Classifying Galaxy Morphology with Deep Learning Algorithms

Evan Imata

ABSTRACT

Due to the importance of galaxy morphology for cosmological research and the abundance of galactic image data, automated methods are required to perform large-scale morphology classification tasks. Here, this issue is addressed by deploying deep learning models to automate the classification process. A custom-built Convolutional Neural Network is trained on galaxy images from the Sloan Digital Sky Survey and compared to data from the Galaxy Zoo 2 project using a root-mean-squared (RMSE) loss metric. The model is trained to identify morphological features such as galaxy mergers, bulges, disks, and spiral arms. This neural network achieves an RMSE loss of 0.1067 on a set of validation images. To atone for the flaws in the first model, a Residual Neural Network is implemented, using a fine-tuned ResNet50 model which achieves an RMSE loss of 0.1193 and reduces over-fitting. The rate of galaxy merger images (z) in the data set is predicted by the ResNet to be $z = 0$. This paper analyzes the performance of different deep learning models and their reliability for executing morphology classification tasks.

1 INTRODUCTION

The morphological classification of galaxies provides insight into unresolved questions in astronomical areas such as star formation, galaxy evolution, and cosmological models of the universe. Studying morphology involves classifying galaxies by observable characteristics, like bulge and disk features, or mergers.

A merger occurs when two galaxies collide and interact. This phenomenon is relevant to active areas of study because merger events may serve as a driving force for rapid star formation and the formation of new galaxies. Thus, obtaining a galactic merger rate, along with classifying other morphological features is essential to understanding the history of galaxy and stellar evolution in the universe.

1.1 Galaxy Zoo 2 Classifications

Galaxy Zoo 2 (GZ2) is a web-based galaxy classification project, in which volunteer voters are shown images of galaxies from the Sloan Digital Sky Survey (SDSS) Data Release 7 (DR7) and answer boolean yes/no questions pertaining to galaxy morphology. The images are shown to voters in a mostly random order; however, certain images that have been classified less may be shown more frequently. A limited sample of SDSS images are selected for GZ2 classification. The GZ2 sample *excludes* SDSS galaxies that have a Petrosian half-light magnitude > 17 in the r-band, a size below petroR90_r < 3 in units of arcseconds, (where **petroR90_r** is the radius that contains 90% of the r-band Petrosian aperture flux), a red-shift (z) value outside the range $0.0005 < z < 0.25$, and objects flagged by SDSS as "SATURATED", "BRIGHT", or "BLENDED" without a "NODEBLEND" flag.

GZ2 Decision Tree			
Class	Question	Responses	Next Class
1	Is the galaxy simply smooth and rounded, with no sign of a disk?	smooth features or disk star or artifact	7 2 **end**
2	Could this be a disk viewed edge-on?	yes no	9 3
3	Is there a sign of a bar feature through the centre of the galaxy?	yes no	4 4
4	Is there any sign of a spiral arm pattern?	yes no	10 5
5	How prominent is the central bulge, compared with the rest of the galaxy?	no bulge just noticeable obvious dominant	6 6 6 6
6	Is there anything odd?	yes no	8 **end**
7	How rounded is it?	completely round in between cigar-shaped	6 6 6
8	Is the odd feature a ring, or is the galaxy disturbed or irregular?	ring lens or arc disturbed irregular other merger dust lane	**end** **end** **end** **end** **end** **end** **end**
9	Does the galaxy have a bulge at its centre? If so, what shape?	rounded boxy no bulge	6 6 6
10	How tightly wound do the spiral arms appear?	tight loose medium	11 11 11
11	How many spiral arms are there?	1 2 3 4 more than four can't tell	5 5 5 5 5 5

Table 1: The morphology classification tasks given to GZ2 voters. Questions and responses are grouped into 11 different classes. The "question" column displays the questions that are asked when a galaxy image is shown to a voter, and "responses" shows the possible answers the voter may choose. The "next class" column shows which class GZ2 will transition to next, given the voter's response to the current class.

In GZ2, there are 37 possible morphological features, otherwise referred to as "labels." These labels are organized into 11 different classification tasks and sorted into a decision tree, in which certain labels are dependent on previous label classifications, as shown in Table 1. The number of questions that voters are presented depend on their answers, but typically range from 1-9. Previous projects, such as Galaxy Zoo 1 (GZ1) did not contain a decision tree structure and contained questions that were independent of each other.

To account for unreliable voters, GZ2 uses an iterative weighting scheme to weigh voter contributions. This process assigns each voters' response a consistency based on how close their answer is to the majority of other voters. The consistency for each answer is then averaged to assign a single consistency value for every user, which is used to weigh their contributions.

Classification bias arises from galaxies that are distant, dimmer, and red-shifted. Red-shifting in a galactic context is a phenomenon in which wavelengths of light from a galaxy are stretched due to the expansion of space, thus making galaxies appear redder in color. Higher red-shift galaxies are also smaller in the GZ2 images as a result of being further away. Therefore, some morphological features of these galaxies are more challenging to identify. This causes a bias within the voting which expert and machine classification is not immune to. GZ2 accounts for this classification bias by assuming that galaxies of similar brightness and size will share similar morphologies. The ratio between pairs of classification responses can be converted into a correction factor that addresses classification bias. (Conselice, 2014)

Galaxies that have relatively higher red-shifts may look close together, but may not be interacting. They appear close because the angular separation between them is reduced as a result of distance. Voters may incorrectly assume that these are mergers, which is known as **"angular separation bias."** (Conselice, 2014)

1.2 Convolution Neural Networks

Rather than classifying large galactic data sets by eye, it is more efficient to implement a convolutional neural network (CNN) that automates this classification process. The CNN is a deep learning algorithm that is trained to identify the morphological labels desired. In this procedure, the Keras open-source library will be used to construct the CNN model architecture. The model simulates layers of neural nodes that are interconnected with different strengths and weights. The neural network learns to classify the 37 labels by being shown GZ2 galaxy images and outputting 37 values. These outputs are compared to the pre-determined label classifications from GZ2, which are probabilities between 0.0 and 1.0. This comparison serves as a basis for evaluating and improving the performance of the CNN model, which will be discussed further.

2 METHODS

2.1 GZ2 Training Data

In order to train the neural network model, the CNN needs to be shown sample images of galaxies. A total of 61,579 RGB galactic images of size 424 × 424 pixels are downloaded from GZ2.

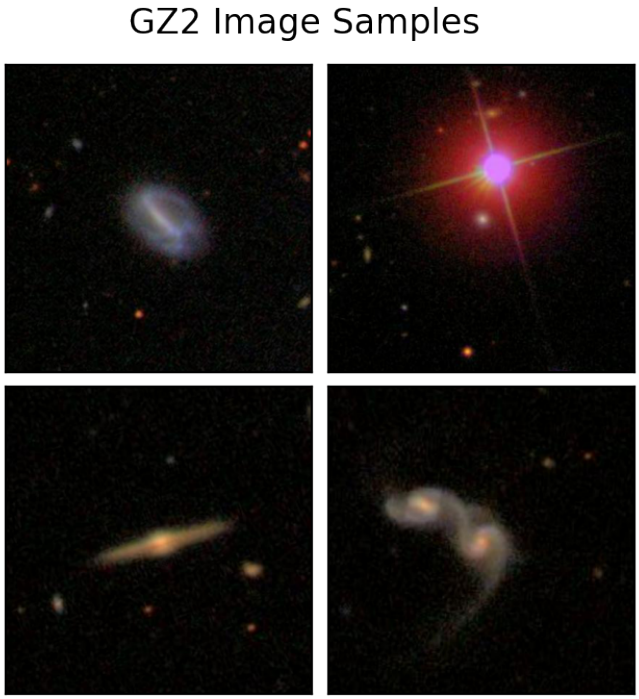


Figure 1: Examples of random images from GZ2. Each image is already pre-processed, such that all galaxies are properly centered and scaled in size.

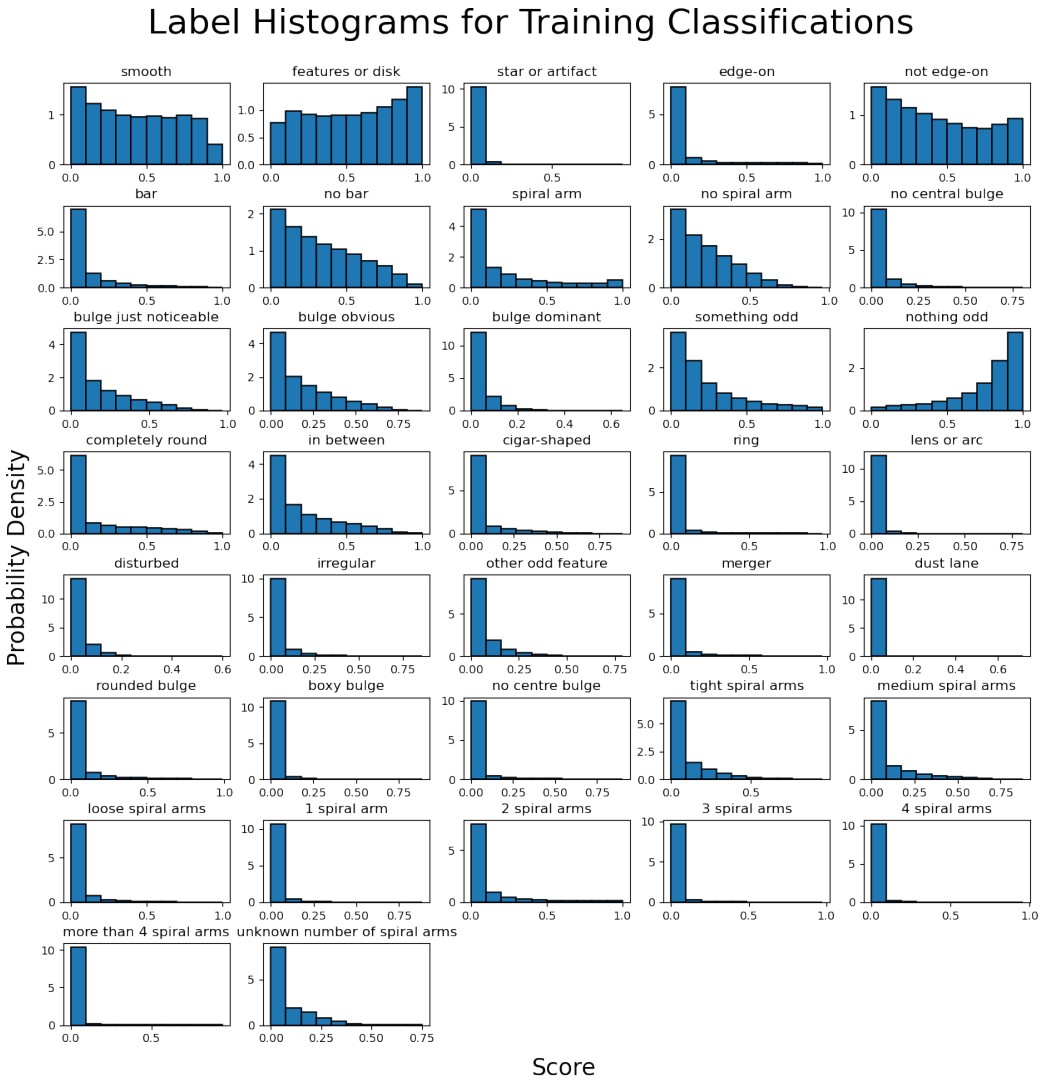


Figure 2: Histogram plots for each of the 37 galactic label classifications from GZ2 with 10 bins. Each histogram corresponds to a different label and displays the probability density distribution of the 61,579 training set galaxies. The x-axis indicates the scores given by voters — for example, for the label "smooth" in the upper left corner, a score of 0.0 indicates that voters ranked those galaxies as being not smooth, and a score of 1.0 indicates galaxies that have been ranked as being smooth. These scores are averages of all the voters' rankings, weighted by voter consistency and dependability, thus providing a continuum of rankings, rather than a binary 0 or 1. The y-axis indicates the probability density, which is the probability of the label between 0 and 1 divided by the bin width, which is 0.1. Based on voter classifications, many of the galaxies in this data set are edge-on (row 1, column 4). Meanwhile, there is more of a spread in terms of feature/disk morphology (row 1, column 2). In the figure, there are two different labels that both pertain to "no central bulge" (row 2, column 5 and row 6, column 3). This is from the GZ2 classification scheme, where one of these relates to the existence of a galactic bulge and the other relates to the shape of the bulge.



Figure 3: "Prototype" galaxies for each of the 37 labels. A prototype galaxy is the galaxy in the training set that contains the highest ranking for a particular label. For example, the galaxy on row 2, column 1 is the image that received the highest weighted average probability ranking for having a "bar" feature. Based on this figure alone, labels such as "edge-on" and "smooth" seem to be more visually apparent and easier to classify, while labels pertaining to the number of spiral arms and mergers may be more challenging. Each image is titled with the respective label, as well as the galactic image ID.

Some of the galactic labels are intended to be mutually exclusive. For instance, a galaxy will never simultaneously have a disk and be smooth. The correlation between different labels in the training set is given by

$$\rho_{ij} = \frac{\langle ij \rangle - \langle i \rangle \langle j \rangle}{\sqrt{\langle i^2 \rangle - \langle i \rangle^2} \sqrt{\langle j^2 \rangle - \langle j \rangle^2}} \quad \text{(Eq. 1)}$$

where ρ_{ij} is the correlation score, i and j are labels — for example, i could be an array of all of the "smooth" label rankings in the training set, $\langle i \rangle$ is the average value of the i labels, and $\langle ij \rangle$ is the average value of the element-wise product of i and j.

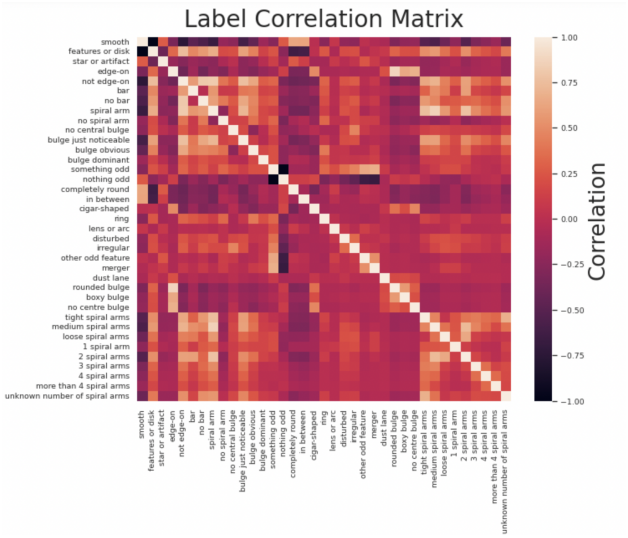


Figure 4: A 37 × 37 correlation matrix for each of the labels. The correlation between each label is calculated using Eq. 1 and represented as a square in the matrix. As indicated in the color bar, an apricot-colored square implies a perfect correlation of 1.00. There is a diagonal line through the matrix showing perfect correlation, since the diagonal represents the correlation between a label and itself, which is expected to be 1.00. As expected, the label "smooth" has no correlation with "features or disk," as the correlation square appears to be -1.00. The label "edge-on" correlates well with "rounded bulge," which is due to the rounded bulge label being dependent on the edge-on label in the GZ2 decision tree.

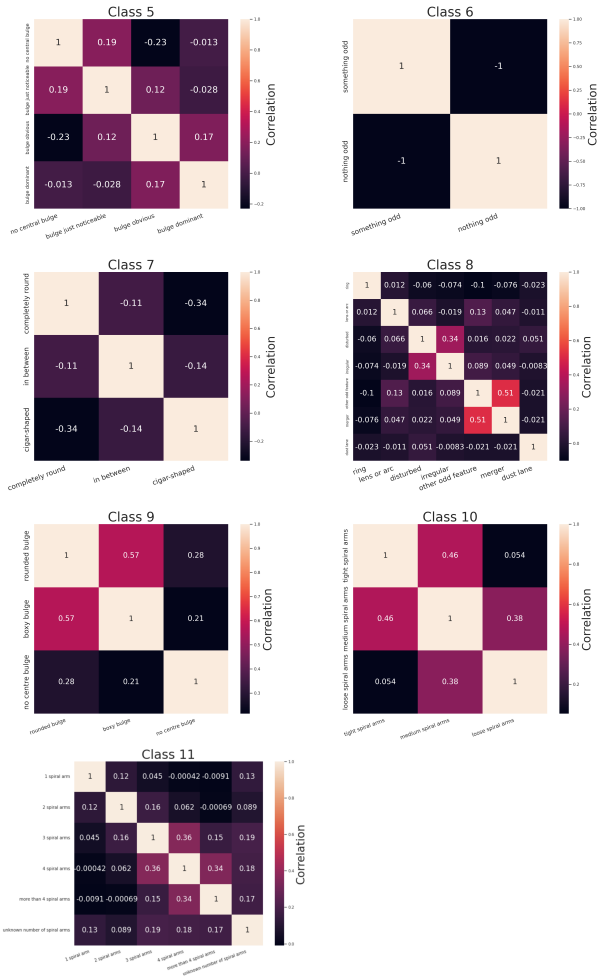
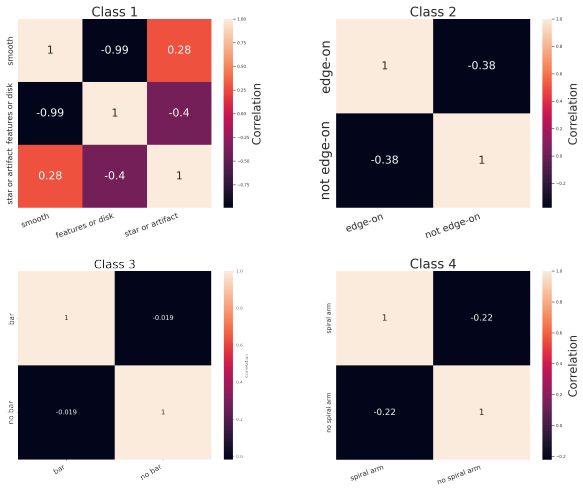


Figure 5: Correlation matrices/heatmaps for the labels, sorted into the classes outlined in Table 1. Once again, an apricot-shade indicates a perfect correlation of 1.00, while a near-black shade indicates no correlation, or -1.00. Unexpectedly, in Class 8, the "merger" label shows small correlation with "other odd feature." This may be a result of classification bias, where the morphological merger features are not as obvious in higher red-shift galaxies. This could have impacts on the galactic merger rate that will be later determined.



2.2 Data Preparation

The quantity of training data is too large for the environment that the CNN resides in. The raw images are 539,328 bytes each, for a total of 539328 × 61579 ≈ 33 gigabytes (GB). The environment contains only 4 GB

of memory to handle image data. Thus, images are downsized. The x and the y axes of the image grid are scaled to a factor 0.23 of the original size. This factor down-scales the image while still preserving enough quality to distinguish morphological features. Since all galaxy images are pre-centered, each image is uniformly cropped, removing margins from all sides. This process results in a data set of down-scaled images that are of size $80 \times 80 \times 3$ and 19200 bytes each. The entire data set of resized images will occupy 1.18 GB, which fits into the memory of the environment with 3 GB leftover.

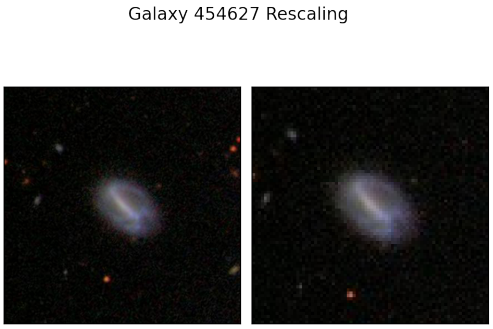


Figure 6: An image of Galaxy 454627 before down-scaling (left) and after (right). The image quality in the down-scaled version is reduced to a coarser pixel grid; however, key features of the galaxy's morphology remains distinguishable. The down-scaled image appears larger due to cropping.

The data is then split into a training and validation set, with the training set containing 80% of the galaxies (49,263 galaxies) and the validation set containing 20% (12,316). The labels from each of these sets is fed into separate Keras **ImageDataGenerator** objects, which yield batches of galaxy labels and their respective images at a time while the CNN is training to further conserve computational memory. Here, the generators are set to yield 100 images at a time for the neural network to train on. Once the training is finished for that batch, the 100 images are purged from memory and replaced with a new batch.

2.3 Loss Function

Before constructing a CNN model, a loss-function needs to be defined. A loss-function evaluates the performance of the CNN during training by comparing the outputs of the neural network to the GZ2 voter classifications. The loss-function that will be used for this galaxy classification procedure is root-mean-squared-error (RMSE) loss, which is defined as

$$L_{\text{RMSE}} = \sqrt{\frac{1}{N_g N_l} \sum_{i}^{N_g} \sum_{j}^{N_l} (l_{\text{true},ij} - l_{\text{pred},ij})^2} \quad \text{(Eq. 2)}$$

where N_g is the number of galaxies, N_l is the number of labels (37), $l_{\text{true},ij}$ is the value of label j for galaxy number i from GZ2, and $l_{\text{pred},ij}$ is the value of label j for galaxy number i predicted by the CNN model. The loss-function essentially averages the squares of the differences between the GZ2 and CNN label values for every galaxy in the data set, then takes the root of that average.

The objective is to have the CNN model minimize this loss function. A series of training images are passed through the neural layers of the CNN. After, the CNN makes predictions on images it has not seen before from the validation set, and those predictions are compared to the real label values from GZ2 using the loss function. Evaluating the validation loss at every step in the training of the CNN model will allow the neural network to readjust its weights and connections between different neural layers, changing the outputs, and take steps toward smaller RMSE values in a process called "back propagation." The partial derivatives of the loss function with respect to each label forms a gradient, and the

CNN undergoes "stochastic gradient descent" by stepping in the opposite direction of the estimated gradient until it finds a minimum in the loss function. Once this is achieved, the CNN will be able to perform predictions on galaxy morphology with reasonable accuracy.

A benchmark for RMSE can be determined by taking $l_{\text{pred},ij}$ in Eq. 2 to be the average of each label in the GZ2 data set. Such a crude assumption gives an RMSE loss of $L_{\text{RMSE}} = 0.1639$. The CNN should be able to achieve a loss lower than this value.

2.4 Building a Custom CNN

Now, a CNN model is constructed in Keras. The model architecture begins with a 2-dimensional convolutional layer that takes batches of images that are $80 \times 80 \times 3$ as inputs. This layer slides a kernel of size 3×3 pixels across the images and picks up on visual features. An odd kernel size is chosen so that the center of the kernel is a pixel, rather than an intersection of four pixels (as would be the case in an even 2×2 kernel). The odd kernel size allows for higher detection of image features. The stride at which the kernel moves across the image is 1 pixel in both the width and height directions. This layer is then followed by a Rectified Linear Unit (ReLU) activation layer, which sets negative numerical inputs to 0.0 while leaving positive inputs intact, and a 2-dimensional max pooling function, which down-scales the input from the ReLU layer along the images' height and width axes by pooling together groups of pixel inputs of size 2×2 and taking the maximum. This pattern of 3 layers is repeated five times. The first convolutional layer begins with 32 output filters, but doubles in quantity with every repeated layer to capture the increasing amount of patterns being picked up by the neural network.

The outputs from this series of convolutional layers is then flattened into a 1-dimensional layer (using the Keras neural layer "Flatten"), which then passes numerical inputs into fully-connected layers. These fully-connected layers consist of three ReLU layers. These layers gradually decrease the size of the CNN outputs and approach the 37 outputs desired (which correspond with the 37 labels). The final quantity of 37 outputs is finally achieved in an ending sigmoid layer, which forces all outputs to be in between 0.0 and 1.0. Within this numerical range, the outputs will resemble probabilities, which aligns with the label classification values from GZ2. This will make it possible to compare the outputs of the CNN to the label values from GZ2 in the RMSE loss function. In between the fully-connected layers, Dropout layers are added. Dropout shuts off certain neural nodes, which prevents the CNN from over-fitting. Over-fitting occurs when a neural network simply memorizes the training data and performs poorly when shown a sample of new data that it has never seen before. Here, dropout layers are set at a rate of 0.2, which is the fraction of nodes in a layer that become shut off. The CNN model is then compiled using the Adam optimization algorithm for performing stochastic gradient descent.

The model is trained by feeding in data using the Keras ImageDataGenerator object, which yields batches of both the training images and their associated GZ2 labels to assess RMSE loss. The CNN is set to complete 50 training iterations, or "epochs." In one epoch, data is inputted in batches of 100 galaxies at a time, for a total 492 batches, or training steps per epoch — since there are 49,263 training images, 100 galaxies per batch $\times$ 492 batches per training step $= 46,200$ galaxies per training step. Thus, some galaxies from the training set are not included in every single epoch. This effect is negligible,

since the excluded galaxies only amount to $\frac{49263-49200}{49263} = 0.00128$ of the entire training set.

This CNN model and all of its neural layers contains 5,206,685 trainable parameters. The number of trainable parameters must not exceed the number of total training data parameters ($80 \times 80 \times 3 \times 49,263 = 945,849,600$), otherwise the model will be over-fitting.

2.5 ResNet50

The Residual Neural Network (ResNet) is a neural network architecture that has proven to perform well with image classification. ResNet achieves stronger accuracy due to its implementation and stacking of "skip layer connections", where a layer output skips one or two layers, using an identity function and vector addition. This preserves and passes on a greater quantity of gradient context between layers, which improves the back propagation process and the performance of the neural network as a whole. The skip layer connection is more efficient than simply stacking more layers onto a neural network, since the latter choice still does not preserve context information about the gradient at different layers.

The ResNet50 — a ResNet with 50 layers — is loaded and fine-tuned to the GZ2 data set as an alternative to the custom CNN that has been constructed in Section 2.4.

2.6 Improving Model Performance

Several methods are implemented to improve the ResNet model's performance on the GZ2 data set. Multiple techniques are used to lessen the effects of over-fitting during CNN training, including data augmentation and learning rate schedulers. The outputs of the ending sigmoid layer are also reweighed to more accurately reflect the GZ2 classification scheme.

2.6.1 Data Augmentation

The quantity of training set images is artificially expanded by randomly flipping images along the vertical and horizontal axes and rotating images randomly between 0.0°and 360°during the training process. During random rotation, the image is not zoomed in, so there are empty corners; this will not disturb the neural network's capacity to learn morphology. Augmented the data in this way reduces the ability of the CNN to memorize the training set data, therefore reducing overfitting. These augmentation techniques are set as parameters in the Keras ImageDataGenerator object, by setting

```
vertical_flip=True
horizontal_flip=True
rotation_range=360.
```

Other augmentation options include translating or rescaling the images, but this is not necessary to include, as all images have already been standardized by GZ2 to be centered and scaled.

2.6.2 Learning Rate Scheduler

All models thus far have been initiated with a learning rate of 1e-3 as a parameter within the Adam optimizer for gradient descent. The "learning rate" of the neural network is a parameter that controls the iterative step size of the CNN as it moves toward the minimum of the loss function gradient. In this CNN model, the Keras learning rate scheduler **ReduceLROnPlateau** is used to decrease the learning rate if the validation loss between epochs begins to plateau. Between epochs, if the RMSE loss does not decrease by a parameter of at least **min_delta,** then the learning rate is decreased by the parameter **factor**. Decreasing the learning rate will allow the

CNN to pick up on finer details in the gradient descent and reduce RMSE. A lower limit to the learning rate is set with the parameter **min_lr**, as it is unfavorable to have the neural network's steps toward the loss minimum be too small. The **patience** parameter defines how many epochs the scheduler will wait after a loss plateau begins before reducing the learning rate. In this scheduler, the parameters are set such that

```
min_delta = 0.005
factor = 0.5
min_lr = 1e-6
patience = 5
```

When training a new ResNet, an initial learning rate of 1e-3 is suitable. However, a ResNet may also be initiated with predefined neural weights, such as the "ImageNet" weights, which have already been trained and have proven to be successful for image classification (Krizhevsky et al., 2020). When using such weights, the initial learning rate during training must be set to a lower value, such as 1e-4 or 1e-5. High learning rates will destroy and change these weights, but low learning rates will simply "fine-tune" them. If the goal is to add on fully-connected layers at the end of the ResNet (such as the sigmoid layer), the learning rate can be set at 1e-3 while the ImageNet weights are frozen, which will only train the newly added layers. Then, a second round of training can be completed after unfreezing all layers and setting the initial learning rate low to fine-tune the entire model.

2.6.3 Reweighing Sigmoid Outputs

The ending sigmoid layer forces all outputs to be probabilities between 0.0 and 1.0. However, this does not entirely reflect the nature of the decision tree classification scheme from GZ2. For example, in Table 1, if the probability of a galaxy having features or a disk in Class 1 is 0.5, then the probabilities of the responses in Class 2, which assumes that there *is* a disk in the first place, must sum up to 0.5. The probability of a disk being viewed edge-on being higher than the probability of a disk existing at all would not make sense. Therefore, labels that depend on the probability of a prior label must be normalized to this corresponding prior label.

Label Reweighing Scheme	
Prior Label	Labels to Normalize
Class 1: Features or disk on galaxy	Class 2: Disk viewed edge-on Disk NOT viewed edge-on
Class 2: Disk NOT viewed edge-on	Class 3: Bar feature through galaxy center No bar feature through galaxy center Class 4: Sign of spiral arm pattern No sign of spiral arm pattern Class 5 : No bulge Just noticeable bulge Obvious bulge Dominant bulge
Class 1: Galaxy is smooth and rounded	Class 7: Completely round In between Cigar-shaped
Class 6: Odd feature	Class 8: Ring feature Lens or arc Disturbed Irregular Merger Dust lane Other odd feature
Class 2: Disk viewed edge-on	Class 9: Rounded bulge Boxy bulge No bulge
Class 4: Sign of spiral arm pattern	Class 10: Tight spiral arms Loose Medium Class 11: 1 spiral arm 2 spiral arms 3 spiral arms 4 spiral arms More than 4 spiral arms Can't tell how many spiral arms

Table 2: The right column of the table ("labels to normalize") displays labels that need to be normalized, and the left column ("prior label") shows the label value that the right column needs to be normalized to. For instance, if the prior label value for Class 1 - "features or disk on galaxy" is 0.75, then the Class 2 labels to normalize - "disk viewed edge-on" and "disk NOT viewed edge-on" *must* then sum to 0.75. It is critical to normalize the labels in the order of the table from top-to-bottom, because some of the labels that are to be normalized (such as "disk viewed edge-on") are later used in the prior label column as the standard to normalize other labels.

Using the reweighing scheme presented in Table 2, the ending sigmoid layer of the ResNet is modified so that its outputs are normalized to the correct labels. This is done by creating a new function that takes sigmoid layer inputs and normalizes outputs in the order of Table 2 from top to bottom. At each step in the normalization procedure, the custom-defined reweighing function creates and modifies a *normalization mask* of coefficients that is calculated and multiplied element-wise to the inputs from the sigmoid layer, resulting in the correctly normalized outputs.

3 RESULTS

3.1 Custom CNN Performance

The custom CNN, described in Section 2.4 is now constructed, trained, and evaluated.

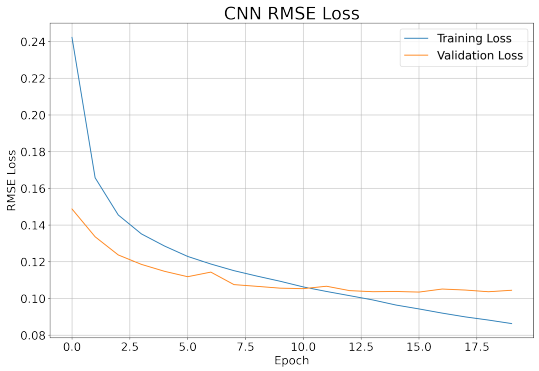


Figure 7: The training and validation loss functions during the first training round of the custom CNN model, where the x-axis displays the epoch during training and the y-axis displays the RMSE loss. At around epoch 10, the validation loss (orange) begins to plateau, while the training loss (blue) continues to decline. This is evidence of over-fitting, as the model continues to learn more and memorize the training set while maintaining the same level of performance in recognizing new images in the validation set. This model plateaus at a validation loss of $L_{\text{RMSE}} = 0.1067$.

To reduce over-fitting in the model shown in Fig. 7, two more Dropout layers with dropout rates of 0.5 are added to shutoff neural nodes and prevent memorization. The number of epochs is also increased to 50.

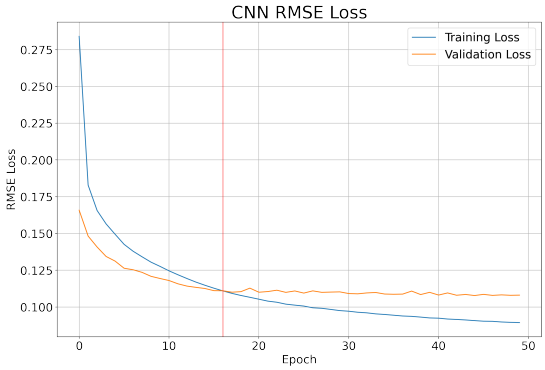


Figure 8: The training and validation loss functions during the second round of training for the custom CNN model with 50 epochs and two additional dropout layers, where the axes are the same as Fig. 7. The epoch where over-fitting begins is now shifted to epoch 16, indicated by the vertical red line. This exemplifies the ability of dropout layers to reduce over-fitting; however, the two additional dropouts were not effective enough to prevent over-fitting entirely. This model plateaus at a validation loss of $L_{\text{RMSE}} = 0.1081$. This is slightly worse than the previous CNN, which is an effect of the dropout layers shutting off neural nodes. The higher validation loss is the cost of the reduction in over-fitting.

Adding dropout layers helped to postpone the onset of over-fitting, but did not extinguish it. Rather than adding more dropout layers, the ResNet is now implemented with alternative methods to reduce over-fitting.

3.2 ResNet50 Performance

The ResNet50 is downloaded and trained on the GZ2 data. During the loading of the ResNet, the image input shape ($80 \times 80 \times 3$) is specified, as well as the sigmoid output size of

37. The otherwise unaltered, out-of-the-box ResNet is trained initially without any predefined weights, data augmentation, learning rate schedulers, or sigmoid layer reweighing.

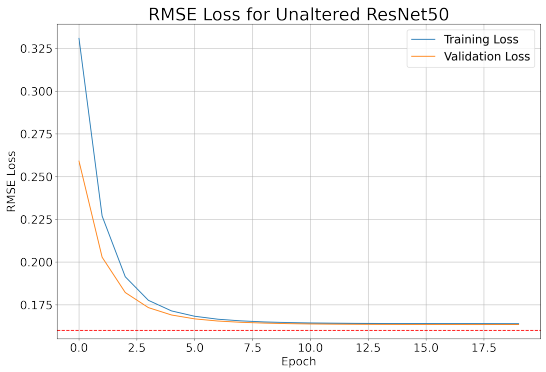


Figure 9: The training and validation loss for the unaltered ResNet50. The training is only run for 20 epochs, as the training and validation loss both plateau within this range of training iterations. The red-dotted line indicates the benchmark 0.16 RMSE value from Section 2.3. The unaltered ResNet50 has not even surpassed this benchmark yet, so modifications are needed. This unaltered CNN plateaus at and has an ending RMSE loss of $L_{\text{RMSE}} = 0.1635$. The validation RMSE plateau is not low enough because reweighing and the learning rate scheduler have not yet been applied. Adding these should help to lower the RMSE floor. The training and validation loss plateau together and do not cross, so the ResNet50 model has avoided over-fitting so far. This is not a result of dropout layers — the ResNet does not contain any — but rather, the result of skip layer connections in a deeper network architecture.

To lower the RMSE floor of the ResNet50 and improve the model's image classification accuracy, data augmentation, learning rate schedulers, and sigmoid layer reweighing are applied, as described in Sections 2.6.1, 2.6.2, and 2.6.3. In addition, the ResNet50 is re-initiated with the ImageNet weights. The implementation of the ImageNet weights causes a large, unchangeable output size in the ResNet. As a result, additional fully-connected layers are added on to slowly scale down ResNet outputs into the 37 reweighed sigmoid outputs.

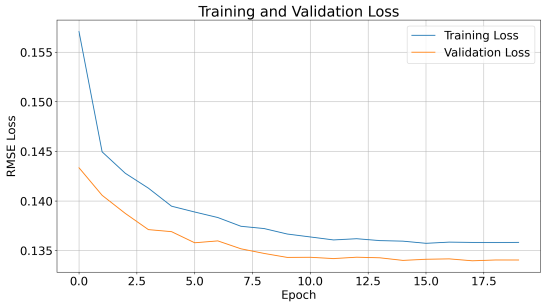
Layer (type)	Output Shape	Param #
resnet50 (Functional)	(None, 3, 3, 2048)	23587712
flatten_4 (Flatten)	(None, 18432)	0
dense_49 (Dense)	(None, 512)	9437696
dense_50 (Dense)	(None, 256)	131328
dropout_29 (Dropout)	(None, 256)	0
dense_51 (Dense)	(None, 128)	32896
dense_52 (Dense)	(None, 64)	8256
dense_53 (Dense)	(None, 37)	2405

Total params: 33,200,293
Trainable params: 9,612,581
Non-trainable params: 23,587,712

Figure 10: A summary of the ResNet50 model with fully-connected layers added. Additionally, a dropout layer is included to prevent over-fitting. The total number of parameters is 33,200,293, which does not exceed the number of training data points, even when all layers are unfrozen. Thus, the model is not in immediate jeopardy of over-fitting.

Because the objective is to predict galaxy labels that have probabilities between 0.0 and 1.0, not only should the benchmark of a 0.16 RMSE loss from Section 2.3 be surpassed; an RMSE of < 0.1 is now desirable. If the maximum possible output is 1.0, then an RMSE *greater* than 0.1 is not negligible and should be improved as much as possible.

First, the added fully-connected layers are trained by freezing the ImageNet weights and training the model for 20 epochs with a learning rate of 1e-3.



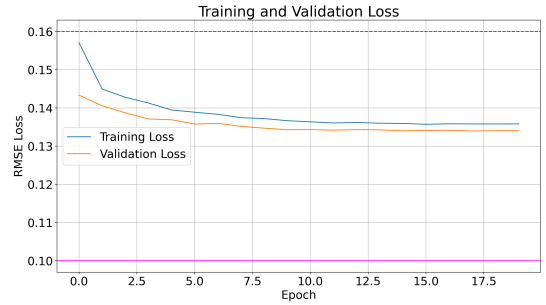
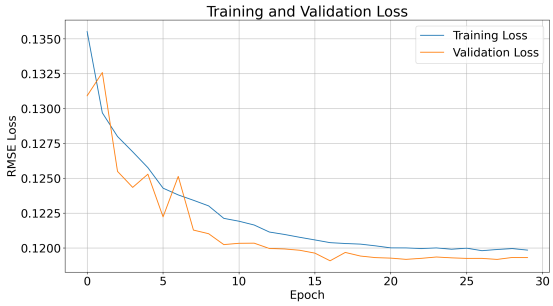


Figure 11: Training and validation RMSE loss functions for the model, with the ResNet50 layers frozen. The top and the bottom figures both display the same training session of the ResNet's added fully-connected layers. However, the bottom figure provides more context. In the bottom figure, the red-dotted line at $y = 0.16$ shows the crude benchmark loss from Section 2.3. The training and validation loss have gone below this threshold, which was desired. This was not achieved by the unaltered ResNet50 in Fig. 9. The horizontal magenta line at $y = 0.1$ shows that the goal of < 0.1 RMSE has not been achieved by the current model, which has yet to be fine-tuned. In the top figure, there is a visual gap between the training and validation loss. This could possibly be indicative of too much dropout, which may worsen the model's overall ability to predict labels in the future.

Now, all of the layers of the ResNet model are unfrozen and fine-tuned by training the model again with an initial learning rate of 1e-4 in the Adam optimizer. This fine-tuning session is run for 30 epochs.



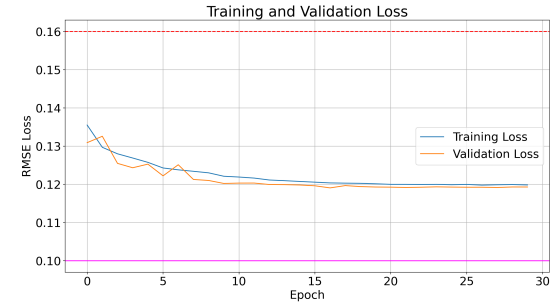


Figure 12: Training and validation RMSE loss functions for the fine-tuned ResNet. Both plots display the same fine-tuning session, where the bottom plot again provides more context. The red-dotted line at $y = 0.16$ gives the RMSE benchmark from Section 2.3, while the magenta line at $y = 0.1$ gives the maximum RMSE value that was desired. The model was not able to achieve a validation loss lower than this threshold. The validation RMSE plateaus and ends at a value of $L_{\text{RMSE}} = 0.1193$. In the top plot, there is still a gap between the training and the validation loss, possibly showing the permeation of the dropout layer effects.

Model Performance	
Model	Loss (L_{RMSE})
Custom CNN (Round 1)	0.1067
Custom CNN (Round 2)	0.1081
Unaltered ResNet50	0.1635
Fine-Tuned ResNet	0.1193

Table 3: A summary of the validation losses for each model trained. At a glance, either of the custom CNN's appear to be the strongest for image classification. However, after inspection of Fig. 8 and Fig. 9, it is evident that the custom CNN has memorized

the training data set, and would not perform well on new data. A substantial amount of dropout would need to be added to the custom CNN to account for this over-fitting, but doing so would likely make the validation loss much higher as well.

After the ResNet50 with fully-connected layers is fine-tuned, the validation loss still plateaus above the desired RMSE loss threshold.

The final validation loss of this fine-tuned ResNet model with data augmentation, reweighed sigmoid layers, and learning rate schedulers is lower than the unaltered ResNet50 from Fig. 9. Though the custom CNN achieves a validation loss closer to the desired $L_{\mathrm{RMSE}} < 0.1$ threshold, the custom CNN is also over-fitted, and would not be successful in making label predictions on galaxy images that it has never seen before. Thus, the final, fine-tuned ResNet is chosen to make label predictions.

3.3 Validation Set Label Predictions

The fine-tuned ResNet is used to predict labels for the 12,316 validation set images.

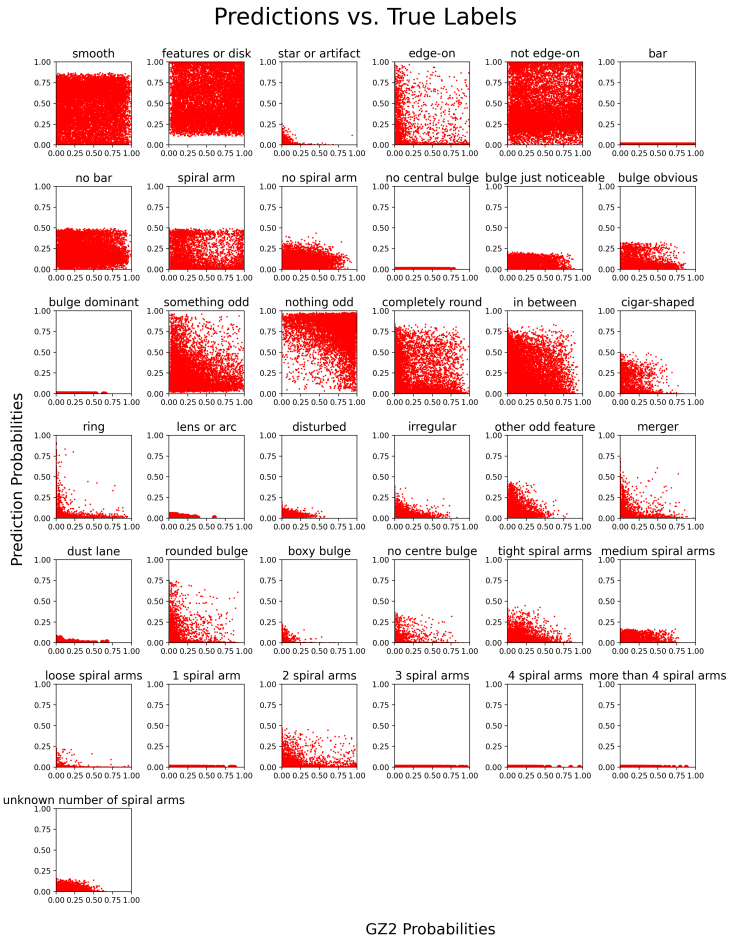


Figure 13: The predicted validation set label probabilities plotted against the GZ2 validation label values.

In Fig. 13, the predictions for "star or artifact," "bar," and predictions related to bulge and spiral arms perform particularly poorly. For the labels mentioned, the GZ2 classification represents a spread of probabilities, whereas the ResNet model predicts < 0.1 or 0.0 for all images, which is likely incorrect. These ResNet prediction flaws are visualized by the label values being sprawled across the horizontal axis but confined to the bottom of the vertical axis, such as with the "bar" label or the "3 spiral arms" label. Most of the labels related to the quantity of spiral arms are predicted to be 0.0 by the model; however, the label "2 spiral arms" is an exception to this. These phenomena may be a result of the decision tree in Table 2, since most of these poorly predicted labels are dependent on the "smooth" and "features or disk" probabilities. From observation of Fig. 13, these first two labels themselves are not well-predicted. For instance, the model predicts that no galaxies have a 1.0 probability of being smooth, evidenced by the horizontal gap in the "smooth" subplot on row 1, column 1. Even though the model predicts no probabilities of 1.0 for this label, GZ2 classifications have many galaxies that are ranked as such. The features or disk label has the opposite issue — it predicts that no galaxies have close to a 0.0 probability of having features or a disk. Since the the "smooth" and "features/disk" ResNet predictions are inherently flawed, this also leads to further issues in the following labels that are dependent on and normalized to these first two labels.

Additionally, the top 5 validation set prototype galaxies from the model predictions are compared with the top 5 prototypes from the GZ2 classifications for the labels: smooth, star or artifact, edge-on disk, ring, lens or arc, 2 spiral arms, and merger.

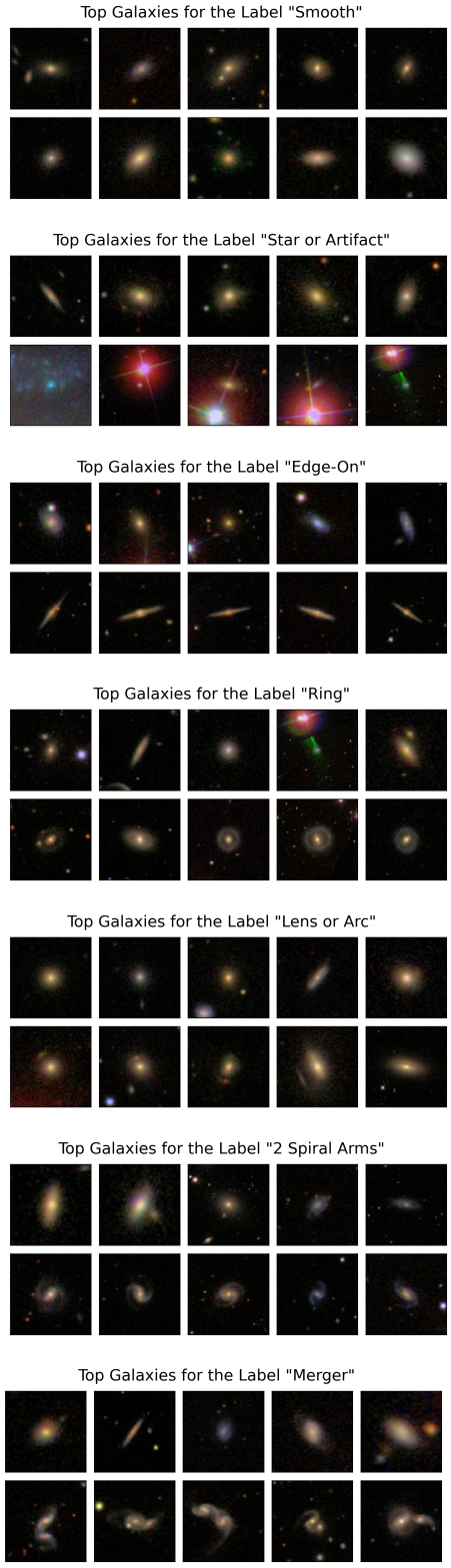


Figure 14: The top rows of each label plot group represent the five prototype galaxies predicted by the model, and the bottom rows represent the five proto-

types using GZ2 classifications. Visually, the prototypes predicted for "smooth" and "lens or arc" seem reasonable; however, the other label predictions do not appear to exhibit any features of their respective labels. For instance, the predictions for the "merger" label does not contain any obvious mergers. This will lead to issues in accurately estimating a merger rate for other data sets.

Although the fine-tuned ResNet appears to predict prototypes for the smooth label with a degree of reasonableness, based on the spread of the smooth label subplot in Fig. 13, the precision of the model is likely not as reliable on varying data sets.

3.4 Predicting the Merger Rate

Through visual inspection of the merger label predictions, a threshold is guessed to define the probability at which the ResNet predicts real mergers.

First, the prototype merger galaxy in the test set is calculated to have a merger probability of 0.74. Thus, a value of 0.6 is guessed as the threshold for real mergers (any galaxy in the validation set with a merger probability above this threshold should be a true merger image).

This threshold yields 8 galaxies in the validation set total. Upon visual inspection, only one of the images exhibits potential features of a merger.



Figure 15: Galaxy image ID: 193873. A galaxy from the validation set that is above the 0.6 probability threshold, which also appears to exhibit merger-like characteristics. However, this image looks like it may be a classification or angular bias scenario, in which two galaxies simply appear close together, but are not interacting or merging at all. The actual GZ2 merger prototypes in Fig. 14 are more visually definitive as mergers.

The image shown in Fig. 15 is the only galaxy of the 8 predicted above the threshold that bares resemblance to a merger. However, this galaxy is not the predicted prototype, and only has a predicted merger label value of $\sim$0.605. Thus, the threshold of 0.6 is not increased so that this potential merger is not excluded.

To predict a galactic merger rate, the fine-tuned ResNet now predicts the label values for a GZ2 test set of 79,900 galaxy images that it has not yet been shown. These images, just as in the training set, are pre-centered and scaled. Before the training set is loaded, it is down-scaled using the same procedure outlined in Section 2.2. These images are once again fed into the ResNet with the Keras ImageDataGenerator object and predicted on by using the Keras model predict method.

The maximum merger value predicted by the ResNet is 0.4, which falls short of the threshold. Using the merger threshold probability of 0.6, it is predicted by the ResNet that there are $z = 0$ merger galaxies in this test set. The $z = 0$ merger rate aligns with previous merger rate estimations of $z \approx 0$ (Lotz et al., 2011). However, prior estimations count the number of mergers over the time span of one gigayear (Gyr), whereas the merger rate calculated in this procedure is confined to the time window of SDSS, which is order of magnitudes narrower. Thus, the agreement between merger rates may not be indicative of any correlation between the two.

4 CONCLUSION

Though a merger rate has been predicted by the neural network, further work needs to be done to lower the RMSE loss function of the fine-tuned ResNet to make it more reliable for galaxy classification. This could possibly entail altering the fully-connected layers that were added onto the ResNet50 model with skip connections instead of dropout, using the Stochastic Gradient Descent (SGD) optimizer instead of Adam, or experimenting with deeper ResNets (such as the ResNet101 or the ResNet152). These options may provide improvements in the neural nets' ability to converge on a loss gradient minimum and reduce the weaknesses of the current implementation.

Additionally, physical phenomena such as red-shifting, which leads to classification bias should be incorporated into the training of the neural network. Classification bias and angular separation bias have effects on the model's ability to predict labels, and future work should attempt to correct for these effects.

With improvement, the galaxy merger rate predicted by the model will provide context for further research into stellar formation and the history of galaxy evolution in the observable universe. The CNN offers a more efficient approach to galaxy classification, but also has the potential to be expanded to other morphology tasks in the realm of space science and the cosmos.

5 REFERENCES

Conselice, Christopher J. "The Evolution of Galaxy Structure over Cosmic Time." Annual Review of Astronomy and Astrophysics, vol. 52, no. 1, 2014, pp. 291–337., https://doi.org/10.1146/annurev-astro-081913-040037.

"Galaxy Zoo 1 Data Release." Galaxy Zoo Data, https://data.galaxyzoo.org/.

He, Kaiming, et al. "Deep Residual Learning for Image Recognition." 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016, https://doi.org/10.1109/cvpr.2016.90.

Krizhevsky, Alex, et al. "ImageNet Classification with Deep Convolutional Neural Networks." Communications of the ACM, vol. 60, no. 6, 2017, pp. 84–90., https://doi.org/10.1145/3065386.

Lintott, Chris J., et al. "Galaxy Zoo: Morphologies Derived from Visual Inspection of Galaxies from the Sloan Digital Sky Survey." Monthly Notices of the Royal Astronomical Society, vol. 389, no. 3, 2008, pp. 1179–1189., https://doi.org/10.1111/j.1365-2966.2008.13689.x.

Lotz, Jennifer M., et al. "The Major and Minor Galaxy Merger Rates atzlt; 1.5." The Astrophysical Journal, vol. 742, no. 2, 2011, p. 103., https://doi.org/10.1088/0004-637x/742/2/103.

Masters, Karen L. "Twelve Years of Galaxy Zoo." Proceedings of the International Astronomical Union, vol. 14, no. S353, 2019, pp. 205–212., https://doi.org/10.1017/s1743921319008615.

Rodriguez-Gomez, Vicente, et al. "The Merger Rate of Galaxies in the Illustris Simulation: A Comparison with Observations and Semi-Empirical Models." Monthly Notices of the Royal Astronomical Society, vol. 449, no. 1, 2015, pp. 49–64., https://doi.org/10.1093/mnras/stv264.

Team, Keras. "Simple. Flexible. Powerful." Keras, https://keras.io/.

Willett, Kyle W., et al. "Galaxy Zoo 2: Detailed Morphological Classifications for 304nbsp;122 Galaxies from the Sloan Digital Sky Survey." Monthly Notices of the Royal Astronomical Society, vol. 435, no. 4, 2013, pp. 2835–2860., https://doi.org/10.1093/mnras/stt1458.

York, Donald G., et al. "The Sloan Digital Sky Survey: Technical Summary." The Astronomical Journal, vol. 120, no. 3, 2000, pp. 1579–1587., https://doi.org/10.1086/301513.

"Zooniverse." Zooniverse.org, https://www.zooniverse.org/projects/zookeeper/galaxy-zoo.