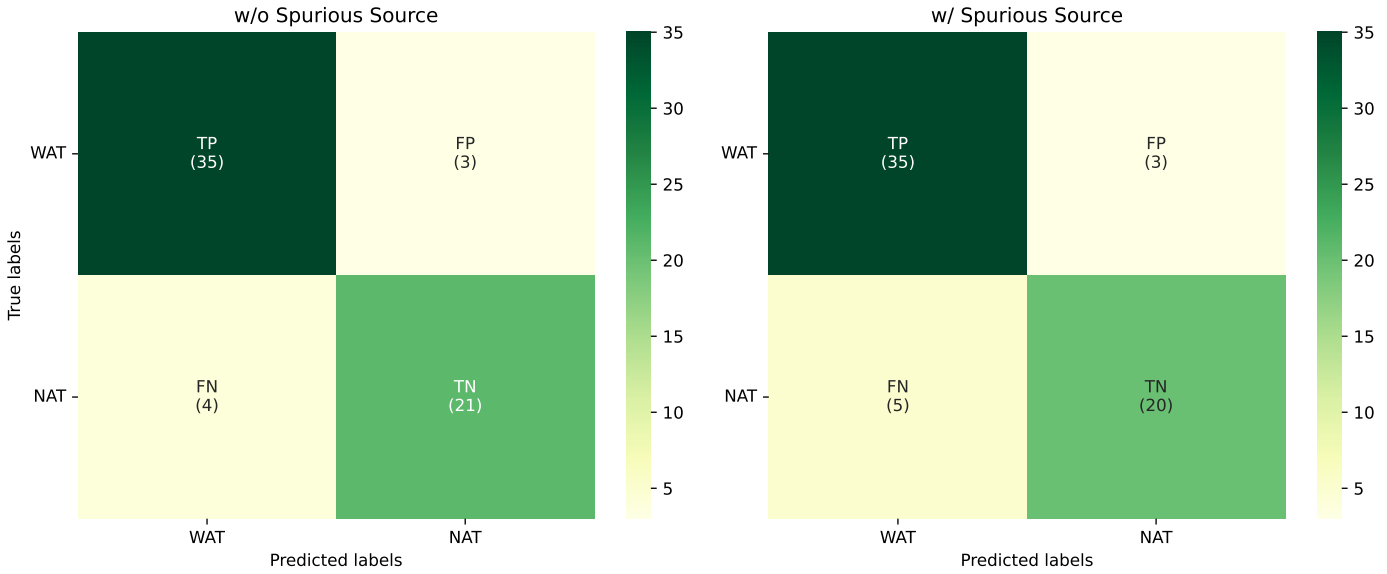


Fig. 4: Confusion matrix for the classification of bent radio AGNs, showing the number of true positives (TP), true negatives (TN), false positives (FP), and false negatives (FN).

$$\text{Precision} = \frac{TP}{TP + FP}$$

The recall of the model is the proportion of TPs to the sum of TPs and FNs, it measures how many of the actual positive samples are correctly classified as positive. We can calculate the recall using the following formula:

$$\text{Recall} = \frac{TP}{TP + FN}$$

The F1-score is the harmonic mean of precision and recall, it is a measure of the balance between precision and recall. We can calculate the F1-score using the following formula:

$$\text{F1-score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

We calculated the accuracy, and class-specific precision, recall, and F1-score for the classification of bent radio AGNs, and reported the results in Table 6.

Our model achieved an accuracy of 87.5% on the classification task with spurious sources, with a precision of 0.875, recall of 0.9211, and F1-score of 0.8974 for WAT galaxies. For NAT galaxies, the model achieved a precision of 0.8696, recall of 0.80, and F1-score of 0.833. Without spurious sources, our model achieved 88.9% accuracy. with precision of 0.8974 precision, 0.9211 recall and 0.9091 F1-score for WAT galaxies. And 0.875 precision 0.840 recall and 0.8571 F1-score for NAT galaxies. These results indicate the effectiveness of our proposed model in classifying bent radio AGNs.

5.3. Discriminative ability

In addition to the results obtained from the classification metrics, we further evaluated the discriminative ability of our model using ROC curves and the corresponding AUC scores. The ROC curve is a graphical representation of the performance of a binary classifier system. It plots the True Positive Rate (TPR) against the False Positive Rate (FPR) at various threshold settings (i.e., different probability values for classifying a sample as positive). The TPR and FPR are defined as follows:

$$\text{TPR} = \frac{TP}{TP + FN}, \quad \text{FPR} = \frac{FP}{FP + TN}$$

The ROC curve shows the trade-off between correctly catching positive cases and mistakenly identifying negative cases across all classification thresholds. While the ROC curve is a useful tool for visualizing the performance of a classifier, the Area Under the Curve (AUC) score provides a single numeric value to represent the classifier's performance. It ranges from 0 to 1, with a score of 1 indicating a perfect classifier and a score of 0.5 indicating a classifier that performs no better than random. The ROC curves for the classification of bent radio AGNs are shown in Fig. 5, along with the AUC score for WAT and NAT galaxies.

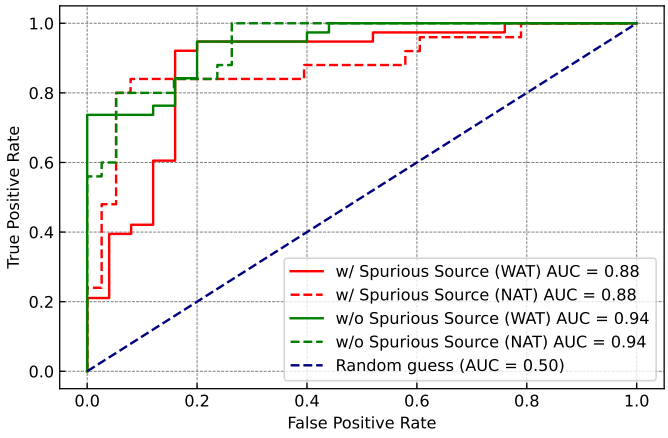


Fig. 5: Class-specific ROC curves for the classification of bent radio AGNs, showing the Area Under the Curve (AUC) score for WAT and NAT galaxies.

The ROC curves for both WAT and NAT galaxies are way above the random guess line, and close to the top-left corner

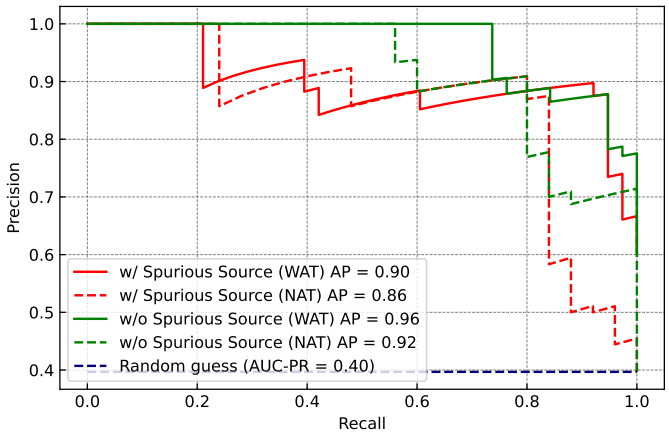


Fig. 6: Precision-Recall (PR) curves for the classification of bent radio AGNs, showing the Area Under the Curve (AUC-PR) score for WAT and NAT galaxies.

of the plot (i.e, curve of perfect classifier), indicating that our model has a high discriminative ability. The AUC score further confirms that with spurious sources, a numerical value of 0.88 for both WAT and NAT galaxies, and without spurious sources 0.94 for both WAT and NAT galaxies.

These high AUC scores indicate that our model can effectively distinguish between WAT and NAT galaxies with a minimal rate of false positives, irrespective of the classification threshold.

ROC curve and AUC score can sometimes be misleading, especially when the dataset is imbalanced. In our case, the dataset is imbalanced, with more WAT galaxies than NAT galaxies. To address this issue, another popular metric called the Precision-Recall (PR) curve and the corresponding AUC-PR score can be used. The PR curve plots the precision against the recall at various threshold settings. The PR curve is particularly useful when the dataset is imbalanced, as it focuses on the positive class (i.e., WAT galaxies) and provides a more informative view of the classifier's performance. By focusing on the performance of the positive class, the PR curve helps to provide a clearer understanding of how well the classifier identifies galaxies in an imbalanced dataset. Similar to the ROC curve, the area under the PR curve (AUC-PR) provides a single numeric value to represent the classifier's performance. The AUC-PR ranges from 0 to 1, with 1 indicating a perfect classifier. A higher AUC-PR indicates better overall performance. The PR curves for the classification of bent radio AGNs are shown in Fig. 8, along with the AUC-PR score for WAT and NAT galaxies.

The PR curves for WAT and NAT galaxies is above the horizontal line at 0.40, which indicates that the model is performing better than a random classifier. The AUC-PR score for WAT galaxies is 0.90 and for NAT galaxies is 0.86, with spurious sources. And without spurious sources, the score for WAT galaxies is 0.96 and that for NAT galaxies is 0.92. This indicates that the model has good discriminative ability for both classes, despite the class imbalance in the data set.

5.4. Model calibration

Model calibration is an important aspect of evaluating the reliability of a classifier. The aim is to align the predicted probabilities of the model with the true probabilities of the data to ensure that the model's predictions are reliable and accurate. One way to evaluate the calibration of a classifier is to use the Expected Calibration Error (ECE) metric. To evaluate the calibration of our model, we calculated the ECE for the classification of bent radio AGNs. The ECE is a measure of the difference between the predicted probabilities and the true probabilities of the model. It is calculated by dividing the samples into M equally spaced bins based on the predicted probabilities and then calculating the difference between the average predicted probability and the true probability for each bin. The ECE is the weighted average of these differences, with the weights being the proportion of samples in each bin.

$$\mathrm{ECE} = \sum_{m=1}^{M} \frac{B_m}{N} \left|\mathrm{acc}(B_m) - \mathrm{conf}(B_m)\right|$$

where B_m is the number of samples in bin m, N is the total number of samples, $\mathrm{acc}(B_m)$ is the accuracy of the model in bin m, and $\mathrm{conf}(B_m)$ is the confidence of the model in bin m. The accuracy and confidence of the model in bin m are calculated as follows:

$$\mathrm{acc}(B_m) = \frac{1}{B_m} \sum_{i \in B_m} \mathbb{I}(\hat{y}_i = y_i), \quad \mathrm{conf}(B_m) = \frac{1}{B_m} \sum_{i \in B_m} p_i$$

where y_i is the true label of sample i, $\hat{y}_i$ is the predicted label of sample i, p_i is the predicted probability of sample i, and $\mathbb{I}$ is the indicator function. The ECE ranges from 0 to 1, with a lower ECE indicating better calibration. The ECE for the classification of bent radio AGNs is shown in Fig. 7.

The ECE plot shows that the model is well calibrated, with the calibration curve lying close to the diagonal. The expected calibration error (ECE) is approximately 0.150 for both WAT and NAT galaxies without spurious sources, and increases to 0.204 for both classes with spurious sources.

This indicates that the model's predicted probabilities are close to the true probabilities, and the model is reliable in its predictions.

All these results indicate that our proposed model is effective in classifying bent radio AGNs, with high accuracy, precision, recall, and F1-score, good discriminative ability, and reliable calibration.

5.5. Comparative analysis

Table 6 presents a comprehensive comparison of the performance of different baseline models along with the proposed RGC models. The evaluation is carried out using accuracy as the overall metric and precision, recall, and F1-score for both WAT and NAT classes to capture class-specific behavior.

Among the baseline models, ConvNeXT demonstrates the best overall performance with an accuracy of 84.12%. It achieves strong results in both WAT (F1-score of 0.87) and NAT (F1-score of 0.79), indicating balanced performance across categories. SWIN-B also performs competitively, achieving 80.95% accuracy with a relatively high WAT recall (0.92), although its NAT recall is lower (0.64), showing that it tends to favor WAT predictions.

ResNet-50 and ViT-B-16 provide moderate performance, with accuracy values of 77.77% and 76.19%, respectively. While ViT-B-16 achieves a balanced trade-off between WAT and NAT metrics (F1-scores of 0.82 and 0.74), ResNet-50 shows a slightly better WAT recall but lower NAT precision. In contrast, VGG-16

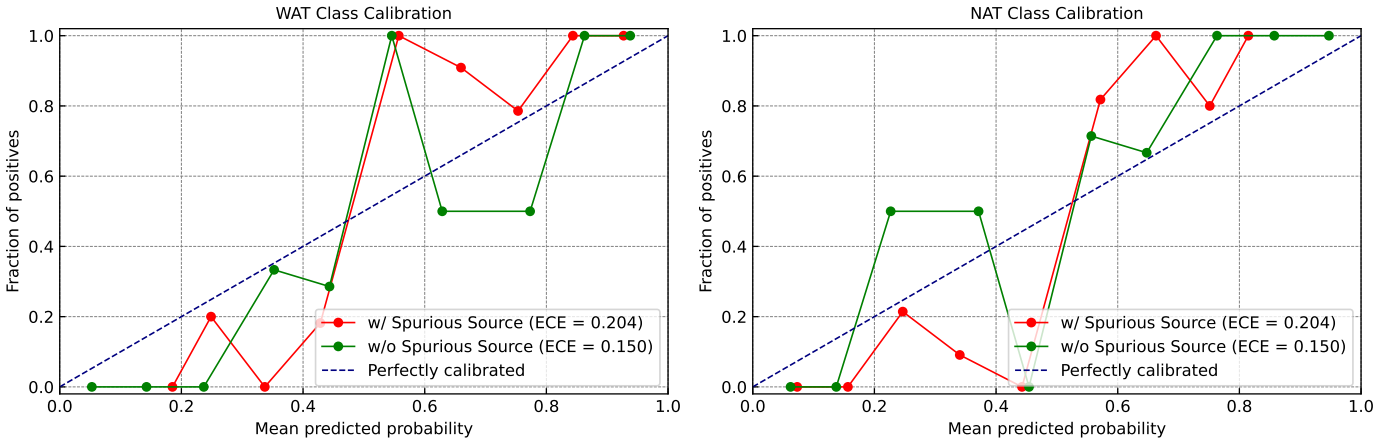


Fig. 7: Expected Calibration Error (ECE) for the classification of bent radio AGNs, showing the ECE curve for WAT and NAT galaxies across different predicted probabilities.

Model	Accuracy [%]	WAT			NAT		
		Precision	Recall	F1-score	Precision	Recall	F1-score
RGC (w/o spurious)	88.9	0.8974	0.9211	0.9091	0.8750	0.8400	0.8571
RGC (w/ spurious)	87.3	0.8750	0.9211	0.8974	0.8696	0.8000	0.8333
ConvNeXT	84.12	0.85	0.89	0.87	0.83	0.76	0.79
SWIN-B	80.95	0.80	0.92	0.85	0.84	0.64	0.73
ResNet-50	77.77	0.79	0.87	0.82	0.76	0.64	0.70
ViT-B-16	76.19	0.76	0.89	0.82	0.77	0.73	0.74
VGG-16	74.60	0.73	0.92	0.81	0.80	0.48	0.60

Table 6: Quantitative evaluation (performance comparison) of different deep learning models on the bent radio AGN dataset. The table presents the accuracy of the models, along with class-wise precision, recall, and F1-score for NAT and WAT sources. RGC (w/o spurious) achieves the highest accuracy and F1-scores across both WAT and NAT, indicating its effectiveness in identifying both morphologies.

lags behind the other baselines, obtaining the lowest overall accuracy (74.60%) and particularly poor NAT recall (0.48), which limits its effectiveness in distinguishing NAT samples.

The proposed RGC models outperform all baselines. RGC without spurious correlations achieves the highest overall accuracy (88.90%) and superior F1-scores for both WAT (0.91) and NAT (0.86), highlighting its effectiveness in capturing both morphologies. RGC trained with spurious correlations also performs strongly (87.30% accuracy), but its NAT recall (0.80) and F1-score (0.83) are slightly lower than the spurious-free version, demonstrating that removing spurious features improves generalization and class balance.

Overall, these results confirm that the RGC framework is highly effective at identifying bent radio AGN morphologies, particularly when trained without spurious correlations, outperforming conventional CNN and Transformer architectures across both global and class-specific metrics.

6. Discussion

$\mathbf{X}_u$ is determined by experiments described in Sec 6.2.

6.1. Where is the the model giving attention? DONE

success story: 1 wat, 1 nat single source 2-panel plot

failure: multiple source what percentage

failure: spurious source brighter, what percentage

brightness vs. attention

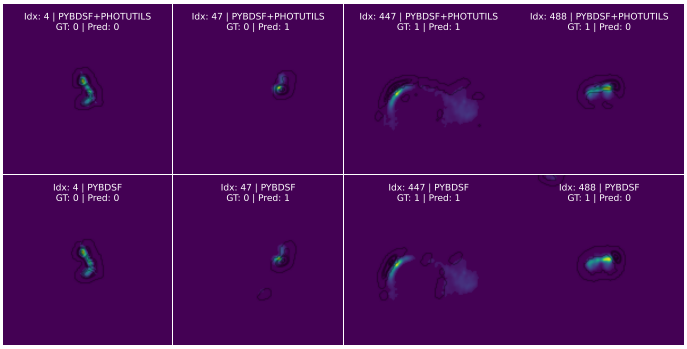


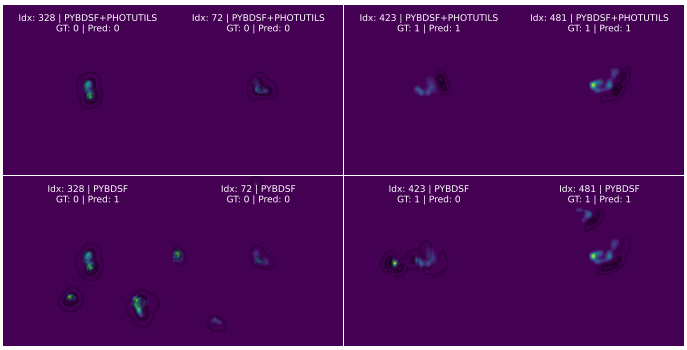
Fig. 8: Row 1 shows images with spurious sources; Row 2 shows the same images without spurious sources, both highlights the model's attention.

In Section 6.1, we explore where the model is directing its attention and how it performs in different scenarios. A success story demonstrates the model's effectiveness with a 1-WAT, 1-NAT, single-source 2-panel plot, showcasing its ability to focus accurately under ideal conditions. However, in failure cases, the model struggles when multiple sources are present, with the performance dropping by `[insert percentage]`. Furthermore, in

the case of a spurious source being brighter, the model's attention mechanism tends to be misled, with the failure rate being [insert percentage]. A key observation is that there is a noticeable relationship between brightness and attention, which could provide insights into how the model prioritizes certain features over others during the task.

6.2. Do spurious sources affect performance? DONE

Check where the model is paying attention on the image.



Use this info in Sec. 5

In this section, we investigate the effect of spurious sources on the model's performance. To analyze this, we first examine the regions where the model is focusing its attention during the image analysis. As discussed in Section 5, [insert the method/approach used to analyze attention]. The results show that when a spurious source is present, [insert performance metric, e.g., accuracy, F1 score, etc.] [increased/decreased] by [insert percentage or other relevant statistic]. This behavior can be attributed to [insert analysis or observation], suggesting that the model is often misled by the brightness of the spurious source, [insert relevant observation about the model's performance under these conditions]. The attention heatmap generated during this process further confirms that the model tends to concentrate on [insert specific areas or features the model pays attention to], which may explain the degradation in performance.

6.3. How does class imbalance affect model performance?

Plot: accuracy vs. imbalance %

$\mathbf{X}_l$

We did not do this part

In this section, we examine how class imbalance influences the performance of the model. Class imbalance refers to the unequal distribution of classes in a dataset, which can significantly impact the model's ability to correctly predict the minority class.

7. Conclusion

Data and Code Availability

Contributor Roles Taxonomy (CRediT)

The first three authors contributed most significantly and are listed first. Author affiliations are presented in order of primacy for each author. Here we describe the author contributions using the taxonomy given in Table 1 of Brand et al. 2015. *Hossain*: Methodology, Software, Data Curation, Investigation; *Asad*: Conceptualization, Data Curation, Investigation,

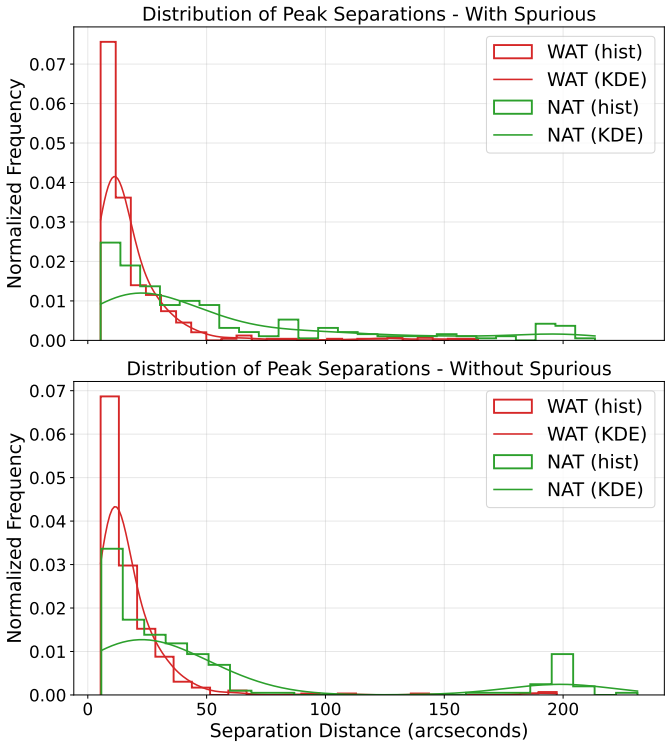


Fig. 9: Distribution of peak separations between the two most prominent features in radio galaxy Grad-CAM heatmaps. Top panel shows sources with spurious components, and bottom panel shows sources without spurious components. Step histograms represent WAT (blue) and NAT (orange) classes, while the overlaid curves show Gaussian KDE estimates of the distributions.

Writing - Original Draft; *Saikia*: Data Curation, Writing - Review and Editing; *Shahal*: Visualization, Editing; *Khan*: Visualization, Data Curation, Data Annotation, Editing; *Akter*: Data Curation; *Ali*: Supervision; *Amin*: Project administration, Funding acquisition; *Momen*: Resources; *Rahman*: Supervision, Conceptualization.

References

Alam, S., Albareti, F. D., Allende Prieto, C., et al. 2015, ApJS, 219, 12
Banfield, J. K., Wong, O. I., Willett, K. W., et al. 2015, MNRAS, 453, 2326
Becker, R. H., White, R. L., & Helfand, D. J. 1995, ApJ, 450, 559
Best, P. N. & Heckman, T. M. 2012, MNRAS, 421, 1569
Bhukta, N., Mondal, S. K., & Pal, S. 2022, MNRAS, 516, 372
Bradley, L., Sipőcz, B., Robitaille, T., et al. 2025, astropy/photutils: 2.2.0
Brand, A., Allen, L., Altman, M., Hlava, M., & Scott, J. 2015, Learned Publishing, 28, 151
Bulbul, E., Liu, A., Kluge, M., et al. 2024, A&A, 685, A106
Chen, T., Kornblith, S., Norouzi, M., & Hinton, G. 2020, in Proceedings of the 37th International Conference on Machine Learning, ICML'20 (JMLR.org)
Conselice, C. J., Wilkinson, A., Duncan, K., & Mortlock, A. 2016, ApJ, 830, 83
Dehghan, S., Johnston-Hollitt, M., Franzen, T. M. O., Norris, R. P., & Miller, N. A. 2014, AJ, 148, 75
Fanaroff, B. L. & Riley, J. M. 1974, MNRAS, 167, 31P
Garon, A. F., Rudnick, L., Wong, O. I., et al. 2019, AJ, 157, 126
Golden-Marx, E., Blanton, E. L., Paterno-Mahler, R., et al. 2021, ApJ, 907, 65
Grill, J.-B., Strub, F., Altché, F., et al. 2020, in Proceedings of the 34th International Conference on Neural Information Processing Systems, NIPS '20 (Red Hook, NY, USA: Curran Associates Inc.)
Hossain, M. S., Roy, S., Asad, K., et al. 2023, Procedia Computer Science, 222, 601, international Neural Network Society Workshop on Deep Learning Innovations and Applications (INNS DLIA 2023)
Kapoor, R. 2023, Journal of Astronomical History and Heritage, 26, 411

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.)
Kurcz, A., Bilicki, M., Solarz, A., et al. 2016, A&A, 592, A25
Lao, B., Andernach, H., Yang, X., et al. 2025, ApJS, 276, 46
Maitra, D. 2021, in 110th Annual Meeting of the American Association of Variable Star Observers (AAVSO)
Merloni, A., Lamer, G., Liu, T., et al. 2024, A&A, 682, A34
Miley, G. K., Perola, G. C., van der Kruit, P. C., & van der Laan, H. 1972, Nature, 237, 269
Mingo, B., Croston, J. H., Hardcastle, M. J., et al. 2019, MNRAS, 488, 2701
Mohan, N. & Rafferty, D. 2015, PyBDSF: Python Blob Detection and Source Finder, Astrophysics Source Code Library, record ascl:1502.007
Ndung'u, S., Grobler, T., Wijnholds, S. J., Karastoyanova, D., & Azzopardi, G. 2023, New A Rev., 97, 101685
Norris, R., Basu, K., Brown, M., et al. 2015, in Advancing Astrophysics with the Square Kilometre Array (AASKA14), 86
Norris, R. P. 2017, Nature Astronomy, 1, 671
Norris, R. P., Marvil, J., Collier, J. D., et al. 2021, PASA, 38, e046
O'Dea, C. P. & Baum, S. A. 2023, Galaxies, 11, 67
Owen, F. N. & Rudnick, L. 1976, ApJ, 205, L1
Pal, S. & Kumari, S. 2023, Journal of Astrophysics and Astronomy, 44, 17
Porter, F. A. M. & Scaife, A. M. M. 2023, RAS Techniques and Instruments, 2, 293
Predehl, P., Andritschke, R., Arefiev, V., et al. 2021, A&A, 647, A1
Proctor, D. D. 2011, ApJS, 194, 31
Rudnick, L. & Owen, F. N. 1976, ApJ, 203, L107
Sasmal, T. K., Bera, S., Pal, S., & Mondal, S. 2022a, ApJS, 259, 31
Sasmal, T. K., Bera, S., Pal, S., & Mondal, S. 2022b, VizieR Online Data Catalog: Head-tail radio galaxies from the VLA FIRST survey (Sasmal+, 2022), VizieR On-line Data Catalog: J/ApJS/259/31. Originally published in: 2022ApJS..259...31S
Scaife, A. M. M. & Porter, F. 2021, MNRAS, 503, 2369
Shimwell, T. W., Hardcastle, M. J., Tasse, C., et al. 2022, A&A, 659, A1
Slijepcevic, I. V., Scaife, A., Walmsley, M., & Bowles, M. R. 2022a, in Machine Learning for Astrophysics, 53
Slijepcevic, I. V., Scaife, A. M. M., Walmsley, M., et al. 2022b, MNRAS, 514, 2599
Slijepcevic, I. V., Scaife, A. M. M., Walmsley, M., et al. 2024, RAS Techniques and Instruments, 3, 19
Sohn, K., Berthelot, D., Li, C.-L., et al. 2020, in Proceedings of the 34th International Conference on Neural Information Processing Systems, NIPS '20 (Red Hook, NY, USA: Curran Associates Inc.)
Urry, C. M. & Padovani, P. 1995, 107, 803
Vardoulaki, E., Backöfer, V., Finoguenov, A., et al. 2025, A&A, 695, A178
Weiler, M. & Cesa, G. 2019, General E(2)-equivariant steerable CNNs (Red Hook, NY, USA: Curran Associates Inc.)
Wong, O. I., Garon, A. F., Alger, M. J., et al. 2025, MNRAS, 536, 3488