Publications of the Astronomical Society of the Pacific, 131:108009 (23pp), 2019 October https://doi.org/10.1088/1538-3873/ab150b

Radio Galaxy Zoo: Knowledge Transfer Using Rotationally Invariant Self-organizing Maps

T. J. Galvin[1,2], M. Huynh[1,3], R. P. Norris[2,4], X. R. Wang[5], E. Hopkins[6], O. I. Wong[3], S. Shabala[7], L. Rudnick[8], M. J. Alger[9,10], and K. L. Polsterer[6]

[1] CSIRO Astronomy and Space Science, PO Box 1130, Bentley WA 6102, Australia
[2] Western Sydney University, Penrith Campus, Locked Bag 1797, Penrith NSW 2751, Australia
[3] International Centre for Radio Astronomy Research (ICRAR), M468, The University of Western Australia, 35 Stirling Highway, Crawley, WA 6009, Australia
[4] CSIRO Astronomy and Space Science, PO Box 76, Epping, NSW 1710, Australia
[5] CSIRO Data61, Australia, Corner of Vimiera and Pembroke Roads, Marsfield NSW 2122, Australia
[6] Astroinformatics, HITS gGmbH, Schloss-Wolfsbrunnenweg 35, 69118 Heidelberg, Germany
[7] School of Natural Sciences, University of Tasmania, Private Bag 37, Hobart, Tasmania 7001, Australia
[8] Minnesota Institute for Astrophysics, University of Minnesota, Minneapolis, MN 55455, USA
[9] Research School of Astronomy and Astrophysics, The Australian National University, Canberra, ACT 2611, Australia
[10] Data61, CSIRO, Canberra, ACT 2601, Australia
Received 2018 July 13; accepted 2019 March 31; published 2019 September 9

Abstract

With the advent of large scale-surveys the manual analysis and classification of individual radio source morphologies is rendered impossible as existing approaches do not scale. The analysis of complex morphological features in the spatial domain is a particularly important task. Here, we discuss the challenges of transferring crowdsourced labels obtained from the Radio Galaxy Zoo project and introduce a proper transfer mechanism via quantile random forest regression. By using parallelized rotation and flipping invariant Kohonen-maps, image cubes of Radio Galaxy Zoo selected galaxies formed from the Faint Images of the Radio Sky at Twenty-cm (FIRST) radio continuum and the *Wide-field Infrared Survey Explorer* (*WISE*) infrared all-sky surveys are first projected down to a two-dimensional embedding in an unsupervised way. This embedding can be seen as a discretized space of shapes with the coordinates reflecting morphological features as expressed by the automatically derived prototypes. We find that these prototypes have reconstructed physically meaningful processes across two channel images at radio and infrared wavelengths in an unsupervised manner. In the second step, images are compared with those prototypes to create a heat map, which is the morphological fingerprint of each object and the basis for transferring the user generated labels. These heat maps have reduced the feature space by a factor of 248, and are able to be used as the basis for subsequent machine-learning (ML) methods. Using an ensemble of decision trees we achieve upwards of 85.7% and 80.7% accuracy when predicting the number of components and peaks in an image, respectively, using these heat maps. We also question the currently used discrete classification schema and introduce a continuous scale that better reflects the uncertainty in transition between two classes, caused by sensitivity and resolution limits.

Key words: galaxies: general – galaxies: jets – galaxies: statistics – radio continuum: general – infrared: general

Online material: color figures

1. Introduction

Radio astronomy is on the verge of a new age as the next generation of instruments nears completion (Norris 2017a). These new instruments offer improvements in sensitivity, fractional bandwidth coverage and survey speed offering orders of magnitude improvement over conventional instruments, enabling us to unlock and explore a younger universe.

Associating a radio source with a single, intrinsic object across multi-wavelength domains is a difficult problem. For example, different wavelengths can trace different physical emission mechanisms, which may not necessarily be localized in a single, compact region. Radio lobes of active galactic nuclei (AGNs) may be separated by some distance from the super massive black hole accreting and ejecting matter, while only the host galaxy is seen at optical and infrared wavelengths. For such sources, it is important to correctly associate these physically separate components spanning different wavelength domains to extract the maximum level of scientific knowledge. For instance, without multi-wavelength data the radio lobes of an AGN may be confused with two nearby, unrelated star-forming

galaxies (SFG). A challenge is that the spatial resolution and sensitivity may be insufficient to separate them into distinct independent classes.

This problem is further exacerbated by the fact that different telescopes, and the data that they produce, have different characteristics and limitations. For instance, optical and infrared surveys typically have higher resolution than radio surveys. As a result, an infrared image may show many objects in the vicinity of a single radio object. Differing sensitivity limits may also influence the number of objects detected, and make identifying associated components harder, particularly if they are faint or missing in a subset of images.

Automated algorithms for such problems are in their infancy. Although near-neighbor matching algorithms are generally robust for unresolved objects, the problem is more difficult to solve reliably for complex morphologies (Alger et al. 2018). It is estimated that 10% of the 70 million objects to be detected by Evolutionary Map of the Universe (EMU) will be complex objects (Norris et al. 2011) requiring sophisticated methods of cross-identification. Experts in the domain area (i.e., astronomers) will be unable to maintain pace to manually inspect every instance.

With the advent of high-performance computing platforms, commodity computing hardware is now capable of solving the problem. In particular, machine-learning (ML) algorithms offer powerful avenues for both supervised and unsupervised data processing, classification and analysis, with applications ranging from photometric redshift estimation (Luken et al. 2019; Norris et al. 2019), star classification (Weir et al. 1995), optical transients (Mahabal et al. 2011), and simple/complex object discrimination (Segal et al. 2019; Lukic et al. 2018). Combining currently available large data sets with the affordable computing resources provided by, e.g., graphics processing units (GPUs), opens novel data analysis techniques.

For image classification, considerable progress has been made by using convolution neural networks (CNN). These networks efficiently recognize hierarchical structures through a series of layered convolution functions after an initial training process. Convolutional filters provide positional invariance to the feature location, which is invaluable when attempting to classify galaxies as they appear across the sky. To deal with rotation invariance, spatial transformation layers that implement e.g., chirp z-transformations or data augmentation are common tools.

Several projects have successfully applied CNNs to galaxy classification problems. Aniyan & Thorat (2017) used CNNs to recognize classes of Fanaroff-Riley (FR; Fanaroff & Riley 1974) radio galaxies and radio galaxies with bent tail morphologies. They found success rates upwards of 95% depending on the morphology presented, with bent-tailed radio galaxies being the most distinguishable. Wu et al. (2019) presents a CNN architecture that is capable of recognizing radio source morphologies, with an initial end-to-end classifier that is both fast (<200 milliseconds per image) and accurate (>90%). Alger et al. (2018) compared the performance of a simple CNN, random forests and linear regression in the task of classifying Radio Galaxy Zoo (RGZ) sources. Lukic et al. (2018) train a CNN on four classes of extended and compact sources, achieving an overall classification accuracy of 94.8% on the RGZ Data Release 1 (DR1) data set.

CNNs are an example of a supervised learning method, meaning that data sets with known labels or features have to be provided for the training process to converge and become a successful predictor. For certain problems, this can be a non-trivial requirement, as the known data set has to be sufficiently large and contain adequate sampling of the desired features or labels to be modeled. Building such data sets is often the most troubling task when attempting to utilize CNNs or similar supervised learning methods.

Projects that use crowdsourcing methods to build these training data sets for galaxy classification include Galaxy Zoo (Lintott et al. 2008) and Radio Galaxy Zoo (RGZ; Banfield et al. 2015), both of which are members of the "Zooniverse" portal.[11] These projects provide an online web platform that allows volunteers to interact and label images. Statistics are then built up of each source through a number of independent, non-experts classifications. Although the classifications are performed by the general public, who are acting as citizen scientists and may have no formal astronomy training, the consensus is generally comparable with expert classification.

An alternative approach is to apply unsupervised ML methods which require no training set of labels or features, but instead attempts to construct and optimize a function that is able to describe the *structure* of the data. Unsupervised clustering and dimensionality reduction methods are powerful tools to structure and explore large data sets in such an unsupervised setting (Gianniotis et al. 2016; Traven et al. 2017). Outlier detection and the search for the unexpected can also be considered unsupervised tasks (Crawford et al. 2016; Norris 2017b). Self-organizing Maps (SOM; Kohonen 1982) provide an unsupervised method to automatically derive a latent grid of discrete prototypes, where closeness in the projected space reflects closeness with respect to the used similarity measure. SOMs have been used in the astronomical literature for a variety of tasks, including the classification of light curves (Brett et al. 2004), clustering and analysis of gigahertz-peaked spectrum sources (Torniainen et al. 2008), detecting structure within point data (Way et al. 2011) and object classification and photometric redshift estimation (Geach 2012).

Applying the SOM method onto image data sets requires special consideration. Even though the simple pixel-wise Euclidean distance between two images does not take spatial structures into account, it is already sufficient to order images

[11] https://www.zooniverse.org/

by shape when rotational invariance is not an issue or when images have been aligned to a common orientation as part of some preprocessing stage. As the pixel values reflect locally measured intensities, pairs of objects that have the same shape should have a distance close to zero. With more and more pixels showing significantly different values, the represented shapes change together with the pixel-wise distance.

The Parallelized rotation and flipping INvariant Kohonen-maps (PINK; Polsterer et al. 2016) software framework exploits GPU acceleration and is designed to extend the basic SOM method to operate on image data where the simple rotation of a subject should not be a considered as part of the *structure* of the data. Similar sources are grouped, irrespective of their rotation and mirroring on the sky, which allows the resulting projection to be used to derive the distribution of shapes within the data set and to recognize and separate unusual or rarely seen objects (Crawford et al. 2016).

In this paper, we assess the effectiveness of dimensionality reduction to transfer user generated labels to unseen objects, using data from RGZ.

The paper outline is as follows. In Section 2 we provide a brief description of the SOM algorithm and PINK, together with a description of the applied method to transfer labels. We provide an outline of RGZ and the training data, the preprocessing steps and object labels in Section 3. An overview of the data experiments and the application of PINK is given in Section 4 with corresponding results being reported in Section 5. We finally provide points for discussion and conclusions in Sections 6 and 7.

2. Self-organizing Maps

A SOM is a commonly used algorithm to project high-dimensional data in a low dimensional space and thereby reflect similarity in the original space as distance in the SOM lattice. In the projected SOM lattice space, "closer" refers to data being more similar and "distant" to be very different, without having a strict and formal connection between distance and similarity. SOMs are neural networks that, by being iteratively trained, learns how to arrange the pre-dominantly features in the input data. Importantly, the training phase that produces the transformation to the lower dimensional space is unsupervised, requiring no input labels to accompany the data. By specifying a similarity measure, the notion of distance in the high-dimensional space can be used to, e.g., make a SOM aware of similar shape structures within spatially correlated image data.

The individual cells that make up the SOM lattice are called neurons. Each neuron has its own set of weights, also called prototypes, and are constantly modified during training to adapt for the incoming data to find a generalized representation. Therefore, the neurons are arranged in a low dimensional space, with 2D and 3D lattices being the most common structures.

We provide a brief outline of the individual steps of the SOM algorithm, below:

1. *INITIALIZE*. The prototypes are initialized with, e.g., noise, zeros, predefined patterns/structures, randomly drawn objects from the data set. Alternatively, initialization could be performed with a pre-trained set of prototypes.
2. *FIND BEST MATCH*. An object is taken from the training data set and its distance to all prototypes is calculated. The prototype, known as the best matching unit (BMU), is the one with the minimum distance to the object. Commonly used distance functions are the Euclidean distance, the Manhattan distance, and the Minkowski distances as their generalized counterpart.
3. *MODIFY MAP*. Based on a neighborhood function that is evaluated against neuron positions on the SOM lattice, all prototypes are modified. To realize an exponential decay between neighboring neurons on the SOM lattice and the BMU, often a Gaussian function is selected. For instance, the Gaussian neighborhood function could be constructed as

$$r(n_1, n_2) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{\frac{(n_1 - n_2)^2}{2\sigma^2}}, \tag{1}$$

where n_1 and n_2 are the coordinates of two neurons on the SOM lattice, and σ may be modified between each iteration to focus more and more on a specific region of interest in prototype space to locally constrain the changes during training. In some cases a cyclic distance function that represents a continuous space that wraps around the edges could be considered.

The difference between each prototype and the current training object is calculated and used to modify the prototypes to be more like the current object. Usually, neurons close to the BMU based on the neighborhood function are made more similar to the current training object, than neurons further away. A basic weighting update scheme can be written as

$$w_i' = w_i + (D - w_i) \times r(n_i, BMU) \times \delta, \tag{2}$$

where w_i and w_i' are the weights of the i-th neuron before and after performing the weighting update, D is the currently select item from the training data set, $r(n_i, BMU)$ is the neighborhood function evaluated using the i-th and current BMU neuron, and δ is the additional learning rate dampener that may also evolve across iterations. Its role is to further control the magnitude of the weighting updates between training iterations.
4. *ITERATE*. Repeat steps 2 and 3 for I number of iterations over all objects in your training data set, where I is sufficiently large to allow the SOM to converge.

5. *RECALL*. Once a sufficient number of training iterations have been performed to produce a stable SOM, map all objects in the data set to the derived prototypes to determine the distances to the prototypes and find the region of best match.

The runtime complexity of this algorithm in a naive implementation is $O(I * N * M\log M)$, depending on the number of iterations I, the number of training objects N and the number of prototypes in the map M. When using a GPU, the PINK software is able to operate in roughly $O(I * N * M)$ time when the number of GPU CUDA processing cores exceeds the number of generated images (see Section 2.1). As this algorithm is not guaranteed to converge, it is important to determine the hyperparameters carefully. The number of prototypes M should be sufficiently large to represent the dominant structures in the training data. Too many prototypes will result in a too large computation time, while too few prototypes will cause the map to shuffle without finally settling in a stable state. If the learning rate is too high, the changes during training might be too abrupt, while too small values result in extremely long computations. Too wide neighborhood functions, as used in step 3, change nearly all prototypes, while too narrow distance functions spatially decouples the individual neurons on the SOM lattice and therefore result in a simple clustering that is not reflecting gradual similarity between neighboring prototypes.

2.1. Rotation and Flipping Invariance

PINK offers several distances to be used for the similarity function by offering pre-implemented distances as well as the ability to use functors to implement user-defined distances. For our experiments, we used a modified Euclidean distance. This is a simple metric, measuring the total distance between the pixel intensities of two images, following

$$\Delta(A, B) = \underset{\forall\,\phi\in\Phi}{minimize}(\phi)\ \sqrt{\sum_{c=0}^{C}\sum_{x=0}^{X}\sum_{y=0}^{Y}(A_{c,x,y} - \phi(B_{c,x,y}))^2}\,, \qquad (3)$$

where A and B correspond to a particular neuron and image, c is the corresponding channel, x, y are the coordinates in the image plane and ϕ corresponds to an affine image transformation taken from a set of transformations Φ. In our case this set includes all possible rotations around the center of the image as well as their mirrored counterparts. To avoid empty patches at the corners that are caused by the rotation operation, PINK uses prototypes that are a factor of $\sqrt{2}/2$ smaller than the input images.

As described in Equation (3), the rotation invariance is introduced through a set of affine image transformations Φ. In principle the Euclidean distance is calculated for a set of possible rotations and the best matching angle is determined by finding the angle corresponding to the lowest distance. This job is done in parallel on a GPU hardware, providing the capability to do thousands of similar operations in parallel. This brute-force comparison with all possible rotations ensures a rotation invariance with respect to the alignment morphological features. Therefore the derived prototypes are rotation invariant representations of the data.

PINK does not include a damping functor or learning handicap term that evolves over time, to have a clear separation between the number crunching and the hyperparameter controlling part. Although an early version did implement such a feature (Polsterer et al. 2015), the current version used throughout this study (version `0.23`) requires the user to vary these parameters across multiple invocations of the PINK program. We list each invocation as a separate training stage in Table 2.

2.2. Comparison to Adaptive-subspace SOM

The adaptive-subspace SOM (Kohonen 1996) builds upon the basic SOM algorithm by attempting to learn invariance to affine image transformations throughout the training process. This is done by mapping the training data into a collection of subspaces by applying various transforms, which may include randomly rotating an input image, shifting an image in some direction and zooming into a region of the image. The BMU is located by exploring each subspace and all generated realizations of the transformed data in a "winner take all" manner. Like the basic SOM algorithm, weight updates are shared among neighboring neurons following some neighborhood weighting function. Transforming the data into these subspaces throughout training and mapping constructs a manifold that is able to project the high dimension input data onto a prototype lattice that is robust to affine transforms. Computationally generating and exploring many of these realizations can be an expensive process.

PINK adopts this approach but focuses on rotation and flipping invariance. Within an astronomical context, these two types of transforms are the least significant when identifying objects and should largely be ignored. Indeed, there are numerous object classification schemes which use scale or angular size as a distinguishing feature. Preserving scale information throughout the training process will allow the constructed prototypes to represent more physically meaningful features that conform to existing astronomical morphologically classification schemes. Similarly, feature translation (i.e., location of a feature on an image) is largely a problem that is perhaps best addressed by source finding software. Broadly, these codes are tasked with locating objects in astronomical images to produce a catalog of sources and their properties, including their central position. This task is not trivial as often domain specific knowledge has to be considered, including instrumental point-spread functions, noise properties that may

be correlated among adjacent pixels, coordinate system transforms, and potentially varying wide-field effects.

2.3. Heat Maps as Morphological Fingerprints

Once the SOM has been trained, then a heat map can be produced for any object. A heat map is a matrix of equal dimension to the SOM lattice, whose values are the similarity measure of a source image (or image cube) to each prototype. In the case of PINK, this measure is the modified Euclidean distance (Equation (3)), and reflects the region of space that some image cube resides in the trained SOM out of the set of rotated and flipped images produced internally by PINK. This heat map is a single channel matrix (multi-channel image cubes and neurons are summed in this statistic).

The derived projection to the found prototypes can be seen as a projection to a space of shapes where individual regions represent characteristic morphologies. Therefore the heat maps represent a morphological fingerprint that characterizes the spatial structure of the image with respect to the automatically represented prototypes.

We convert each modified Euclidean distance measure $\Delta(A, B_m)$ at the m-th position of the heat map of a particular object to a likelihood L_m, by first normalizing $\Delta(A, B_m)$ so its minimum is equal to one, then using

$$L_m = \frac{\frac{\Delta(A,B_m)}{\Delta(A,B_m)^{\psi}}}{\sum_{m=1}^{M} \frac{\Delta(A,B_m)}{\Delta(A,B_m)^{\psi}}}, \quad (4)$$

where ψ is a stretching parameter which we nominally set to 10. The purpose of the ψ parameter is to introduce non-linearity in the transform between a simple Euclidean distance metric to a likelihood, where more emphasis is placed on the pixels with a smaller Euclidean distance. It is important to note that by scaling with the sum over all individually stretched and transformed Euclidean distances the sum over all likelihoods is one. Equation (4) is carried out as vector operations across all elements in the heat map.

2.4. Knowledge Transfer with Quantile Forest Regression

A secondary method of attempting to classify objects is to use their Euclidean distance or likelihood matrix as a whole, rather than selecting the single most likely neuron after building the label distribution across the SOM. When projecting an input image onto a trained SOM lattice, it is being placed into a lower dimension feature space. Rather than attempting to classify features in the image directly, the similarity measure produced by PINK can instead be thought of as a fingerprint from which a classification can be made. This lower dimensional space which is rotation invariant can be used as the basis for more generalized methods.

A Random Forest Classifier (RFC; Breiman 2001) is a supervised ML method which will construct a series of decision trees acting against an input set of training data to describe its corresponding labels. Those decision trees can be seen as a segmentation of the feature space orthogonal to the dimension axis based on a specific information criterion. To improve classification and control over-fitting, the input training data can be subdivided to train a collection of individual decision trees, whose predictions are then collected and averaged. Instead of using the mean prediction, the evaluation of the individual results is helpful to understand the distribution of the individual predictions further than the mean allows (Meinshausen 2006). Therefore we built a quantile regression forest based on the standard `scikit-learn`[12] (Pedregosa et al. 2011) `RandomForestRegressor` class by inspecting the predictions of all individual ensemble members, separately. While training the `RandomForestRegressor` we utilized a stratified k-fold cross validation strategy (Mosteller & Tukey 1968; Geisser 1975). This approach randomly segments the data into k number of sets, and across k repetitions each segmented group is selected once for testing with the the remaining k-1 being used as training data. During each round the training segments of data were crafted to have an equal balance of class labels. For this work we use five folds.

The `RandomForestRegressor` was configured to construct an ensemble of 128 discrete decision trees individually constructed against bootstrapped subsamples generated from the training data set. We empirically selected the number of discrete trees as a compromise between accuracy of the `RandomForestRegressor` while being able to efficiently leverage all available CPU cores through parallelization. Other hyperparameters of the `RandomForestRegressor` were kept as their default values.[13] Results from these individual trees were averaged together to make a prediction. Input features were the 2 dimensional likelihood matrices flattened to a one dimensional vector, which for a SOM lattice of 15×15 neurons, constituted $M = 225$ features.

The quantile regression forest using the individual cells of the heat maps as input features can be used to transfer user generated labels to yet unseen objects. In the context of this work, we use the quantile regression forest to predict the corresponding RGZ class labels of object images that we describe in the following section.

3. Data from Radio Galaxy Zoo

RGZ asks members to classify objects with complex radio morphologies across multiple wavelengths. Upon participation, RGZ users become "citizen scientists", members of the general public who undertake work in collaboration with professional scientists. Utilizing a Web-based interactive front end, the citizen

[12] http://scikit-learn.org/stable/index.html
[13] Using `scikit-learn` version `0.19.1`.

scientists are presented with a collection of multi-wavelength images and asked to classify various components and properties. The idea is to generate a sample of answers from multiple responses to build a consensus of what are the true radio and IR components of a single complex source. Although individual citizen scientist participants may not have domain expertise, the collective answers tend to be consistent with answers provided by experts in the field (Banfield et al. 2015).

Since its initial public launch, the service provides in excess of 170,000 radio source components for the citizen scientists to classify using a collection of publicly available astronomy data sets, which we describe below.

3.1. The Data

For this experiment we obtained an internal pre-release copy of the RGZ Data Release 1 (O. I. Wong et al. 2018, in preparation) catalog, which provides training labels produced by the citizen scientists of 103,930 radio components. We summarize the data they used and their procedure for classification below roughly following Banfield et al. (2015).

The primary sample that makes up the RGZ database is sourced from the Very Large Array (VLA) 1.4 GHz Faint Images of the Radio-Sky at Twenty centimeters (FIRST; Becker et al. 1994). This program covers roughly 10,000 square degrees of sky at a resolution of $5''$ down to a 1σ r.m.s noise level of 150 μJy/beam. In all, FIRST detected approximately 947,000 discrete objects.

Complementing this radio survey, RGZ used the Wide-field Infrared Survey Explorer (*WISE*) all-sky program (Wright et al. 2010). With four wavelength bands corresponding to 3.4, 4.6, 12 and 22 μm (labeled as W1, W2, W3, and W4) reaching 5σ point-source sensitivities of 0.08, 0.11, 1. and 6.0 mJy, respectively, the survey is a powerful tool to study the stellar and interstellar processes in galaxies.

RGZ include in their data set sources from the FIRST survey that satisfy two simple criteria: (1) the source has a signal to noise ratio in excess of 10, and (2) the source appears resolved. This second criterion excludes simple, compact radio sources, leaving complex sources with difficult-to-match morphologies.

Banfield et al. (2015) define a source as being resolved if it satisfies

$$\frac{S_{\rm peak}}{S_{\rm int}} < 1.0 - \left(\frac{0.1}{\log\left(S_{\rm peak}\right)}\right), \tag{5}$$

where $S_{\rm peak}$ is the flux density in units of mJy/beam and $S_{\rm int}$ is the total integrated flux of a source respectively. Their final data set used for training the map consists of about 100,000 sources.

For each object in DR1, we downloaded the corresponding Flexible Image Transport System (FITS; Wells et al. 1981) images from the FIRST[14] and *WISE*[15] postage stamp services.

In total, we had access to images from the FIRST and *WISE* surveys of 103,930 objects with corresponding labels from RGZ DR1. These labels presented in subsequent tables and figures encode the number of radio "components" (N_C) and "peaks" (N_P) as "N_C_N_P". Although the number of components is derived from the RGZ participants, the number of peaks is obtained as a product from the RGZ data processing pipeline. The term "component" refers to discrete individual radio source components identified above a 4σ pixel intensity threshold, and "peak" refers to the number of peaks within the set of components (Banfield et al. 2015). An object that appears to be a point source may be classified as "1_1", as it only has a single component with a single distinguishing peak. An AGN whose jets have a small angular separation could be classified as "1_2", as the single component of contiguous pixels would have multiple peaks. A more complex AGN with a distinction region of separation between its radio lobes may be classified as "2_2", as it exhibits two individual components, each with one peak.

Accompanying each object from RGZ DR1 is a consensus level (CL) which indicates how consistent a classification was. It is defined by Banfield et al. (2015) as $N_{\rm consensus}/N_{\rm all}$, where $N_{\rm consensus}$ is the number of volunteers who agree on the arrangement of the radio components, and $N_{\rm all}$ is the total number of classifications of an object. A CL closer to one indicates a more reliable classification, in the sense that a larger number of participants agreed. For the label transfer experiments using a random forest regressor, objects with a high consensus level exceeding a value of 0.6 have been chosen. A secondary selection was made so that labels matched the same set used by Wu et al. (2019). Applying this initial criteria produced a sample of about 50,000 objects with a highly imbalanced distribution of labels, where the more complex "1_3", "2_3" and "3_3" classes having fewer then 800 objects in each. To better balance the class labels we randomly selected 2,000 objects in each, with repetition allowed. Duplicate object labels were subsequently dropped. The distribution of the consensus levels across the individual classes is shown in Figure 1 while numbers of those 7,464 objects with respect to the provided labels are given in Table 1. This subset of objects are used for the random forest regression experiment. We note that more complex labels generally have a lower radio CL. Naturally, increased complexity and ambiguity among an object image can lead to more disagreement among the participants.

No attempt is made to isolate or filter out images with potentially more than one discrete object in the field. PINK will only learn consistent features. If by chance there are additional sources within the field secondary to the centered subject, these inconsistent features would be treated similar to noise. Throughout training these secondary sources, as they are inconsistent in terms of their proximity to the centered object, would be filtered out in a manner similar to noise.

[14] https://third.ucllnl.org/cgi-bin/firstcutout
[15] https://irsa.ipac.caltech.edu/ibe/docs/wise/allsky/4band_p3am_cdd/

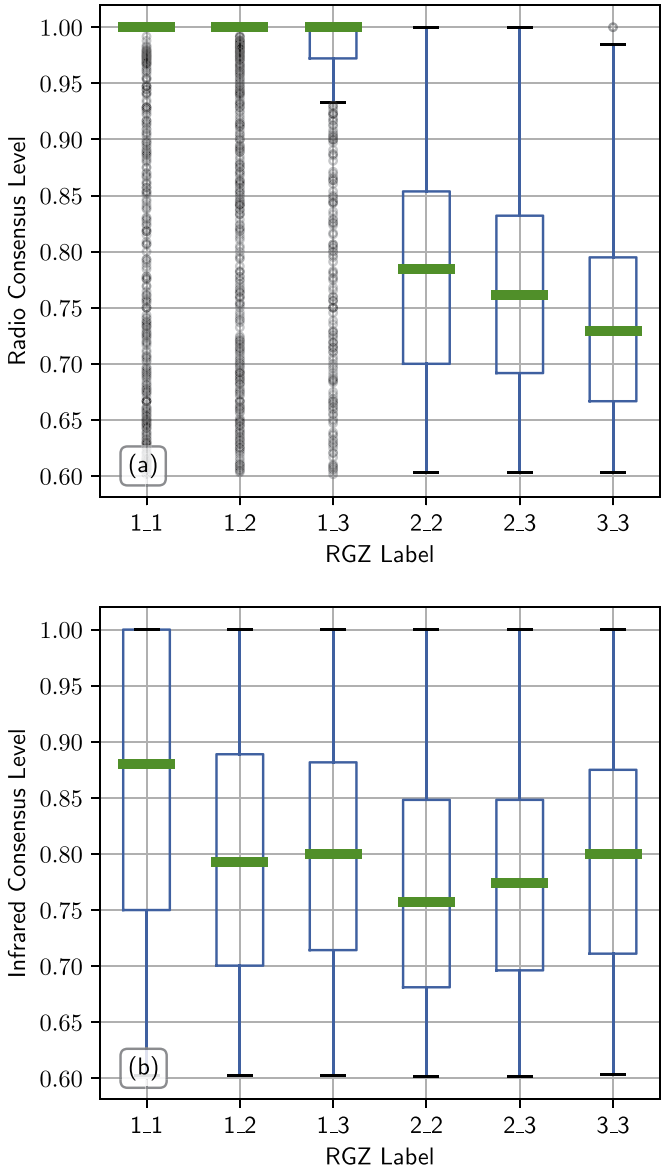

Figure 1. Distribution plots of the radio (a) and infrared (b) consensus levels of objects selected for the training and testing of label distribution across the SOM of six RGZ defined classes. Each box spans from the lower to upper quartile, with the solid horizontal line representing the median value. For classes whose interquartile range is zero as the majority of their CLs were one, only the median horizontal line is shown. Whiskers from the boxes show the interquartile range extended by 150%. Data points outside the whiskers are shown as circles.

(A color version of this figure is available in the online journal.)

3.2. Data Preprocessing and Preparation

The initial preprocessing step applied to all *WISE* W1 band FITS images was to reproject them onto the same pixel grid of the corresponding FIRST radio continuum image using the `astropy` (Astropy Collaboration et al. 2013, 2018) affiliated `python` based `reproject`[16] module. PINK does not use the World Coordinate System (WCS; Greisen & Calabretta 2002) and operates on binary data matrices. Although this reprojection stage is not necessary for PINK to function, the convenience of having aligned features to the human observer across multi-wavelengths when preparing the data and inspecting the learnt features in the trained SOM was worth the small processing overhead, particularly when crafting image cubes whose channels contain images of different wavelength. We applied distinct sets of preprocessing stages separately to the FIRST and *WISE* images, which we implement in the `ImagesToBin.py` script and describe below.

Table 1
An Overview of the Types of Labels and Their Counts Which are Used to Build the Random Forest Classifier

Label	Components	Peaks	Number	Fraction
1_1	1	1	1,947	26.1%
1_2	1	2	1,786	23.9%
1_3	1	3	775	10.4%
2_2	2	2	1,585	21.2%
2_3	2	3	631	8.5%
3_3	3	3	740	9.9%
Total			7,464	100.0%

3.2.1. FIRST

The initial step applied to all FIRST images was searching for pixels whose values were designated as "Not a Number" (`NaN`). These values represent pixels masked out of the imaging process, and were common for objects located near the edge of a FIRST mosaic region. These values were removed by first characterizing the mean and standard deviation of valid pixels around the outer edge region and the image, and then randomly drawing a value assuming a normal distribution with the derived quantities to replace them with. This ensures that the replacement of missing values does not introduce any morphological features.

Noise among the pixels of the FIRST radio continuum images is correlated due to the convolution of the Very Large Array (VLA) point-spread function, even after applying the iterative `clean` algorithm (Högbom 1974) and its more modern derivatives. Applying ML methods to such data that exhibits distinct structure in the background has to be done with care, as components or properties of the noise may be learnt as distinguishing features.

After correcting the background fluxes for bias by subtracting the mean background pixel value from the image, all values below a one standard deviation threshold are considered noise. Therefore all values are shifted so that the one standard

[16] https://reproject.readthedocs.io/en/stable/

deviation threshold is now the new zero point of the image and all negative values are clipped. This is done to prevent the background from being considered in Equation (3) as a spatial characteristic. Afterwards a scaling is applied to place all images onto a consistent intensity scaling, making the data intensity-invariant with respect to the applied distance function.

3.2.2. WISE

Unlike the FIRST data, the noise characteristics of *WISE* images are not correlated among nearby pixels as it is based on an infrared array. The fact that there is no underlying structure in the noise means that PINK throughout its training process should not find features other than consistent source morphologies, aside from calibration issues. No pixel threshold clipping was applied to the *WISE* image data as it could in fact mask out real, faint features.

We replaced missing pixel values by sampling from a normal distribution whose mean and standard deviation were derived from the entire field of the image. Although this included real emission, and is not a true characterization of the image noise properties, the number of pixels replaced was minimal. Image data were then placed on a logarithmic scale, and a min-max normalization was then performed following:

$$I_{normalized} = \frac{I - \min(I)}{\max(I) - \min(I)}, \quad (6)$$

where I is the image data to be normalized.

3.2.3. Image Cubes

We created a two channel image cube using the preprocessed FIRST and *WISE* data compatible with the PINK binary format. When crafting these final image cubes we applied a 95% and 5% weighting factor to the FIRST and *WISE* images, respectively. This was done to encourage PINK to first focus on identifying the broad radio morphologies before isolating the *WISE* features.

We show these steps of the preprocessing in Figure 2 for `FIRST J140118.8+061210`.

4. Experiments

4.1. Training the SOM

This work builds upon the maps presented by Polsterer et al. (2016) by running PINK jointly on the FIRST and *WISE* data sets, creating a single two channel map with radio and IR features. PINK itself has a number of user-defined parameters that influence the algorithm and final convergence.

The SOM was initialized with random noise with a preferred direction. We used bilinear interpolation when generating rotated images. No periodic boundary conditions (i.e., edge wrapping neighborhood function) were used. Training of the SOM for each experiment was carried out across five steps, each targeting and refining different feature scales across the surface of the SOM. The neighborhood function (which includes the learning rate) and desired number of rotations were set using the `-dist-func` and `-numrot` PINK arguments. Specific values used across different training stages are listed in Table 2. A single iteration refers to using each item in the training data set once when constructing the SOM.

The goal of these consecutive stages is to establish the large scale structure and broad layout of source morphologies across the SOM surface, distinguish subsets of object types among the broad regions, and fine tune the neurons and their surrounding features. A fine level of rotational increments at the earliest training stages is not needed, where only the broad structure of the SOM is established. The values presented in Table 2 were empirically selected as a compromise between training time and accuracy. At the earliest training stages 48 interpolated images corresponds to $\sim 8^\circ$ increments. The neuron dimensions are a factor of $\sqrt{2}/2$ smaller then then the original $5' \times 5'$ input images. Assuming the worst case where a feature is on the border of the prototype, 8° of rotation corresponds to a shift of roughly $20''$. With the FIRST pixel size being $\sim 1\farcs8$, this represents a potential misalignment of $11/2 = 6$ pixels between a rotated image and an assumed prototype, in the worst case.

In Figure 3 we show the relative change of a SOM during its training phase across these five learning stages. Intermediary SOMs were output by PINK at 5% intervals throughout training during each stage. We calculate the Euclidean distance (Equation (3)) between the final map, and the maps produced at each intermediary stage there after. During the initial stage, where the broad layout of the SOM is established, there is a high rate of change in the Euclidean distance statistic. There is also a high rate of change over time starting at roughly the 65% point. This is the point where we begin refining the map now that the general shape is well established. Other learning stages exhibit relatively smaller changes. The oscillating behavior towards the end of the training is an indicator that individual steps during a single iteration end in more or less the initial state and therefore the map has converged. To assess this behavior we calculate the total intensity of each image cube and found the median value to be $\sim$150. The relative change of the entire SOM surface during the last training stages was $\sim$10. Hence, the change per neuron is approximately $(10/225)/150 = 0.03\%$ the median total image cube intensity. Although additional training stages could be made by decreasing the learning rate and region of influence of the neighborhood function further, any improvements would be small and unnecessary for our purposes.

Experiments were run across multiple compute nodes, each equipped with four NVIDIA Tesla P40 GPUs, 60 CPU cores and a total of 128 GB main memory.

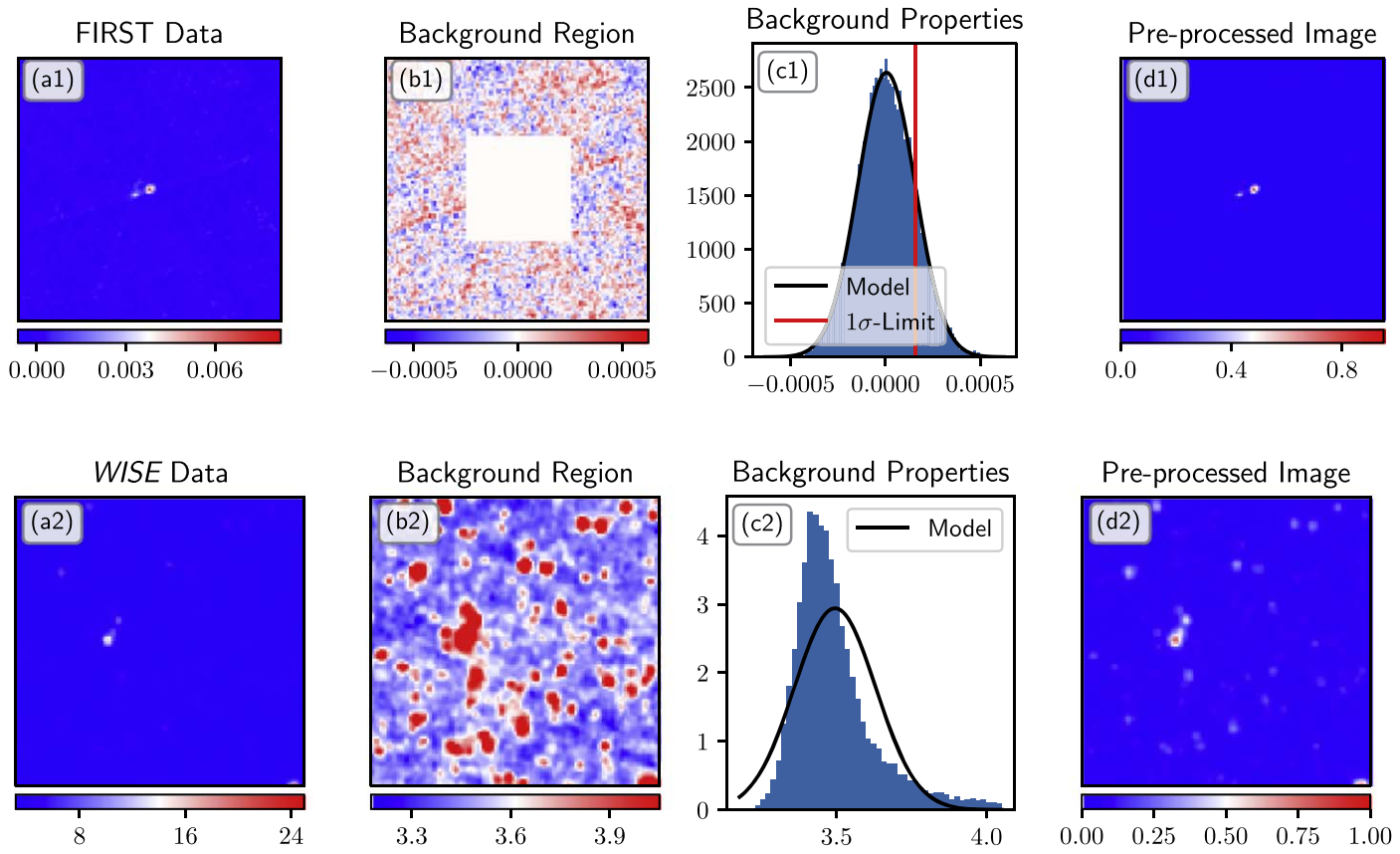

Figure 2. Preprocessing stages applied to the FIRST (top row) and *WISE* W1 band data (bottom row) data sets for `FIRST J140118.8+061210`. Column (a) shows the initial images from the FITS images. In panels (b1) and (b2) we highlight the masking region used to obtain an estimate of the background noise statistics of the FIRST radio continuum image of each source. The overlaid Gaussian in panel (c1) shows the model we construct to replace empty and missing pixel values. The final FIRST preprocessed image, shown in (d1), places the pixels between the red vertical line in panel (c1) to the maximum pixel value onto a zero to one intensity scale. Empty pixels in the *WISE* input image were replaced using the overlaid Gaussian model shown in panel (c2), based on all pixel intensities of the data presented in panel (b2). To better emphasize the noise characteristics we have applied a stretch to hide the brightest pixels of panel (a2). The *WISE* data were then placed onto a logged scale before being normalized onto a zero to one intensity scale, shown as the final preprocessed image in panel (d2). We show the pixel intensity range under each appropriate panels as a accompanying color bar. Pixel values in the original FIRST and *WISE* images are Jy/beam and Digital Numbers (DN) respectively.
(A color version of this figure is available in the online journal.)

Table 2
Parameters Used for Training a 15 × 15 SOM on 100,000 Objects

Training Stage	Sigma	Learning Rate	Rotations	Iterations
1	1.5	0.10	48	2
2	1.0	0.05	92	5
3	0.7	0.05	92	5
4	0.7	0.05	360	5
5	0.3	0.01	360	10

4.2. Sampling the Density of RGZ Labels

An initial experiment performed using the PINK produced SOM was to assess how well the individual neurons do at separating object labels from the labeled RGZ data set. In a qualitative sense, a visual inspection of the outputted SOM lattice across all channels (as we show later in Section 5) does show an evolution in morphologies. We produce a measure of the label distribution across individual neurons using the set of high consensus level RGZ objects. For a properly trained SOM, there should be a representative prototype for each training object image upon its lattice. Hence, distributing object labels to their corresponding prototype should result in a clustering of labels, particularly if the labels themselves are robust.

The likelihood matrix (Equation (4)) of each of the 7,464 high consensus labeled objects was used to distribute labels to neurons. These objects are a subset of those used to train the SOM with. To account for uncertainty in an object's position, 100 realizations (with repetition allowed) of a. object's position were drawn following its probabilities contained in its likelihood matrix, where each realized position would receive

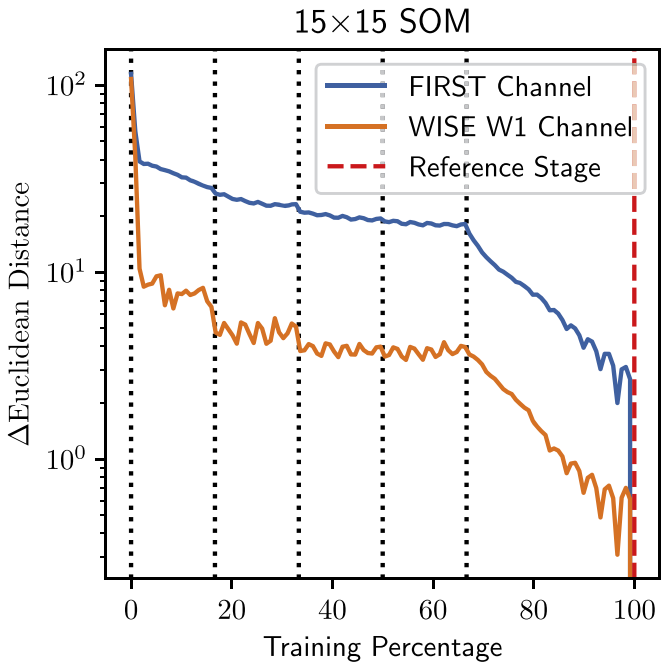

Figure 3. Change of the Euclidean distance of the SOM throughout training relative to its final state. We highlight the two SOM channels independently. The vertical black dotted lines represent a change in the training stage as described in Table 2, and the vertical red dashed line is the selected reference point.
(A color version of this figure is available in the online journal.)

a copy of the object's label. Further, as the relative change of morphologies across adjacent neurons should be gradual, in principle distributing RGZ classifications in this manner should strengthen the clustering of labels and build a more reliable discrete probability function of each prototype's complexity. Prototype neurons with strong clustering of a labeled classification can be considered as reliable examples of that class. More complex prototype neurons should exhibit a spread of labels, particularly for the feature that is ambiguous or where the set of associated object CLs are below one.

4.3. Label Transfer Experiment

We conducted some experiments to check how well the user derived labels can be transferred using the dimensionality reduction method provided by PINK. To analyze the prediction quality, the classification task based on the labels described in Table 1 has been split into independent regression tasks. Instead of predicting the combined number of components and number of peaks, those values are estimated separately. Therefore the knowledge transfer experiments have been executed on both sets of labels. We explicitly treat this as a regression problem in order to characterize the ambiguity of an object within the label itself. Consider an object that is in an intermediary stage between one and two components. Its classification of either discrete label is dependent on the sigma contour level chosen to separate or highlight features within an object's image and, in the case of RGZ, the subjective preference of the citizen scientist. A regressed value may be more appropriate to highlight the fuzzy or uncertain "classification" of an object.

The trained SOM is first used to calculate a heat map for all objects with labels as described in Section 2.3. Those heat maps are considered as input features for the regression task, as they encode the morphological characteristics of the objects through their distances to the learned prototypes. By performing k-fold cross validation, a certain number of objects with labels is omitted when building up the regression model and the predictions are calculated for the omitted objects. This allows for a proper prediction on all 7,464 objects without having them being part of the used training data set . As we used a quantile regression forest, we cannot only evaluate the mean over all ensemble members but can check the proper distribution of predictions, which in addition allows us to compute something equivalent to a consensus level.

5. Results

5.1. Visualization of Neurons and Similarity

Once the PINK software is applied to the training data set it outputs the trained set of weights. Visualizing these weights allowed us to inspect the features in the images that the SOM has learnt to be distinguishing. When trained against image cubes with multiple planes or channels, there will be a corresponding channel in the trained SOM surface.

We show in Figure 4 a visualization of the SOM trained by PINK using the preprocessed FIRST and *WISE* image cubes data described in Section 3.2. This particular experiment trained a grid of 15 × 15 of neurons. Within these 225 neurons, there is a clear change of source morphologies.

Within Figure 4(a), the neurons in the region bound by coordinates (A, 11), (A, 14), (I, 14) and (I, 9) contain unresolved point sources, including those with visible artefacts caused by the uncleaned VLA point-spread function. The remaining neurons consist of extended objects. Objects with a clear, well defined separation between two components are contained to the lower left-hand corner focused around position (M, 2) with an approximate bounding radii of 2 neurons. The upper half of the map is made up of objects which exhibit islands of pixels connecting different sets of peaked pixels. Neurons in the upper left quadrant of the SOM are sources which appear to exhibit brighter peak pixels, more diffuse structure and indication of a separate component than those in the upper right corner. The region bound by (I, 7), (I, 8), (K, 11) and (K, 5) of the map shows neurons with two point-source-like components separate from one another in both the FIRST and *WISE* channels. The lower right corner of the *WISE*

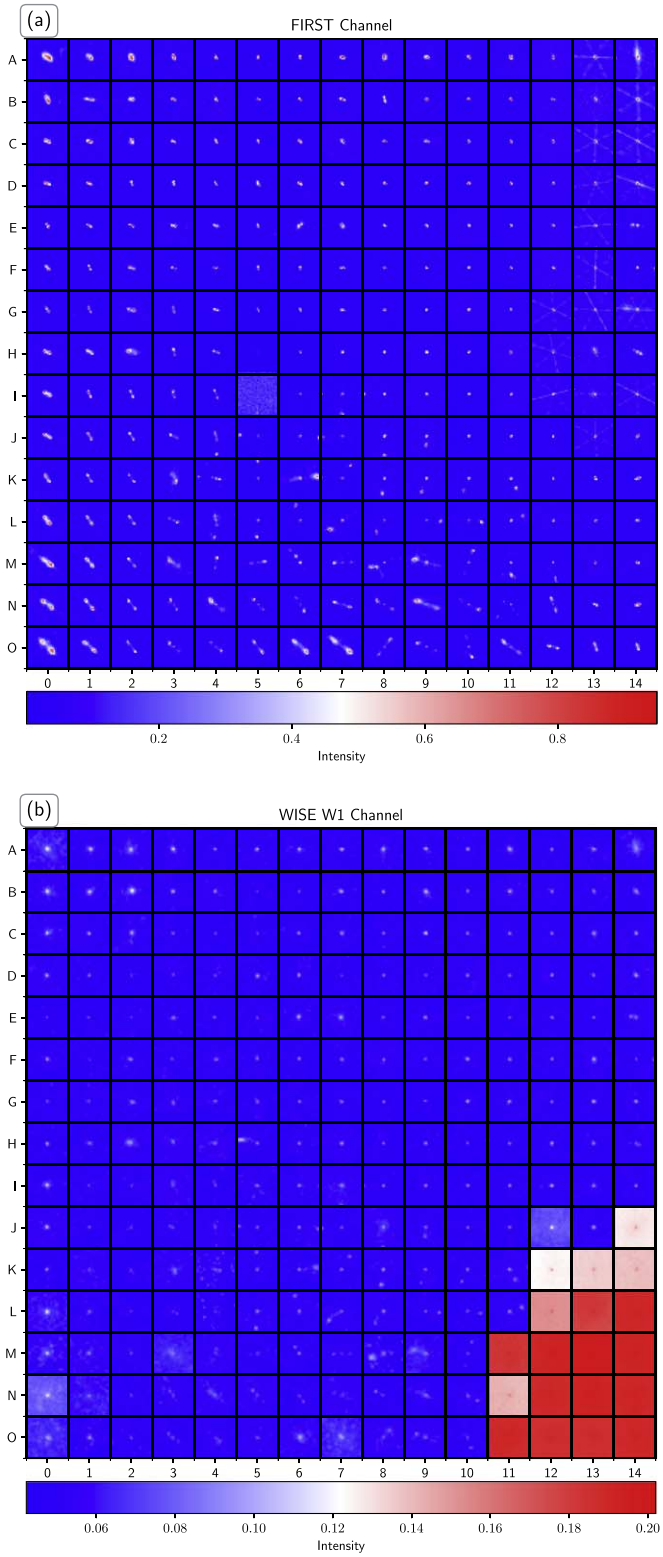

Figure 4. We show the FIRST and *WISE* W1 channels of the SOM as the top *(a)* and bottom *(b)* panels respectively. The solid black horizontal and vertical lines separate adjacent neurons on the SOM surface.
(A color version of this figure is available in the online journal.)

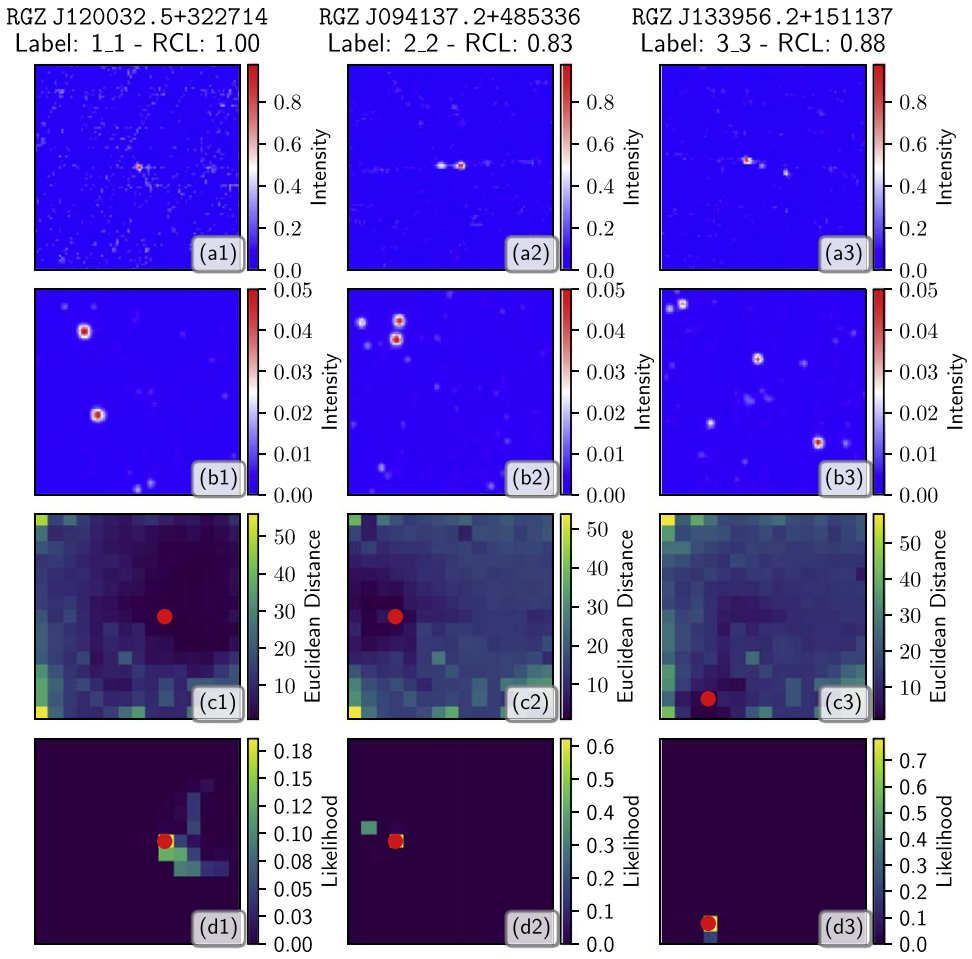

Figure 5. FIRST (row a) and *WISE* (row b) channels of the input cube supplied to PINK of objects `RGZ J120032.5+322714`, `RGZ J094137.2+485336` and `RGZ J133956.2+151137` with preprocessing stages applied. Under the object name of each column we include the RGZ designated label and the radio consensus level, abbreviated as RCL. (row c) The image similarity matrix produced by PINK for rows (a) and (b) using the SOM presented in Figure 4. (row d) The likelihood matrix produced using row (c) and a $\psi = 10$. The red circle in rows (c) and (d) denote the corresponding neuron from Figure 4 PINK judged to be most similar. (A color version of this figure is available in the online journal.)

channel contain mostly image cubes which appear to have a calibration issue in their data.

We present in Figure 5 examples of sources of different RGZ classifications with increasing complexities being mapped to the trained SOM lattice shown in Figure 4. We include the preprocessed FIRST and *WISE* images of `RGZ J120032.5+322714`, `RGZ J094137.2+485336` and `RGZ J133956.2+151137` which were provided to PINK. We also include in accompanying panels the corresponding Euclidean distance similarity matrix produced by PINK and the likelihood matrix constructed using Equation (4) evaluated with a stretching parameter $\psi = 10$. Both measures show distinct regions of activation in the reduced feature space. For the more complex labels, the likelihood matrices have a more compact region of activation, indicating a more conclusive prototype has been identified by PINK. For example, the simple object `RGZ J120032.5+322714`, whose RGZ label is "1_1" and has a radio consensus level of 1.0, has a slightly larger region of activation. This region on the SOM lattice smoothly transforms from simple point-source structures to more complex morphologies. Since these are smooth changes, there is some ambiguity as to the most similar prototype, particularly as the majority of pixels in the input FIRST radio continuum image have been masked out as part of the preprocessing stages.

5.2. RGZ Label Density

We present in Figure 6 the distribution of labels across the corresponding neuron grid for the SOM shown in Figure 4. Labels were distributed across neurons using the likelihood matrix of each source, created with $\psi = 10$, and 100 realizations. As a vertical color bar we highlight the discrete

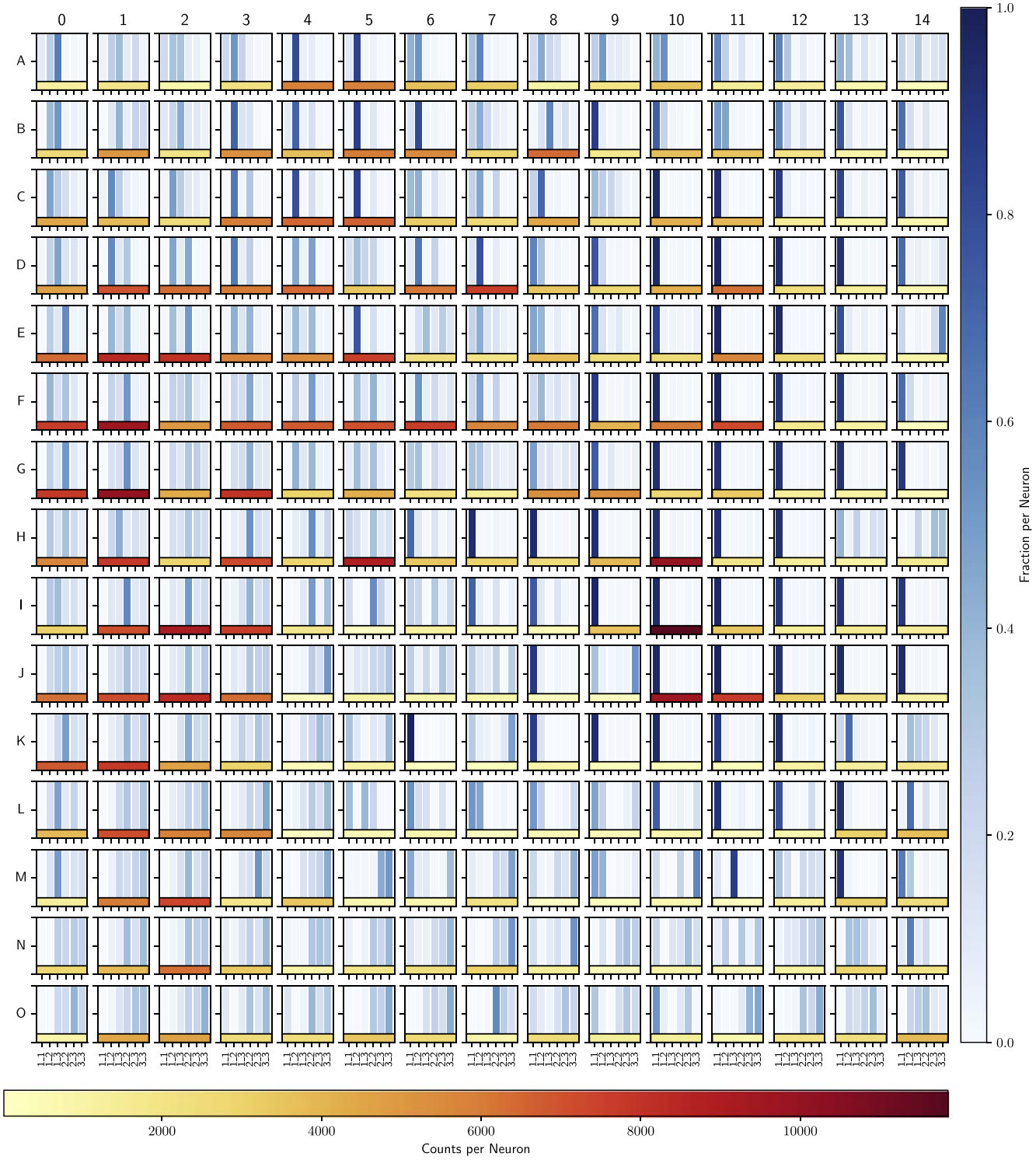

Figure 6. Distribution of labels of the training objects outlined in Table 1 across the SOM lattice presented in Figure 4. Each histogram represents a single trained neuron, and the vertical color bars of each label represents its contribution to the total set of labels of that neuron. Labels were distributed in a Monte-Carlo fashion across 100 realization where their positions were randomly selected using each objects likelihood matrix. The horizontal bar in the lower half of each histogram encodes the number of objects placed in the corresponding neuron across all realizations. A redder color represents a higher number of labels in that neuron. The vertical bars in each histogram correspond to "1_1", "1_2", "1_3", "2_2", "2_3" and "3_3" RGZ labels.

(A color version of this figure is available in the online journal.)

probability distribution function of labels for each segment on the SOM.

Visualizing the density of labels in this manner, it is clear that there is evolution of morphologies among the neurons. The simpler classes with only a single component are clustered well in the upper half of the map, particularly around the (I, 10) region. This is especially true for the "1_1" label which show little activation to the left of column six.

Similarly, there are regions of ambiguity that can easily be distinguished. For instance, neurons surrounding (E, 4) show activation of multiple labels with two peaks ("1_2" and "2_2"). Objects which PINK has deemed to be similar to these prototypes carry with them more difficulty to consistently classify the number of components. The neurons at positions (A, 4) and (E, 4) both appear to strongly preference one of (A, 4)'s ambiguous labels. Similar behavior can also be seen for neurons between (N, 1) to (N, 8), where a variety of labels with two and three components show activation which gradually morphs into the activation of the "3_3" label.

It could therefore be considered that these regions where multiple labels are showing signs of activation are ambiguous prototypes and represent some type of intermediary stage in terms of morphology. The smooth transitions of features learnt by PINK have been able to capture these ambiguous object prototypes.

Included in Figure 6 as a horizontal color bar is an indication of the number of labels associated with each neuron. This approach of visualizing the SOM gives an indication of the relative importance of each prototype morphology, particularly when considering the frequency of the labeled morphologies in the training data set. Point-source-like objects (corresponding to a "1_1" label) are concentrated around the (I, 10) neuron, with counts above 12,000. Neurons whose discrete probability density function indicate more complex morphologies tend to have lower per neuron counts. Although this is somewhat explained by their lower fraction in the labeled data set, an additional component is that PINK has required more neurons to accurately capture the gradual change in shapes among their collective set of morphologies. This thereby spreads the counts of those labels over a larger set of neuron locations.

5.3. Label Transfer

By grouping the likelihood matrices based on the corresponding RGZ label, an average or median likelihood matrix can be constructed for each class. These aggregate matrices highlight the typical set of neurons active across the SOM lattice for each label class. We show in Figure 7 the median likelihood matrix (calculated using $\psi = 10$) for each class of sources when projected onto the SOM lattice shown in Figure 4. Across the six classes, there is a clear separation between the regions of high likelihood and the gradient

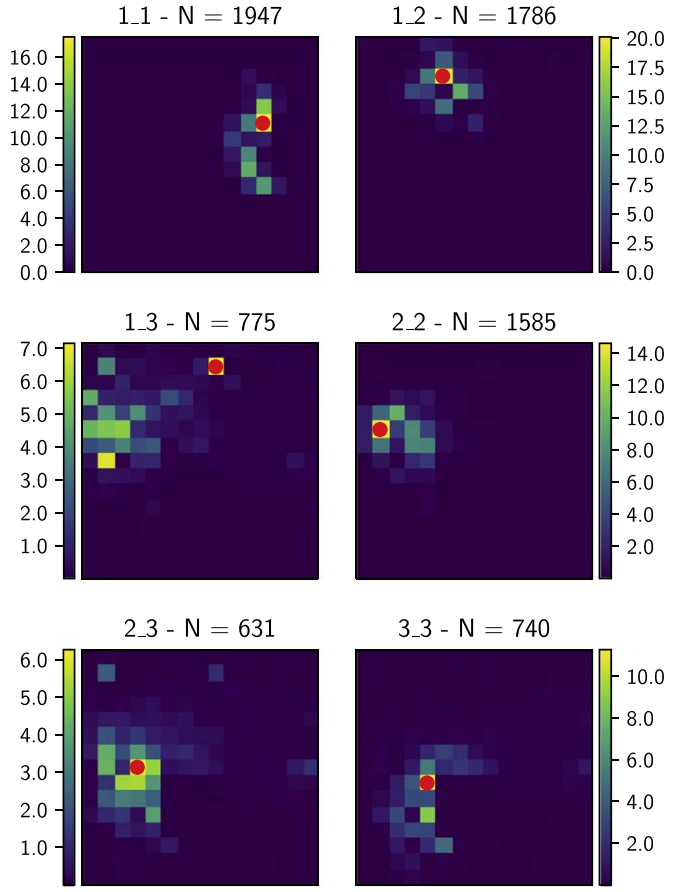

Figure 7. Median likelihood matrix of each class of sources in the high consensus level training set, normalized for their total pixels values sum to one. We include the label and the total number of objects with that label as a title for each panel. The overlaid red circle in each panel represents the neuron with the largest median probability across the surface.
(A color version of this figure is available in the online journal.)

surrounding them. These structures can be used to make a prediction of an object's corresponding label.

As outlined in Section 2.4, we trained quantile random forest classifiers on the likelihood matrices of each labeled RGZ object to predict their classified features. We independently constructed two `RandomForestRegressors` to predict the number of components and peaks for each labeled object, and show in Figure 8 the results of their predictive power grouped by the RGZ designated label. The intersection of the predicted number of components axis and the predicted number of peaks axis can be considered the equivalent RGZ "N_C_N_P" style label. However, as described in Section 2.4, we are taking the mean of the predicted labels across each of the independently trained decision trees, allowing for a "fuzzy" label to be considered. Constructing labels in this manner allows for a mechanism to encode the ambiguity of an object's classification.

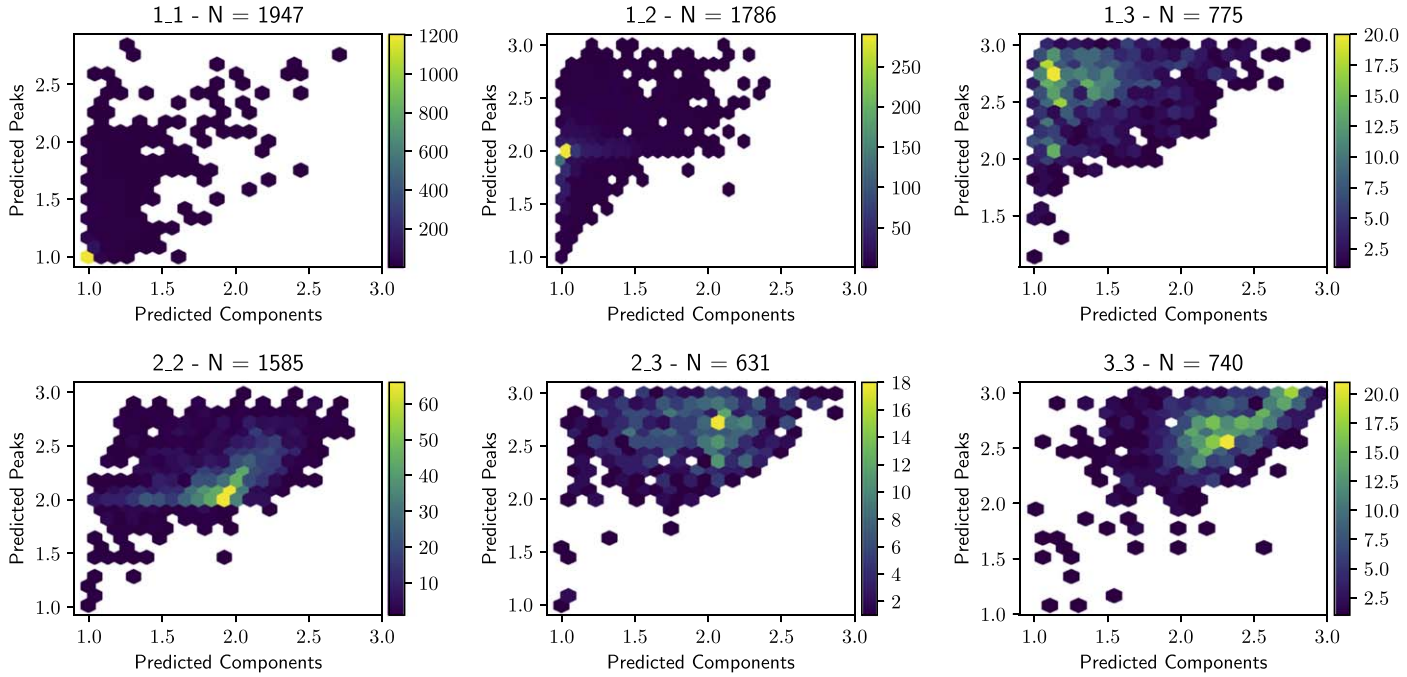

Figure 8. Two-dimensional histograms highlighting the predicted number of components and classes made by the `RandomForestRegressor` for each object grouped by their corresponding RGZ classifications. The intersection of each axis would produce an equivalent RGZ "N_C_N_P" style label.
(A color version of this figure is available in the online journal.)

The predicted regressed values shown in Figure 8 for objects whose RGZ classification specifies a single component each show that the bin with the most counts agree with the corresponding label. For the "1_1" and "1_2" there is a clear separation between the bin with the maximum number of counts and their surrounding cells. Regressed values for objects with a "1_3" RGZ classification are more spread across the surface. The location of the maximum bin (which is located approximately at 2.75) also suggests that there is a fraction of voting trees who favor a lower number of peaks.

Following this, there is also a spread of predicted components and peaks made by the trained `RandomForest-Regressor` for more complex RGZ labels. For objects classified as having two components, there is clear structure among the two-dimensional histograms which broadly agree with the corresponding RGZ label. The spread across the lattice can be seen as natural, as the increased degree of complexity in an object's morphology would lead to a larger disagreement in the cohort of decision trees. This behavior is also exhibited as a lower consensus level for these RGZ objects (Figure 1).

The transition and ambiguity between "2_3" and "3_3" can also be seen in their corresponding histograms in Figure 8. A small but clear set of bins in the "2_3" histogram is beginning to form in the approximate location of "2_2.5", which is where the peak bin is located for "3_3". Between these histograms, however, the region of activation is trending to higher complexity for the RGZ "3_3" label.

6. Discussion

6.1. Distance from Labeled Features

The use of fuzzy labels (e.g. continuous predicted features opposed to discrete labels) allows for an assessment of the distance between the predicted number of components and peaks from the classified number to be made, a comparison which we present in Figure 9. For objects where the entire cohort of decision trees unanimously voted for a single feature which matched the corresponding RGZ label, the distance would be zero. As objects start to exhibit more complex features, ambiguity among the decision trees would trend them away from the zero point. Results of the predictive performance in subsequent discussion of the `RandomForest-Regressor` are based on the results when each object was in the testing segment of the *k*-fold cross validation method.

We find through this comparison that the method of classifying to reduced feature space produced by the similarity measure of PINK is a powerful predictor of an object's label. For this test we produced a likelihood matrix for each source with $\psi = 1$. Both panels of Figure 9 exhibit a strong feature centered on zero far above the residual error across all radio consensus levels. The slight positive bias seen in the

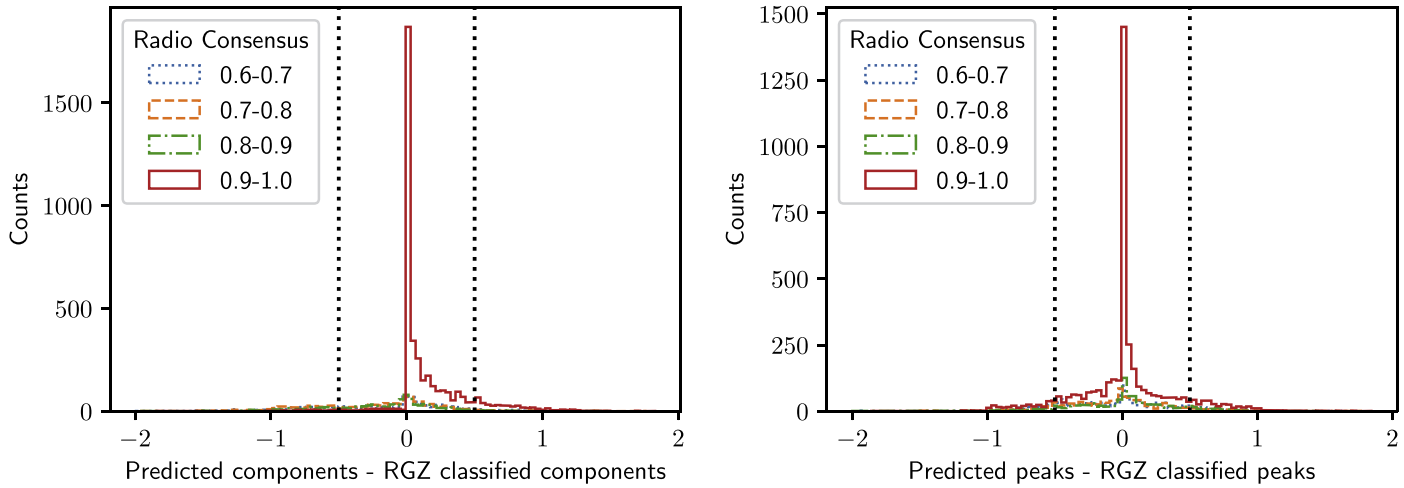

Figure 9. Difference between the number of components and peaks predicted by a `RandomForestRegressor` trained using the likelihood matrix and the number of components or peaks classified by the RGZ citizen scientists. The results have been grouped by the radio CL. The dotted vertical lines represent the regions that would be rounded to zero.
(A color version of this figure is available in the online journal.)

comparison of predicted components to RGZ classified components originates from the large number of RGZ labels included in the training labeled data set and the feature predictor never under-predicting the number of components. Although there appears to be a positive skew in the number of predicted components, this is explained simply by the `RandomForestRegressor` never predicting less than 1 component for objects with "1_1", "1_2" or "1_3" RGZ labels, which are collectively a large fraction of the labeled training data and have high CLs. There are no systematic biases when similar figures are made on a RGZ label basis.

If the soft label scheme we adopt here is converted to a hard label boundary, similar to the RGZ labels themselves, the vertical dotted lines represents the region which would be "rounded" to zero - a correct classification. For the number of predicted components and peaks this encompasses 79.0% and 80.7% of the data set, respectively. For reference, using the fuzzy labels, we compute the sum of the absolute distance from zero for this inner region to be 751.4 and 920.3 for the components and peaks regressors. We note that this is not necessarily an indication that the training method is poor, as the training labels themselves carry uncertainty.

A complete overview of the performance of the predictive component and peak `RandomForest-Regressors` is presented in Table 3, including the predictive performance across RGZ labels and CL subsets.

A powerful mechanism of the use of the `RandomForest-Regressor` is that it is not tied directly to the use of the PINK produced similarity matrices, whether they be in the form of a Euclidean distance or likelihood. They can be supplemented with additional catalog space information, such as redshifts or optical magnitudes. Incorporating this information can help to add additional constraints when making a prediction. As an example, we trained a `RandomForest-Regressor` using both the likelihood matrices and the RGZ radio CLs of each object as a single feature vector of length 226. We show in Figure 10 the difference between predicted and RGZ classified number of components and peaks. There is a clear improvement in the performance of the regressor when compared to Figure 9, particularly for the predicted number of components. When converted to a hard classification scheme, 85.7% and 80.7% of the objects would have a correctly predicted number of components and peaks if the RGZ labels are accepted as ground truth. The sum of the absolute distance for the predicted number of components and peaks is 503.2 and 919.5 respectively. The inclusion of the radio consensus labels has improved the predictive power of the number of components, while the number of peaks has remained similar to training with purely the PINK similarity measures.

We include in Table 4 a complete overview of the predictive performance of the `RandomForest-Regressors` trained using the likelihood matrix supplemented with each object's CL. These results are also broken down across RGZ labels and CL subsets.

The addition of supplementary features is not limited to a single feature. In practice a large collection of data can be added, including sparse or incomplete types of data. For example, spectroscopic classifications from the Sloan Digital Sky Survey (SDSS; York et al. 2000) can be added to objects if they are available without having to exclude others which do not have SDSS identifications.

For both sets of `RandomForestRegressors` trained using outputs of PINK the predictive performance of the "1_3" and "3_3" across most consensus levels was less than 50%.

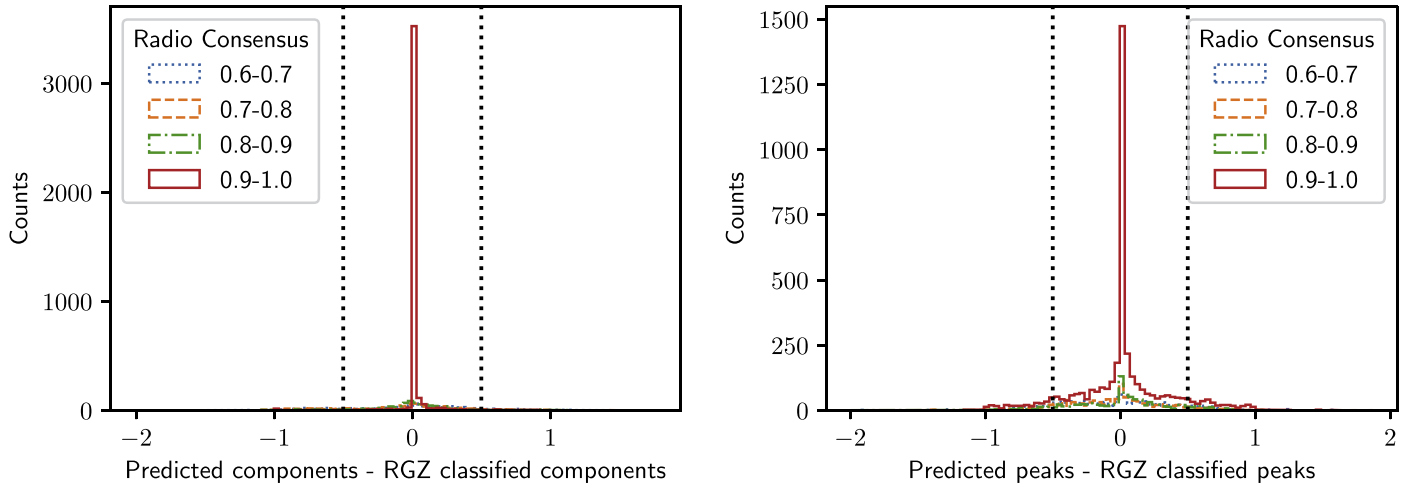

Figure 10. Difference between the number of components and peaks predicted by a `RandomForestRegressor` trained using the likelihood matrix supplemented with the radio CL and the number of components or peaks classified by the RGZ citizen scientists. The results have been grouped by the radio CL. The dotted vertical lines represent the regions that would be rounded to zero.
(A color version of this figure is available in the online journal.)

Table 3
An Overview of the Predictive Performance of the Two `RandomForestRegressors` Predicting the Number of Components (top) and Number of Peaks (bottom) That Have been Trained Using the Likelihood Matrix Produced by PINK

Predicted Component Performance								
RGZ Radio CL	Prediction Result	1_1	1_2	1_3	2_2	2_3	3_3	Summary
0.6–0.7	Incorrect	23/0.90/0.22	17/0.77/0.17	28/0.85/0.19	61/0.69/0.14	32/0.70/0.16	214/0.90/0.31	375/0.84/0.27
	Correct	111/0.10/0.13	84/0.14/0.16	22/0.19/0.12	339/0.19/0.13	146/0.22/0.14	65/0.36/0.10	767/0.19/0.15
0.7–0.8	Incorrect	10/1.15/0.35	10/0.95/0.23	21/0.80/0.26	91/0.67/0.13	51/0.67/0.13	225/0.89/0.27	408/0.82/0.26
	Correct	56/0.06/0.09	51/0.18/0.15	21/0.20/0.15	403/0.17/0.12	170/0.19/0.14	67/0.31/0.11	768/0.18/0.14
0.8–0.9	Incorrect	3/1.11/0.06	12/0.74/0.24	17/0.86/0.31	90/0.67/0.12	40/0.65/0.13	111/0.96/0.29	273/0.81/0.27
	Correct	58/0.05/0.10	66/0.14/0.14	25/0.27/0.15	422/0.17/0.13	135/0.21/0.13	36/0.32/0.11	742/0.18/0.14
0.9–1.0	Incorrect	27/0.74/0.26	201/0.73/0.18	223/0.81/0.27	30/0.69/0.12	19/0.70/0.14	21/0.96/0.31	521/0.77/0.24
	Correct	1659/0.03/0.07	1345/0.12/0.13	418/0.21/0.14	149/0.19/0.13	38/0.19/0.12	1/0.26/0.00	3610/0.09/0.13
Summary		1947/0.06/0.18	1786/0.20/0.26	775/0.44/0.35	1585/0.26/0.23	631/0.31/0.24	740/0.78/0.36	7464/0.27/0.33
Predicted Peak Performance								
RGZ Radio CL	Prediction Result	1_1	1_2	1_3	2_2	2_3	3_3	Summary
0.6–0.7	Incorrect	40/0.98/0.30	16/0.67/0.12	21/0.72/0.16	86/0.65/0.11	55/0.72/0.22	80/0.75/0.29	298/0.74/0.24
	Correct	94/0.12/0.14	85/0.15/0.14	29/0.28/0.13	314/0.21/0.15	123/0.27/0.15	199/0.26/0.14	844/0.22/0.15
0.7–0.8	Incorrect	16/1.06/0.39	11/0.75/0.16	16/0.80/0.28	87/0.68/0.14	57/0.71/0.21	78/0.69/0.22	265/0.72/0.23
	Correct	50/0.08/0.13	50/0.16/0.15	26/0.25/0.11	407/0.17/0.14	164/0.23/0.14	214/0.23/0.15	911/0.19/0.15
0.8–0.9	Incorrect	10/1.00/0.29	12/0.65/0.14	12/0.79/0.20	90/0.64/0.10	50/0.75/0.30	35/0.77/0.28	209/0.71/0.23
	Correct	51/0.08/0.11	66/0.12/0.13	30/0.23/0.14	422/0.17/0.14	125/0.26/0.14	112/0.23/0.15	806/0.18/0.15
0.9–1.0	Incorrect	215/0.83/0.27	203/0.68/0.13	218/0.77/0.22	25/0.67/0.11	22/0.68/0.14	5/0.85/0.29	688/0.76/0.22
	Correct	1471/0.05/0.11	1343/0.14/0.14	423/0.25/0.14	154/0.15/0.14	35/0.26/0.15	17/0.25/0.18	3443/0.12/0.14
Summary		1947/0.17/0.32	1786/0.21/0.23	775/0.43/0.30	1585/0.27/0.23	631/0.39/0.28	740/0.37/0.29	7464/0.27/0.29

Note. The results have been broken down by the RGZ radio CL, the RGZ label and whether the prediction result is correct if a hard classification scheme is used. For each intersection (including the summary column and row) we include the number of objects in that subset, their average absolute distance from zero (a perfect prediction), and the standard deviation of the absolute distance from zero.

Table 4
An Overview of the Predictive Performance of the Two `RandomForestRegressors` Predicting the Number of Components (top) and Number of Peaks (bottom) That Have been Trained Using the Likelihood Matrix Produced by PINK Supplemented by the RGZ Radio CL of Each Object

Predicted Component Performance								
RGZ Radio CL	Prediction Result	1_1	1_2	1_3	2_2	2_3	3_3	Summary
0.6–0.7	Incorrect	31/0.91/0.26	48/0.86/0.22	46/0.97/0.26	20/0.64/0.12	20/0.62/0.12	198/0.80/0.25	363/0.82/0.26
	Correct	103/0.15/0.14	53/0.21/0.16	4/0.38/0.12	380/0.17/0.13	158/0.23/0.14	81/0.34/0.11	779/0.20/0.14
0.7–0.8	Incorrect	14/1.05/0.39	36/0.86/0.23	35/1.00/0.28	22/0.64/0.13	24/0.62/0.10	199/0.78/0.19	330/0.80/0.23
	Correct	52/0.12/0.13	25/0.19/0.14	7/0.27/0.14	472/0.15/0.12	197/0.20/0.14	93/0.33/0.12	846/0.18/0.14
0.8–0.9	Incorrect	9/0.84/0.23	42/0.77/0.19	36/0.98/0.26	11/0.57/0.06	20/0.67/0.12	106/0.83/0.22	224/0.82/0.23
	Correct	52/0.13/0.12	36/0.24/0.16	6/0.36/0.11	501/0.12/0.11	155/0.17/0.12	41/0.32/0.10	791/0.15/0.13
0.9–1.0	Incorrect	2/0.58/0.05	33/0.75/0.14	50/0.79/0.22	38/0.82/0.18	15/0.76/0.19	21/1.16/0.48	159/0.83/0.28
	Correct	1684/0.00/0.02	1513/0.02/0.06	591/0.02/0.07	141/0.16/0.13	42/0.16/0.12	1/0.29/0.00	3972/0.02/0.06
Summary		1947/0.04/0.17	1786/0.10/0.25	775/0.22/0.40	1585/0.18/0.18	631/0.25/0.20	740/0.67/0.31	7464/0.18/0.30
Predicted Peak Performance								
RGZ Radio CL	Prediction Result	1_1	1_2	1_3	2_2	2_3	3_3	Summary
0.6–0.7	Incorrect	43/0.96/0.29	16/0.65/0.12	22/0.71/0.15	93/0.64/0.11	54/0.71/0.22	83/0.73/0.28	311/0.73/0.24
	Correct	91/0.12/0.14	85/0.16/0.14	28/0.27/0.14	307/0.21/0.14	124/0.27/0.14	196/0.26/0.13	831/0.22/0.15
0.7–0.8	Incorrect	15/1.11/0.36	11/0.75/0.17	15/0.81/0.27	84/0.68/0.13	51/0.72/0.21	71/0.71/0.23	247/0.74/0.23
	Correct	51/0.10/0.14	50/0.17/0.15	27/0.25/0.12	410/0.18/0.14	170/0.23/0.14	221/0.24/0.15	929/0.20/0.15
0.8–0.9	Incorrect	10/0.99/0.32	10/0.67/0.12	13/0.75/0.20	88/0.64/0.11	55/0.72/0.28	37/0.76/0.28	213/0.71/0.23
	Correct	51/0.09/0.11	68/0.13/0.14	29/0.24/0.14	424/0.17/0.14	120/0.25/0.13	110/0.22/0.15	802/0.18/0.15
0.9–1.0	Incorrect	209/0.83/0.27	200/0.68/0.13	231/0.76/0.23	26/0.65/0.12	24/0.67/0.14	6/0.79/0.26	696/0.75/0.22
	Correct	1477/0.05/0.11	1346/0.13/0.14	410/0.25/0.13	153/0.15/0.13	33/0.25/0.14	16/0.22/0.17	3435/0.11/0.14
Summary		1947/0.17/0.32	1786/0.21/0.23	775/0.43/0.30	1585/0.27/0.23	631/0.38/0.27	740/0.37/0.28	7464/0.27/0.29

Note. The results have been broken down by the RGZ Radio CL, the RGZ label and whether the prediction result is correct if a hard classification scheme is used. For each intersection (including the summary column and row) we include the number of objects in that subset, their average absolute distance from zero (a perfect prediction), and the standard deviation of the absolute distance from zero.

These classes also happened to be the two of the smaller groups in our labeled data set . As an initial test we restricted the size of each class to 631 objects (the number of objects in the smallest group) and performed the same tests. We found that this balanced data set performed in a consistent manner to the previous `RandomForestRegressors` without any meaningful difference in its accuracy. There are potentially two compounding effects which are contributing to the relatively poor performance of these classes. objects with these labels are beginning to exhibit complex features that are more dependent on the subjective classification nature of the participating citizen scientists. Defining the number of components based on an image sigma threshold mechanism, particularly for objects whose brightness is close to image sensitivity limits, are particularly susceptible to inconsistent responses. This type of behavior is captured by the radio CL. We also suspect that this issue is being aggravated by there not being enough space on the SOM lattice for these distinct complex features to distinguish themselves. Examining Figure 7 there is overlap in the activation regions of these complex classes. Experimenting with larger SOM sizes maintained the relative proportions of these regions. A more likely approach to improve performance of these classes is to produce a type of hierarchical SOM where data is segmented based on an initial layer, and additional SOMs are trained on these subsets separately in a manner similar to a growing hierarchical SOM (Dittenbach et al. 2000).

6.2. Improvement from PINK

A point to consider is whether the PINK method itself is an effective resource that helps the `RandomForest-Regressor` predict object labels. Given that preprocessing stages were employed to better emphasize important features, there is potential for the `RandomForestRegressor` to perform equally well at distinguishing important shapes and classifying morphologies.

To explore this, we provided the 7,464 preprocessed image cubes used by PINK to the `RandomForest-Regressor` class. These image cubes were flattened to a $2 \times 167 \times 167 = 55{,}778$ feature vector. We utilized the same experimental setup used in the previous section to produce Figures 8 and 9, and performed the same number of cross validation tests.

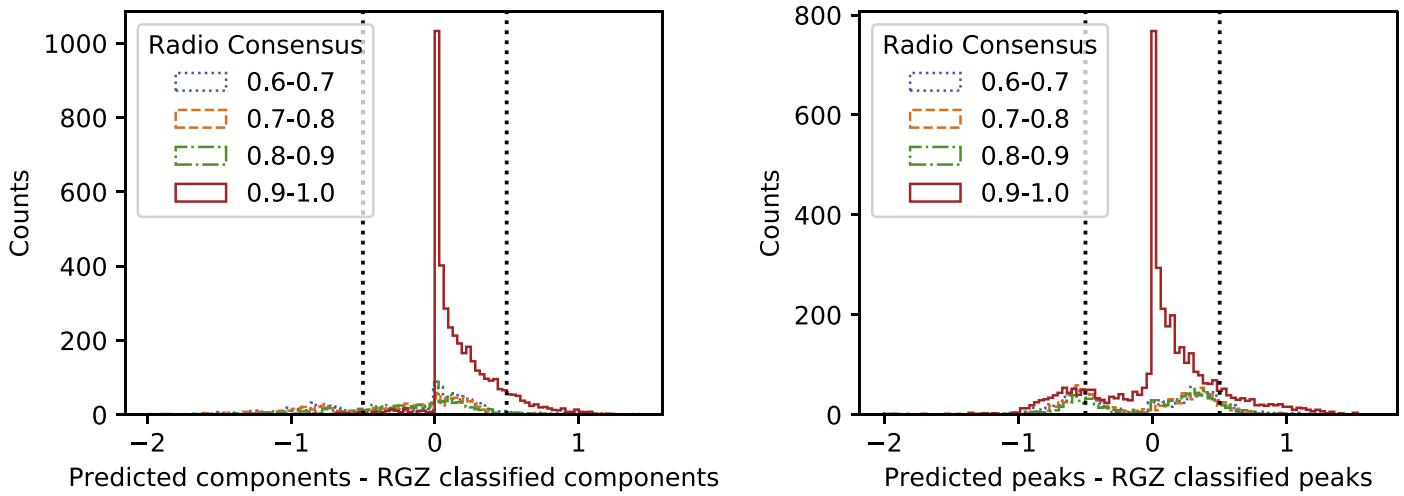

Figure 11. Difference between the predicted number of components and peaks against the number classified by the RGZ citizen scientists, where the results have been grouped by the radio CL. The `RandomForestRegressors` has been trained using the preprocessed images provided to PINK. The dotted vertical lines represent the regions that would be rounded to zero.
(A color version of this figure is available in the online journal.)

In Figure 11 we show the difference between the predicted number of components and peaks and the corresponding RGZ label. Inspecting the difference between the predicted and labeled number of peaks, there is a clear positive bias. Although this was also seen in Figure 9, the effect seen is much more pronounced and extends to lower ranked consensus levels. Unlike Figure 9 however, examining the difference between the predicted and labeled number of peaks shows strong features. Not only is there a strong peak set near the zero made up of the high consensus level objects with a positive skew, there are additional features approximately centered at −0.5 and 0.5.

The presences of the features in Figure 11 indicates that the `RandomForestRegressor` is not capable of handling the affine transforms applied to resolved objects with complex morphologies. Performing similar experiments with different `RandomForestRegressor` options set produced consistent behavior. We can therefore consider PINK as a useful tool to assist with not only reducing the feature space by a factor of ∼248, but also with producing more reliable classification or knowledge transfer tools by factoring out affine transforms of similar morphologies.

An important point to emphasize is that if the fuzzy labels of Figure 11 were rounded to their nearest label, then 78% and 68% of the data set would be centered at zero. At a glance, these are comparable to the results of applying the `RandomForestRegressor` to the PINK produced likelihood matrices, and in fact the total fraction of objects with the correct number of components is higher than the 74% seen in Figure 9. However, when the fuzzy labels are used to calculate the sum of the absolute distance of each object, the residual distance for the predicted components and peaks are 915.71 and 1059.06, respectively. Both of these are at least ∼15% higher than the distances measured using the PINK based regressor method, with the number of components being 30% higher. The use of a hard classification scheme concealed the residual error of the object's predicted labels. It may potentially be more useful to consider a soft classification scheme to better capture anomalous behavior in labels and their distribution.

We include in Table 5 an overview of component and peak `RanddomForestRegressor` predictors, broken into RGZ label and radio CL subsets. Although there are some classes that perform similarly to models trained on PINK produced products when a hard classification scheme is adopted (Tables 3 and 4), we highlight that the average absolute distance is higher for almost all items. This measure indicates that there was a large amount of disagreement among the ensemble of trees forming the random forest.

6.3. Learnt Multi-wavelength Features

Identifying and associating resolved discrete components of a single intrinsic object is a difficult problem, and one that is further complicated when attempting to incorporate multi-wavelength data.

In the radio domain, Line et al. (2017) attempt to use spectral index information to associate related radio components between different catalogs. Across a small radio frequency range, the spectral energy distribution of radio sources is characterized well by a power law, which is incorporated as an additional metric when assessing whether multiple discrete components at a particular frequency are related to a single component at another frequency.

Expanding this cross-cataloging to a wider wavelength space, further complicates the problem. Different physical processes often mean that components of complex sources are separated

Table 5
An Overview of the Predictive Performance of the Two `RandomForestRegressors` Predicting the Number of Components (top) and Number of Peaks (bottom) That Have been Trained Using the Preprocessed Images that were Provided to PINK

Predicted Component Performance								
RGZ Radio CL	Prediction Result	1_1	1_2	1_3	2_2	2_3	3_3	Summary
0.6–0.7	Incorrect	28/0.84/0.26	16/0.73/0.19	13/0.83/0.22	46/0.67/0.13	35/0.71/0.14	279/1.00/0.29	417/0.91/0.29
	Correct	106/0.12/0.14	85/0.16/0.14	37/0.23/0.14	354/0.19/0.12	143/0.20/0.12	None	725/0.18/0.13
0.7–0.8	Incorrect	15/0.78/0.23	13/0.79/0.23	13/0.84/0.17	57/0.65/0.14	32/0.71/0.14	287/1.02/0.28	417/0.92/0.29
	Correct	51/0.11/0.13	48/0.18/0.14	29/0.20/0.14	437/0.19/0.12	189/0.19/0.12	5/0.23/0.12	759/0.18/0.12
0.8–0.9	Incorrect	7/0.72/0.16	10/0.65/0.20	16/0.72/0.17	53/0.63/0.11	28/0.73/0.13	143/1.07/0.29	257/0.89/0.31
	Correct	54/0.10/0.13	68/0.16/0.15	26/0.22/0.15	459/0.19/0.13	147/0.19/0.13	4/0.32/0.13	758/0.18/0.13
0.9–1.0	Incorrect	74/0.66/0.19	211/0.69/0.18	183/0.73/0.17	24/0.62/0.10	9/0.64/0.12	22/1.12/0.31	523/0.71/0.20
	Correct	1612/0.10/0.12	1335/0.16/0.14	458/0.21/0.14	155/0.19/0.12	48/0.21/0.13	None	3608/0.14/0.14
Summary		1947/0.14/0.20	1786/0.23/0.24	775/0.37/0.29	1585/0.24/0.19	631/0.28/0.23	740/1.01/0.30	7464/0.31/0.34
Predicted Peak Performance								
RGZ Radio CL	Prediction Result	1_1	1_2	1_3	2_2	2_3	3_3	Summary
0.6–0.7	Incorrect	69/0.88/0.25	8/0.64/0.08	35/0.77/0.16	74/0.60/0.10	110/0.66/0.16	204/0.71/0.27	500/0.71/0.23
	Correct	65/0.10/0.14	93/0.16/0.13	15/0.33/0.12	326/0.32/0.12	68/0.37/0.10	75/0.41/0.07	642/0.29/0.15
0.7–0.8	Incorrect	27/0.85/0.23	4/0.62/0.13	23/0.76/0.31	82/0.59/0.10	126/0.64/0.15	200/0.66/0.20	462/0.66/0.19
	Correct	39/0.10/0.13	57/0.17/0.12	19/0.37/0.12	412/0.30/0.12	95/0.36/0.10	92/0.40/0.10	714/0.30/0.14
0.8–0.9	Incorrect	20/0.84/0.23	3/0.76/0.08	23/0.78/0.29	77/0.60/0.07	98/0.66/0.24	106/0.68/0.24	327/0.68/0.22
	Correct	41/0.11/0.13	75/0.15/0.13	19/0.35/0.12	435/0.30/0.11	77/0.37/0.09	41/0.39/0.11	688/0.29/0.14
0.9–1.0	Incorrect	386/0.87/0.24	146/0.64/0.12	441/0.74/0.22	20/0.57/0.09	33/0.65/0.11	15/0.66/0.19	1041/0.77/0.23
	Correct	1300/0.09/0.12	1400/0.17/0.13	200/0.34/0.11	159/0.30/0.12	24/0.37/0.10	7/0.41/0.05	3090/0.16/0.15
Summary		1947/0.29/0.38	1786/0.21/0.19	775/0.61/0.27	1585/0.35/0.16	631/0.53/0.21	740/0.60/0.24	7464/0.37/0.30

Note. The results have been broken down by the RGZ Radio CL, the RGZ label and whether the prediction result is correct if a hard classification scheme is used. For each intersection (including the summary column and row) we include the number of objects in that subset, their average absolute distance from zero (a perfect prediction), and the standard deviation of the absolute distance from zero.

and appear to be distinct. For instance, radio lobes of AGN can be separated by some distance from the host galaxy, which is often an unresolved point source at infrared wavelengths.

Human pattern recognition, being well suited for the problem, has already been employed as a solution. However, even harnessing the collective power of citizen scientists through RGZ and related projects, it is unlikely to scale to the millions of objects to be discovered by EMU and the Square Kilometre Array (SKA) era of radio astronomy.

Alternative codes show promise at this specific multi-wavelength cross-cataloging problem. The Likelihood Ratio in PYthon (LRPY; Weston et al. 2018) package implements a modified version of the likelihood method (Richter 1975) to attempt to solve the problem for single radio sources, but cannot yet tackle multi-component radio sources.

Fan et al. (2015) approach the problem of multi-component sources within a Bayesian framework. Constructing a model of AGN and their morphologies, they are able to cross-catalog components of radio objects, such as cores or lobes, to IR components which are not necessarily co-located in the same region of sky. This method incorporates a "prior distribution" when assessing the potential match. The prior distribution incorporates known information about a particular model or morphology.

Using a PINK trained SOM based on multi-wavelength image cubes of objects, it should be possible to craft additional prior distribution information. To demonstrate this, we extracted six neurons from the SOM shown in Figure 4, where each neuron corresponding to a peak pixel of the median likelihood matrices presented in Figure 7. These six neurons are presented in Figure 12. We emphasize that the features in each panel are constructed in an unsupervised manner by PINK, and can be considered ideal prototypes of a set of objects in the input data set. PINK has identified not only a range of radio features, including AGN produced radio lobes, but also where the corresponding IR host is located in the *WISE* channel. These IR components were constructed from images which had no sigma noise clipping or masking of features unrelated to the centered component.

There are also a set of neurons which have learnt to recognize objects which could have a unrelated, nearby galaxies within the field. These neurons, which we include as Figure 13, show two

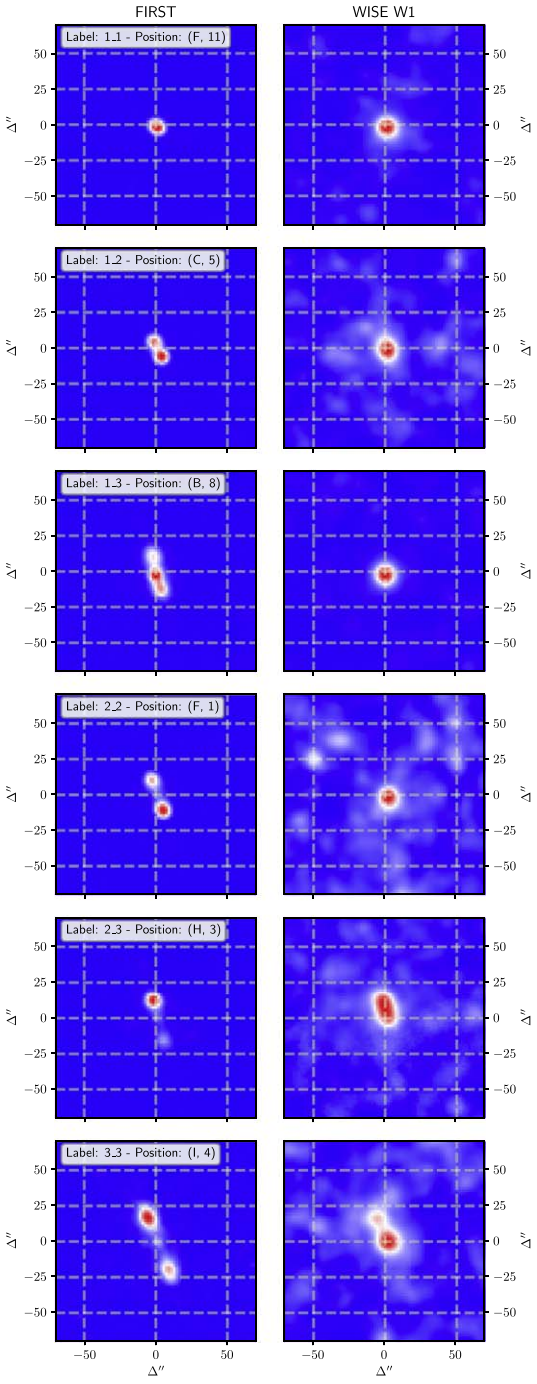

Figure 12. Example of trained neurons produced by PINK showing the learned features for the position of each of the red markers in Figure 7. The left-hand column shows the radio data and the right-hand column shows the infrared data. In these examples, the infrared sources lies between two radio lobes. The overlaid annotation in the upper region of each panel in the left column indicates the corresponding label and the position of the neuron in Figure 4. Each panel has had a square root transform applied to better emphasis their learnt features.
(A color version of this figure is available in the online journal.)

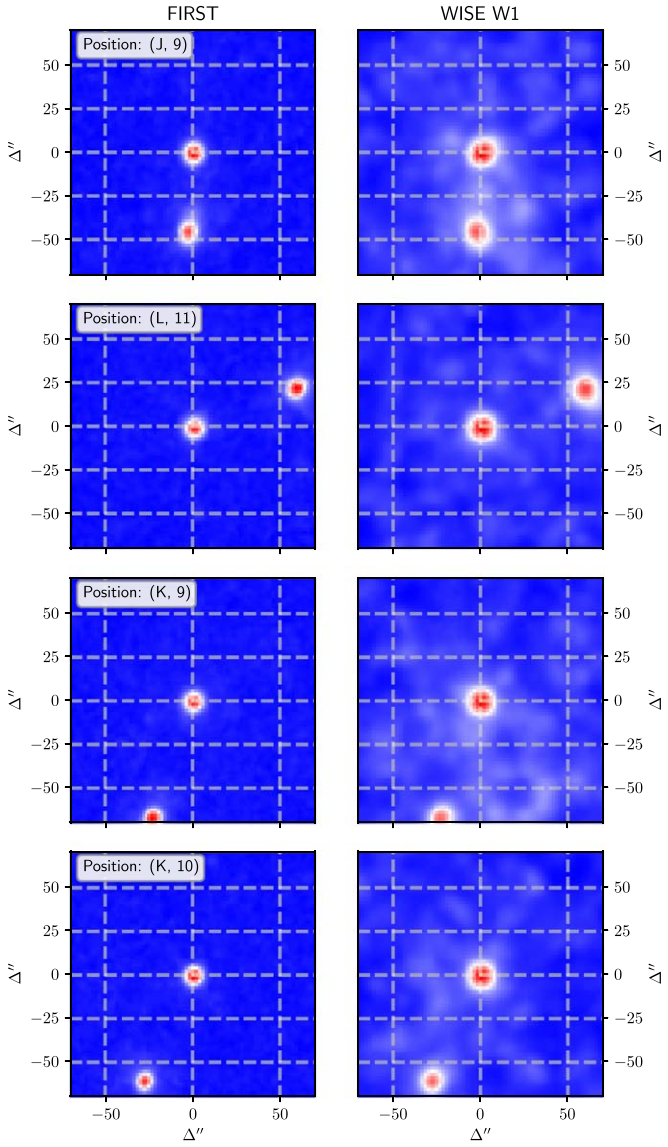

Figure 13. Examples of selected neurons from Figure 4 with what could be classified as independent galaxies. The left-hand column shows the radio data and the right-hand column shows the infrared data. In these examples, an unrelated source in the field is detected in both the radio and infrared images.
(A color version of this figure is available in the online journal.)

separate components in both the FIRST and *WISE* channels. This could suggest that these objects are intrinsically unrelated to one another, as they both have a clear IR component but just happen to be within a close proximity to one another by chance.

We propose that if the training sample of objects and their morphologies are an accurate representative samples of data to be cross-cataloged, then the trained neurons themselves can be used as archetype morphologies. Classifying the features across the different multi-wavelength channels of each neuron would be fairly

trivial when compared to the task of classifying the raw input source images directly, even for a large SOM lattice. The weighting updates throughout the training stage of PINK minimizes noise. With automated source finders, this could be performed with little intervention from an expert. Alongside the complete set of likelihood matrices of objects and the extracted set of features across all neurons, a robust set of well characterized prior functions can be incorporated into Bayesian based cross-cataloging methods. The smooth nature of the SOMs would allow for interpolation to be implemented in constructed prior functions—an important characteristic for distance separating candidate features.

Being an unsupervised method, these priors can be updated with little human effort for different combinations of surveys. Surveys of any wavelength have their own set of sensitivities, limitations and biases. Training any ML method on one data set and applying its learnt knowledge onto another can easily introduce potential issues, reducing its predictive power and ultimate use. This could be particularly troublesome for radio continuum data, where the telescope array configuration and observing frequency not only influences the type of angular scales and physical processes the instrument is sensitivity to, but also affects the point-spread function that has to be modeled and removed from the imaging process. Understanding these effects would be critical to ensure the transfer of knowledge between different data sets using previously trained ML methods.

Developing and extracting useful multi-wavelength features in an unsupervised manner to craft prior distributions for cross-cataloging problems offers an ideal characteristic, where the only cost to retrain is computation time.

An alternate approach to this cross-cataloging problem is to exploit the position of the IR host in each of the neurons. Assume a SOM has been trained based on image cubes (with radio and IR information) for positions centered on islands of pixels found by a source finder on a radio image. Once trained, neurons will show locations of radio components and the IR host. Knowing the radio position of an object and the offset of the IR host in the BMU neuron with respect to this position, then the corresponding sky position of a IR host for each radio component can be estimated. An internal match looking for common estimated IR host locations could be used to identify related object, linking together their radio and IR host components. This method is being investigated by Hopkins et al. (in prep.).

7. Conclusion and Future Work

We have used the PINK software to produce a multi-channel SOM using 100,000 source images based on the RGZ DR1. We find that

1. PINK is able to produce a SOM that exhibits a range of morphologies that are representative prototypes of the training objects used,
2. Labels produced by RGZ participants are able to be clustered on the surface of the constructed SOM,
3. Similarity measures produce by PINK can be used as a basis of object classification and can improve knowledge transfer,
4. A `RandomForestRegressor` was used as a mechanism to efficiently derive and apply a soft classification scheme based on labeled training objects,
5. Adding additional features alongside the PINK produced similarity measures improved the predictive power of the `RandomForestRegressor`,
6. A soft classification scheme that avoids discrete intervals or boundaries should be considered in the near future as it can reveal hidden biases or inconsistencies,
7. The similarity measure produced by PINK assisted in improving the classification performance of labeled training data when the absolute distance was considered as a metric, and
8. Physically meaningful features across multiple wavelengths can be constructed by PINK and potentially used to craft prior distribution functions for Bayesian based methods.

At the time of writing, PINK does not make available to the user the transform matrix that is used to place an input image onto the SOM lattice, although this feature is now planned to be included in a future release. RGZ maintains information of the position of all user clicks as the citizen scientists are classifying features within an object's image. An interesting application of these transforms would be applying the transform information derived by PINK onto user click information maintained by RGZ to project them onto a set of trained prototypes. Each prototype should then show robustly characterized regions highlighting important features within each neuron in a manner similar to a kernel density estimator. Although an individual object image may only have been inspected tens of times by RGZ scientists through the Zooniverse, when transformed and placed onto a trained SOM surface there is potentially information on thousands of user clicks, highlighting consistent features within each neuron. We plan to investigate this once our feature request has been added to a future version of PINK.

We are also investigating whether hierarchically structured SOMs, similar to Dittenbach et al. (2000), can be used to segment data sets into different classes in a unsupervised (or semi-supervised) manner which can then be used as the basis of more reliable classifications scheme. Our `RandomForestRegressor` performed poorly on objects with more complex RGZ labels. We suspect that this is a combination of effects, including inconsistent responses among the citizen scientists (which is characterized by the CL) and complex morphologies being to compressed on the SOM surface. A hierarchical SOM may produce a better set of features for other ML methods, including random forest classifiers, to operate on.

We thank the anonymous referees whose thoughtful feedback improved the presentation and quality of this manuscript.

K.P. and E.H. gratefully acknowledge the support of the Klaus Tschira Foundation.

Partial support for L.R. comes from U.S. National Science Foundation grant AST17-14205 to the University of Minnesota.

This publication has been made possible by the participation of more than 250,000 volunteers in the Galaxy Zoo Project. The data in this paper are the result of the efforts of the Radio Galaxy Zoo volunteers, without whom none of this work would be possible. Their efforts are individually acknowledged at http://rgzauthors.galaxyzoo.org.

The authors thank Chen Wu for providing a publicly accessible repository for code and initial training data and for assistance throughout the project.

This publication makes use of data products from the *Wide-field Infrared Survey Explorer* and the *Spitzer Space Telescope*. The *Wide-field Infrared Survey Explorer* is a joint project of the University of California, Los Angeles, and the Jet Propulsion Laboratory/California Institute of Technology, funded by the National Aeronautics and Space Administration. SWIRE is supported by NASA through the SIRTF Legacy Program under contract 1407 with the Jet Propulsion Laboratory. This publication makes use of radio data from the Australia Telescope Compact Array and the Karl G. Jansky Very Large Array (operated by NRAO). The Australia Telescope Compact Array is part of the Australia Telescope, which is funded by the Commonwealth of Australia for operation as a National Facility managed by CSIRO. The National Radio Astronomy Observatory is a facility of the National Science Foundation operated under cooperative agreement by Associated Universities, Inc.

Code used throughout the data preparation, PINK operation stages and production of Figures are hosted at https://github.com/tjgalvin/Pink_Experiments.

The `astropy` (Astropy Collaboration et al. 2013, 2018) affiliated `python` based `reproject`[17] module was used for preprocessing of the data.

References

Alger, M. J., Banfield, J. K., Ong, C. S., et al. 2018, MNRAS, 478, 5547
Aniyan, A. K., & Thorat, K. 2017, ApJS, 230, 20
Astropy Collaboration, Robitaille, T. P., Tollerud, E. J., et al. 2013, A&A, 558, A33
Astropy Collaboration, Price-Whelan, A. M., Sipőcz, B. M., et al. 2018, AJ, 156, 123
Banfield, J. K., Wong, O. I., Willett, K. W., et al. 2015, MNRAS, 453, 2326
Becker, R. H., White, R. L., & Helfand, D. J. 1994, in ASP Conf. Ser. 61, Astronomical Data Analysis Software and Systems III, ed. D. R. Crabtree, R. J. Hanisch, & J. Barnes (San Francisco, CA: ASP), 165
Breiman, L. 2001, Mach. Learn., 45, 5
Brett, D. R., West, R. G., & Wheatley, P. J. 2004, MNRAS, 353, 369
Crawford, E., Norris, R. P., & Polsterer, K. 2016, arXiv:1611.02829
Dittenbach, M., Merkl, D., & Rauber, A. 2000, Neural Networks, 2000. IJCNN 2000, Proceedings of the IEEE-INNS-ENNS International Joint Conference on, Vol. 6 (Piscataway, NJ: IEEE), 15
Fan, D., Budavári, T., Norris, R. P., & Hopkins, A. M. 2015, MNRAS, 451, 1299
Fanaroff, B. L., & Riley, J. M. 1974, MNRAS, 167, 31P
Geach, J. E. 2012, MNRAS, 419, 2633
Geisser, S. 1975, J. Am. Stat. Assoc., 70, 320
Gianniotis, N., Kügler, S. D., Tiňo, P., & Polsterer, K. L. 2016, arXiv:1601.05654
Greisen, E. W., & Calabretta, M. R. 2002, A&A, 395, 1061
Högbom, J. A. 1974, A&AS, 15, 417
Kohonen, T. 1982, Biol. Cybern., 43, 59
Kohonen, T. 1996, Biol. Cybern., 75, 281
Line, J. L. B., Webster, R. L., Pindor, B., Mitchell, D. A., & Trott, C. M. 2017, PASA, 34, e003
Lintott, C. J., Schawinski, K., Slosar, A., et al. 2008, MNRAS, 389, 1179
Luken, K. L., Norris, R. P., & Park, L. A. F. 2019, PASP, 131, 108003
Lukic, V., Brüggen, M., Banfield, J. K., et al. 2018, MNRAS, 476, 246
Mahabal, A. A., Djorgovski, S. G., Drake, A. J., et al. 2011, BASI, 39, 387
Meinshausen, N. 2006, J. Mach. Learn. Res., 7, 983
Mosteller, F., & Tukey, J. W. 1968, in Handbook of Social Psychology, ed. G. Lindzey & E. Aronson, Vol. 2 (Boston, MA: Addison-Wesley)
Norris, R. P. 2017a, NatAs, 1, 671
Norris, R. P. 2017b, PASA, 34, e007
Norris, R. P., Hopkins, A. M., Afonso, J., et al. 2011, PASA, 28, 215
Norris, R. P., Salvato, M., Longo, G., et al. 2019, PASP, 131, 108004
Pedregosa, F., Varoquaux, G., Gramfort, A., et al. 2011, J. Mach. Learn. Res., 12, 2825
Polsterer, K. L., Gieseke, F., & Igel, C. 2015, in ASP Conf. Ser. 495, Astronomical Data Analysis Software an Systems XXIV (ADASS XXIV), ed. A. R. Taylor & E. Rosolowsky (San Francisco, CA: ASP), 81
Polsterer, K. L., Gieseke, F., Igel, C., Doser, B., & Gianniotis, N. 2016, European Symp. on Artificial Neural Networks, Vol. 24
Richter, G. A. 1975, AN, 296, 65
Segal, G., Parkinson, D., Norris, R. P., & Swan, J. 2019, PASP, 131, 108007
Torniainen, I., Tornikoski, M., Turunen, M., et al. 2008, A&A, 482, 483
Traven, G., Matijevič, G., Zwitter, T., et al. 2017, ApJS, 228, 24
Way, M. J., Gazis, P. R., & Scargle, J. D. 2011, ApJ, 727, 48
Weir, N., Fayyad, U. M., & Djorgovski, S. 1995, AJ, 109, 2401
Wells, D. C., Greisen, E. W., & Harten, R. H. 1981, A&AS, 44, 363
Weston, S. D., Seymour, N., Gulyaev, S., et al. 2018, MNRAS, 473, 4523
Wright, E. L., Eisenhardt, P. R. M., Mainzer, A. K., et al. 2010, AJ, 140, 1868
Wu, C., Wong, O. I., Rudnick, L., et al. 2019, MNRAS, 482, 1211
York, D. G., Adelman, J., Anderson, J. E., Jr., et al. 2000, AJ, 120, 1579

[17] https://reproject.readthedocs.io/en/stable/