Self-Supervised Learning for Cross-Survey Galaxy Morphology Classification

G036 (s1918275, s1957945, s1950841)

Abstract

Advancements in deep learning for the image classification task has led to improvements on the galaxy morphology classification task. The current state-of-the-art method is a SimCLR architecture which uses all the best practises: data augmentation, self-supervised learning and fine tuning. This method performs well in the binary classification task for galaxy images however its performance in the multiclass classification task has yet to be explored. There are also problems with the datasets used for galaxy morphology classification as the quality of the images vary across different imaging surveys and the filtering process for these images lead to imbalanced distribution of labels. In this study, we apply the SimCLR and Autoencoder architecture to the multiclass classification task to assess the robustness of learned feature representations when adapting to new tasks. We experiment with Galaxy10 SDSS and DECals on a supervised model following the ResNet architecture and observe that our SimCLR and Autoencoder models outperform the baseline by 9% in classification accuracy. We demonstrate that the learned representations from our self-supervised models can generalize across different imaging surveys and can mitigate the effects of imbalanced datasets.

1. Introduction

Morphological classification of galaxies refers to the categorisation of galaxies based on visual appearance, structure and shape as observed from Earth. As galaxies exhibit a wide variety of different structures, they are mainly grouped into 5 classes: spirals, barred spirals, elliptical, lenticular and irregulars. Whilst galaxies have been primarily grouped into these 4 classes, advancements in galaxy evolution studies have shown that features observed at certain stages of a galaxy's life cycle can provide valuable insights into its evolution.

The Hubble classification, also known as the Hubble Tuning Fork Diagram, is an important tool in modern astronomy and helps scientists and astronomers to understand the structure and evolution of the galaxies. It also allows them to study the relationship between the morphological features such as shape, size and luminosity of galaxies and the underlying processes that drive their formation. This information sheds light onto the formation of the stars, the universe as a whole and how they change over time and what elements are required for life, further advancing research in extraterrestrial life.

Throughout history, the classification of galaxies was based solely on the visual appearance of the galaxies (Conselice, 2014). The Galaxy Zoo project, founded in July 2007, was created to allow volunteers to help quickly classify the galaxies in the Sloan Digital Sky Survey (SDSS) based on their shape. However, manually examining the images to classify the visual appearance of galaxies is a tedious and time-consuming process. Although the number of classifications performed by humans increased greatly, the actual classification itself was still slow. Additionally, it was slowly becoming impossible to perform classification due to the sheer volume of more detailed data produced by newer telescopes and much of the potentially useful data remained unused and unexplored (Barchi et al., 2020). In the past, the most common techniques used by astronomers were Fourier Transforms for temporal analysis, Least-Square Regressions or X2 goodness of fit. However, even these traditional methods were misused, leading to confusion when comparing studies (Feigelson & Babu, 2004). Hence, researchers developed automated classification techniques such as Convolutional Neural Networks (CNNs), Random Forest Classifiers and Ensemble Classifiers.

Prior research has only focused on specific sets of galaxy morphology, for example (Cavanagh et al., 2021) developed a CNN architecture and applied it to 4 of the 5 main classes of galaxies, i.e spirals, elliptical, lenticular and irregular. They achieved an overall accuracy of 81%. However, when this model was adapted to classify on a sample of the Galaxy Zoo dataset, specifically tailored towards ellipticals and spirals, their classification accuracy using the CNN architecture fell by 4%.

Another paper (Zhu et al., 2019) developed a variant of ResNet for the Galaxy Zoo 2 dataset which focuses on completely round smooth, in-between smooth, cigar-shaped smooth, edge-on and spiral galaxies. In this paper, they reported that VGG-16 and ResNet-50 achieved an accuracy of 93.13% and 94.09% respectively. However, (Kalvankar et al., 2020) expand the Galaxy Zoo 2 dataset to include two additional categories and evaluate the VGG-16 and ResNet-50 models on this dataset. Both models significantly decrease in performance, VGG-16 drops to 75.47% accuracy and ResNet-50 drops to 83.23% accuracy.

The main challenge these models fail to solve stems from the quality of labels for each galaxy. Labels for galaxy imaging surveys are based on researcher needs, as such,

labels vary greatly across different datasets. For example, Galaxy10 DECals contains a class of images for merging galaxies, whereas Galaxy10 SDSS does not. Furthermore, the labelling process of the images from the Galaxy Zoo project is done by thresholding a confidence value derived from public voting statistics. When more rigorous filtering is applied to this labelling process, the size of the dataset decreases significantly and leads to an unbalanced distribution of images for certain labels. This trade-off between label quality and dataset size motivates the application of self-supervised learning as it enables us to use larger amounts of data through the inclusion of unlabelled data. By leveraging larger amounts of unlabelled data, we can improve on model performance as it removes the need for labelled datasets to identify class-specific features of galaxies. As such, we can improve on feature representation learning whilst being invariant to label quality.

In this paper, we aim to develop an efficient deep-learning model that learns robust representations in a self-supervised manner, which generalise well across different astronomy surveys.

We construct a feature extraction module which learns lower-dimensional representation and forwards this to a dense neural network for classification. The two feature extraction architectures we consider are SimCLR and Autoencoders. The SimCLR architecture learns feature representations by maximizing the similarity of two images which have different augmentations applied to them. The augmentations we consider are Gaussian Noise, Crop & Resize, ColorJitter, Gray, Gaussian Blur and Random Rotation. The Autoencoder architecture learns feature representations by reconstructing the original image from a lower-dimensional representation.

We demonstrate the effectiveness of SimCLR and Autoencoders based on generalization across other datasets and compare its performance to supervised learning models. We assess the effectiveness of different image augmentations for SimCLR by their performance in linear evaluation and fine-tuning accuracy on Galaxy10 SDSS. We found a combination of simple but effective data augmentations for the classification of astronomy images. From our results, Gaussian Noise with double Crop & Resize is the most effective combination of data augmentations. Our SimCLR architecture achieves an overall accuracy of 89.12% on Galaxy10 SDSS and 87.49% on Galaxy10 DECals which is an improvement of 9% over the ResNet-18 architecture on both datasets. Our Autoencoder architecture achieves an overall accuracy of 90.79% on Galaxy10 SDSS and 88.75% on Galaxy10 DECals which is a 10% and 11% improvement over the ResNet-18 architecture respectively. We also investigate model performance across the two imaging surveys to assess the robustness of our pre-trained models.

2. Data set and task

The dataset used in this paper is a combined dataset; the image data was obtained from the Sloan Digital Sky Survey (SDSS), while the data labels were obtained from the Galaxy Zoo project, as well as Galaxy10 which is a rigorously filtered subset of Galaxy Zoo project.

2.1. Galaxy Zoo Project

The Galaxy Zoo project is a citizen science project which provides the morphological classification of galaxy images from the SDSS dataset(Willett et al., 2013). The project largely speeds up the morphological classification of the huge amount of existing galaxy images from SDSS. However, the public volunteering characteristic also makes the labels for each galaxy highly messy and unreliable, which makes the deep learning models hard to achieve a promising result on the labels provided by Galaxy Zoo.

2.2. SDSS Dataset

Sloan Digital Sky Survey (SDSS) is a survey aimed to produce images (indicate shape) and spectrums (indicate distance) of the galaxies over a large area of the sky by the 2.5m f/5 telescope from the Apache Point Observatory, New Mexico (York et al., 2000). The survey started from the year 2000, until 2012, the survey had 9 data releases which covers over 35% of the sky(Anderson et al., 2012). In this paper, we're using a subset of the SDSS dataset picked by Galaxy Zoo, which contains 243,434 images of the brightest galaxies in the SDSS dataset.

2.2.1. Galaxy10 SDSS Dataset

Galaxy10 SDSS dataset is a subset of the Galaxy Zoo project. The dataset only selects the galaxy images with more than 55% of the votes agreeing on the same classes(Ast). The dataset contains 21758 coloured galaxy images which have a size of 69×69 pixels. Examples of the data are shown in Figure 8. The class distribution is shown in Table 1.

Class	Name	Numbers
Class 0	Disk, Face-on, No Spiral	3461
Class 1	Smooth, Completely round	6997
Class 2	Smooth, in-between round	6292
Class 3	Smooth, Cigar shaped	394
Class 4	Disk, Edge-on, Rounded Bulge	1534
Class 5	Disk, Edge-on, Boxy Bulge	17
Class 6	Disk, Edge-on, No Bulge	589
Class 7	Disk, Face-on, Tight Spiral	1121
Class 8	Disk, Face-on, Medium Spiral	906
Class 9	Disk, Face-on, Loose Spiral	519

Table 1. Galaxy10 SDSS Dataset Class Distribution.

2.3. DECals Dataset

Dark Energy Camera Legacy Survey (DECals) is another survey that aims to produce detailed sky images in multiple optical bands, the survey uses the 4-meter dark energy

camera located at Cerro Tololo Inter-American Observatory, which is the most efficient observer for wide-field astronomy surveys(Dey et al., 2019). The survey started in 2014, and it had 10 data releases until 2023. Compared with SDSS, DECals had a higher resolution, and higher sensitivity with wider observable wavelength and thus could provide more morphological details.

2.3.1. Galaxy10 DECals Dataset

Galaxy10 DECals is a combined dataset, where it uses the labels from the Galaxy Zoo 2 dataset, which applied a more rigorous filter and replaces the image from SDSS with images with higher resolution gathered from DECals. Compared with Galaxy10 SDSS dataset, Galaxy10 DECals dataset contains more distinct classes and removes class 5 from the SDSS dataset which only had 17 instances. The class distribution of the Galaxy10 DECals dataset is shown in Table 2, and example images of each class are shown in figure7.

Class	Name	Numbers
Class 0	Disturbed Galaxies	1081
Class 1	Merging Galaxies	1853
Class 2	Round Smooth Galaxies	2645
Class 3	In-between Round Smooth Galaxies	2027
Class 4	Cigar Shaped Smooth Galaxies	334
Class 5	Barred Spiral Galaxies	2043
Class 6	Unbarred Tight Spiral Galaxies	1829
Class 7	Unbarred Loose Spiral Galaxies	2628
Class 8	Edge-on Galaxies without Bulge	1423
Class 9	Edge-on Galaxies with Bulge	1873

Table 2. Galaxy10 DECals Dataset Class Distribution.

2.4. Data Preprocessing

For our pre-training dataset, we centre-cropped every image from the SDSS unlabeled dataset, which contains 243434 images, from 424×424 pixels to 256×256 pixels to only keep the area of interest of the image since the galaxy is located at the centre of each image. We then resample the images into 128×128 pixels by using the cv2.resize() function from OpenCV. After that, we sampled 51000 images uniformly from the unlabeled dataset. These two steps are carried out to deal with the limited memory of our experiment setup.

2.5. Data Split

In this paper, we will use the 51000 rigorous filtered labelled data from the Galaxy10 SDSS dataset and 17736 labelled data from Galaxy10 DECals dataset as the labelled datasets.

To simulate the condition with an extremely limited amount of data, we downsampled each dataset to smaller subsampled versions, each containing 300, 30 and 3 images per class to explore the performance of the pre-trained model with a limited amount of labelled data, i.e., for Galaxy10 SDSS, Galaxy10 DECals, we create 3 other subsampled versions with 3000, 300 and 30 images are included.

To be noted, for the class in with only 17 images in the Galaxy10 SDSS dataset, we applied data augmentation to generate new data so the class could have 300 and 30 images in the subsampled versions.

When training with a labelled dataset, the dataset was split into training, validation and testing datasets with a ratio of 70:10:20.

3. Methodology

3.1. Convolutional Neural Network

A Convolutional Neural Network (CNN) is a distinct variant of an Artificial Neural Network (ANN). In contrast to conventional ANNs, CNNs incorporate "convolutional layers" that execute convolution operations on the input data. These layers use a weight matrix that moves across the input, applying consistent weights at various locations through convolution. Typically, the convolution process is followed by average pooling and nonlinear activation. As the kernel moves through the input data, feature and pattern data are extracted.

CNNs are widely used in image processing, as they can effectively compress high-dimensional images to extract features without losing meaningful information (Sharma et al., 2018). This is different from traditional ANNs, which face difficulties when handling high-dimensional data, to a massive number of trainable parameters.

3.1.1. ResNet

ResNet, short for Residual Network, is a CNN-based architecture proposed in (He et al., 2016). ResNet is designed to mitigate the vanishing gradient problem, which is a common problem that happens in the training of the deep neural network. The vanishing gradient problem will slow down the training process and harm the performance of the network.

The main difference between a ResNet and a normal deep neural network is the residual connection, which is an identity connection that connects the input of each residual block to the output of the residual block, as shown in Figure 1.

In our experiments, we applied a relatively small ResNet-18 architecture because not only is it commonly used in self-supervised learning, but also unfortunately due to the limited computational resources we had.

3.2. Self-supervised Learning

Self-supervised learning is a form of unsupervised learning in which the model learns representations from an unla-

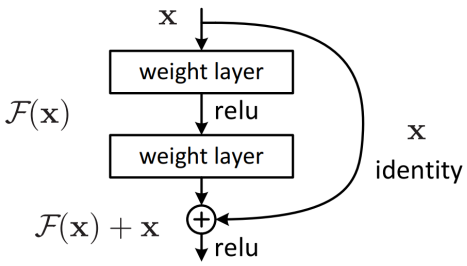

Figure 1. A ResNet building block. (Photo from (He et al., 2016))

beled dataset by completing the pretext task given during training time. Distinct from other unsupervised learning methods, pretext tasks generate a set of pseudo-labels. In the process of self-supervised learning, the model is trained to determine a mapping function F, which accurately associates each data instance x_i with the corresponding pseudo-label y_i produced by the pretext tasks. This is achieved by minimizing the loss function L, as defined below:

$$min \frac{1}{N} \sum_{i=0}^{N} L(f(x_i), y_i) \tag{1}$$

Self-supervised learning has proven to be an effective pre-training strategy for supervised tasks. Numerous self-supervised techniques currently are able to generate competitive representations in comparison to fully supervised models, demonstrating strong performance in various downstream applications (Ericsson et al., 2021).

In this paper, we will investigate the performance of two self-supervised learning architectures, namely SimCLR and Autoencoder, in the task of galaxy morphological classification.

3.2.1. SimCLR

SimCLR (Simple Contrastive Learning) is a contrastive learning framework first proposed in 2020 by (Chen et al., 2020). The SimCLR model learns to produce meaningful visual representations by maximizing the agreement between different augmented versions of the same picture. As shown in Figure 2, SimCLR consists of four components: a data augmentation scheme T, neural network encoder f, a small neural network projection head g and a contrastive loss function L.

The data augmentation scheme T will produce two different augmented views x_i and x_j for every data x provided. x_i and x_j will be considered as a positive pair. In our implementation, we applied the data augmentation of Gaussian Noise with double crop and resize, as our experiments in 4.1 show this data augmentation combination is crucial to achieving good performance.

The encoder f is a neural network that extracts a representation h from the augmented data view x. The SimCLR architecture provides freedom to the design of the encoder

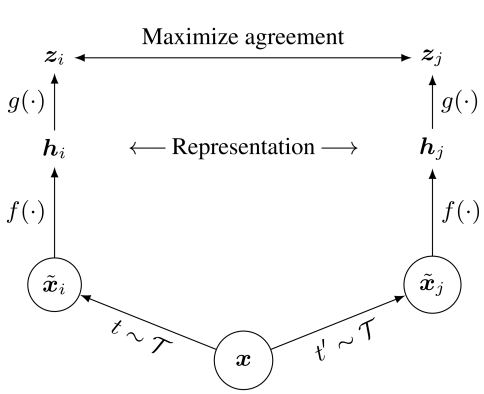

Figure 2. An overview of the SimCLR framework, photo from (Chen et al., 2020).

network. In our implementation, we applied the ResNet architecture proposed by (He et al., 2016), more specifically the ResNet-18 architecture.

The projection head g projects the visual representation extracted by encoder f to the space where contrastive loss is applied. Our implementation of the project head follows the implementation in (Chen et al., 2020), which is an MLP with one hidden layer and ReLu as the activation function.

For the contrastive loss function L, we applied an NT-Xent (the normalised temperature-scaled cross entropy loss) shown as follows.

$$\text{sim}(A, B) = \frac{A \cdot B}{||A||||B||} \tag{2}$$

$$l_{i,j} = -\log \frac{\exp(\text{sim}(z_i, z_j)/\tau)}{\sum_{k=1}^{2N} \mathbb{1}_{[k \neq i]} \exp(\text{sim}(z_i, z_k)/\tau)} \tag{3}$$

Where $\mathbb{1}_{[k \neq i]} \in \{0, 1\}$ returns 1 iff $k \neq i$, and τ indicate the temperature constant.

The NT-Xent loss will award the high similarity pair between positive pairs (the two augmented views generated from the same data) in the batch and penalise high similarities between negative pairs (all other augmented views generated from different data in the batch).

3.2.2. Autoencoder

Autoencoder is another popular type of self-supervised representation learner. The Autoencoder model learns to produce a meaningful visual representation by minimising the difference between input data y_i and reconstructed data $\hat{y}_i$. An Autoencoder has three major components, as shown in Figure 3:

A neural network base encoder f that extracts a visual representation h_i from the data y_i provided. In our implementation, we adopt the common used ResNet (He et al., 2016), more specifically we selected a ResNet-18 architecture due to the limitation of computation resources.

A neural network base decoder g project the representa-

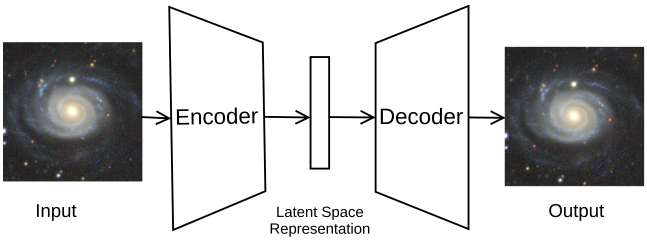

Figure 3. An overview of the Autoencoder framework

tion h_i back to the data space. In our implementation, we adopted a symmetric Autoencoder design, our decoder also used ResNet-18 architecture.

A loss function L, which compare the difference between data y_i and reconstructed data $\hat{y}_i$. We applied a common used mean squared error shown is follows.

$$\text{MSE}(y, \hat{y}) = \frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y}_i)^2 \tag{4}$$

3.3. Data Augmentation

Data augmentation is a strategy that increases the dataset's diversity by applying a range of transformations to the existing data, generating slightly modified versions, and then adding them to the dataset. Data augmentation could effectively reduce the overfitting of the model during training (Shorten & Khoshgoftaar, 2019).

In our experiments, we applied data augmentation to our training dataset to increase the dataset size. We also used data augmentation in training the SimCLR model to generate two different augmented views which are used to generate pseudo labels.

We tested six popular data augmentation in computer vision, including crop and resize, rotation, colour jittering, gaussian blur, gaussian noise and grey scale.

3.4. Performance Metrics: Macro-F1

The macro-F1 scores are recorded and plotted the figures across the report. We chose to compare macro-F1 scores across the different sizes of datasets over other scores values because they allowed us to evaluate the performance of the model on the imbalanced classes in the dataset. Since the class with the highest number of images can dominate the accuracy of the model, using the macro-F1 scores, each class contributes equally to the final score, irrespective of its size. This meant that macro-F1 was less sensitive to class imbalance and provided us a better understanding of the how well the models is performing on all classes. The formula of Macro F1 is shown as follows.

$$\text{F1} = 2 \times \frac{\text{TP}}{2\text{TP} + \text{FP} + \text{FN}} \tag{5}$$

4. Experiments

4.0.1. Experimental settings

The experiments are conducted on the Edinburgh University teaching cluster with NVIDIA GTX1060 6GB and 16 GB RAM. All of our experiments are implemented using PyTorch Lightning (Falcon et al., 2019).

Unless otherwise specified, the models are pretrained using the settings in the Table 3, fine-tuned and linearly evaluated using the settings shown in Table 4.

Config	Value
Optimizer	Adam (Kingma & Ba, 2014)
Base Learning Rate	0.001
EarlyStopping Min_delta	0.002
EarlyStopping Patience	10 epochs
EarlyStopping Criteria	NTXent Loss
Maximum Epoch	60
Minimum Epoch	5
Batch Size	8

Table 3. Experimental settings used to run the pretrainings for our two proposed models.

Config	Value
Optimizer	Adam
Base Learning Rate	0.01
Learning Rate Schedule	cosine decay (Loshchilov & Hutter, 2016)
EarlyStopping Min_delta	0.0001
EarlyStopping Patience	10 epochs
EarlyStopping Criteria	Macro F1
Maximum Epoch	200
Minimum Epoch	5
Batch Size	32

Table 4. Experimental settings used to run linear evaluation and fine tuning for our two proposed models.

4.0.2. Data Augmentation Details

All of the data augmentation used in the experiments is implemented using *torchvision.transforms* package aside from Gaussian Noise which was implemented using PyTorch. The value of the parameters for each data augmentation is shown in Table 5, and all other parameters not specified were set to default values. Each data augmentation will have a 50% chance of not being applied in order to increase the variety and randomness in training.

Type of Augmentation	Parameter	Value
ColorJitter	brightness	0.5
	contrast	0.5
	saturation	0.5
	hue	0.1
Crop+Resize	size	96
GaussianBlur	kernel_size	9
GaussianNoise	mean	0
	std	0.05
Gray	probability	0.2
RandomRotation	degrees	(0,360)

Table 5. Parameters and values used for the different data augmentation techniques in the experiments carried out throughout the research.

4.1. Choosing the data augmentation

We first performed experiments on the six data augmentation mentioned previously, using the parameters described in Table 5. The results obtained are shown in Figure 4.

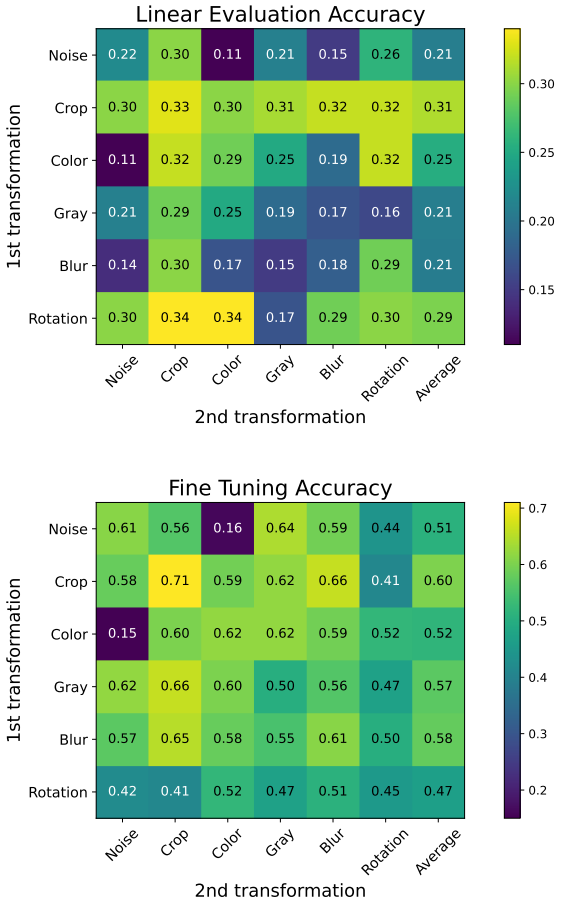

Figure 4. The first heatmap shows the linear evaluation accuracies while the second heatmap shows the fine tuning accuracies. The accuracies are obtained when the first transformation is applied, followed by the second transformation.

From Figure 4, we observed that applying Crop & Resize for both the first and second transformations (hence applied twice) led to an overall best performance in fine tuning, achieving an accuracy of 0.71, while achieving one of the highest in linear evaluation (accuracy=0.33). Additionally, we observed that Crop & Resize transformed achieved the best average linear evaluation (accuracy=0.31) and fine tuning (accuracy=0.60) accuracies.

Since applying Crop & Resize twice are both geometrical transformations on an image, we ran another set of experiments to pick a color space transformations on the images: Gaussian Blur, Color Jittering, Gray and Gaussian Noise. We conducted two separate experiments where in one, the transformation was applied before double Cropping, and in the second one, the transformation was applied after double Cropping. These experiments are run using the same parameters as described in Table 5, and the results obtained are displayed in Table 6.

Evaluation Accuracies with Double Crop&Resize				
Type of Augmentation	Linear Evaluation		Fine Tuning	
	Before	After	Before	After
ColorJitter	0.310	0.339	0.592	0.642
GaussianBlur	0.371	0.387	**0.671**	0.659
GaussianNoise	**0.409**	0.378	0.663	0.613
Gray	0.347	0.370	0.639	0.588

Table 6. The linear evaluation and fine tuning results obtained when applying the data augmentation techniques followed by applying Crop&Resize twice.

From Table 6, we observed that applying Gaussian Noise followed by Crop & Resize applied twice, yielded to the highest overall linear evaluation results (accuracy=0.409), while applying Gaussian Blur followed by double Crop & Resize resulted in the highest fine tuning accuracy (accuracy=0.671). However, the difference between applying Gaussian Noise first and applying Gaussian Blur before double cropping (difference in accuracies is 0.038) outweighed the difference between applying Gaussian Blur first and applying Gaussian Noise first (difference in accuracies is 0.008). Hence, the combination of Gaussian Noise, followed by double Crop & Resize is chosen as data augmentations for future experiments.

4.2. Performance on Galaxy10 SDSS

The proposed models, Autoencoder and SimCLR, were first pretrained on the unlabelled pre-processed SDSS dataset and then fine-tuned using the 4 different sizes of the labelled Galaxy10 SDSS dataset (30 images, 300 images, 3000 images and 21785 images (whole dataset)). The metric used for evaluating the performance of the 4 models is the macro-F1 score, and the choice for using it over other scores is explained in Section 3.4.

The first experiment conducted consisted of 30 images (3 images per class) of the balanced dataset. We observed that SimCLR model performed better than both the baseline model and the Autoencoder by a score of 0.082 and 0.145 respectively, and that the baseline outperformed the proposed Autoencoder model by a score 0.063.

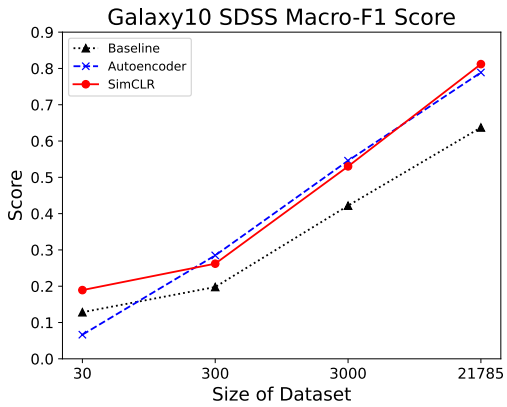

Figure 5. The graph plots the macro-F1 scores obtained when applying the 3 different proposed models; baseline ResNet18, Autoencoder, and SimCLR, to the different sizes; 30 images (3 images per class), 300 images (30 images per class), 3000 images (300 images per class) and 21785 images of Galaxy10 SDSS dataset.

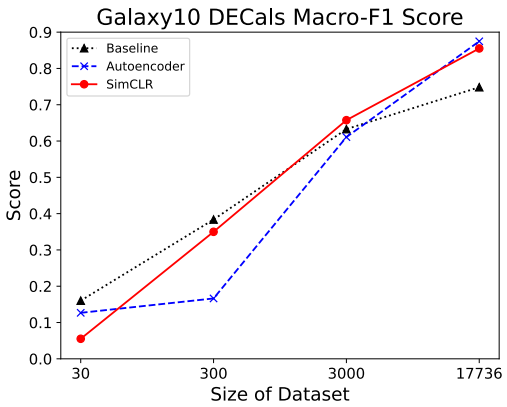

Figure 6. The graph plots the macro-F1 scores obtained when applying the 3 different models; baseline ResNet-18, Autoencoder, and SimCLR, to different sizes of the dataset; 30 images (3 images per class), 300 images (30 images per class), 3000 images (300 images per class) and 17736 images of the Galaxy10 DECals dataset.

In the second experiment, we increased the number of images to 300 (30 images per class) and we immediately observed a significant improvement in the macro-F1 score when using the Autoencoder model (achieved the highest macro-F1 score of 0.285). The SimCLR model performed just slightly worse than the Autoencoder (achieved second highest macro-F1 score of 0.262) while the baseline achieved only a score of 0.198.

In the third experiment, both proposed models performed better on 3000 images (300 images per class) than the baseline model, achieving a macro-F1 score of 0.611, 0.530 and 0.413 for Autoencoder, SimCLR and the ResNet-18 baseline respectively.

Finally, for the fully unbalanced labelled dataset of Galaxy10 SDSS (21785 images), SimCLR and Autoencoder achieved a macro-F1 score of 0.812 and 0.789 respectively, while the baseline ResNet-18 achieved a score of 0.637 only.

These experiments showed that the proposed models have outperformed the baseline model.

The other performance metrics are presented in Table 7.

4.3. Performance Galaxy10 DECals

The proposed models, Autoencoder and SimCLR, were first pretrained on the unlabelled pre-processed SDSS dataset and then fine-tuned using the four different sizes, similar to the Galaxy10 SDSS dataset of the labelled Galaxy10 DECals dataset. Once again, the metric used for evaluating the performance of the 4 models is the macro-F1 score, and the choice for using it over other scores is explained later in Section 3.4.

Similarly to the experiments with Galaxy10 SDSS, the first experiment consisted of fine tuning the model using 30 images (3 images per class) of the balanced Galaxy10 DECals dataset. We observed that the baseline ResNet-18 model, which achieved macro-F1 score = 0.160, outperformed both proposed models by 0.032 with the Autoencoder model, and 0.105 with the SimCLR model.

In the second experiment, we used 300 images of the balanced DECals dataset and observed a significant increase in the performance of SimCLR (macro-F1 score = 0.350), but still performed slightly worst than the baseline model, which achieved a macro-F1 score = 0.384, by 0.034. On the other side, the proposed Autoencoder model performed relatively bad compared to both the baseline and the proposed SimCLR model, only achieving 0.166.

In the third experiment, we found that fine-tuning using 3000 images (300 images per class) of the balanced Galaxy10 DECals dataset achieved relatively close scores across all 3 models. (Baseline ResNet-18 achieved a score of 0.633, Autoencoder achieved a score of 0.611 and SimCLR achieved a score of 0.657). From Figure 6, we can also see that the SimCLR model performed slightly better than the baseline model, which performed slightly better than the Autoencoder.

Finally, in the last experiment where 17736 images were used, SimCLR and Autoencoder achieved a macro-F1 score of 0.855 and 0.875 respectively, while the baseline ResNet-18 achieved a macro-F1 score of 0.748.

The other performance accuracies are presented in Table 8.

4.4. Performance of Galaxy10 SDSS v/s Galaxy10 DECals

From Figure 5 and Figure 6, we observed that the proposed models performed better when fine-tuned on the Galaxy10 DECals dataset than on the Galaxy10 SDSS dataset by a score of 0.086 in Autoencoder and a score of 0.043 in SimCLR. This could be due to the images from DECals

being of much higher resolution since they were 256×256 pixels while SDSS was only 69×69 pixels. Furthermore, we also observed that the performance of both the Autoencoder and SimCLR were slightly worst than the baseline ResNet-18 on Galaxy10 DECals for 30 images, while SimCLR performed slightly better than the baseline model in Galaxy10 SDSS. In the case of 300 images, we observed that both proposed models outperformed the baseline on the Galaxy10 SDSS dataset, while SimCLR performed slightly worst than the baseline and Autoencoder performed poorly, both on the other dataset. These outcomes could be because the model was initially pre-trained using the unlabelled Galaxy10 SDSS dataset, which was a different survey dataset to Galaxy10 DECals. Using the full datasets on both models saw a larger increase in performance, where the models eventually outperformed the baseline model. Hence, we can see that our models are able to generalise well with different survey datasets.

5. Related Work

The recent advancement in deep learning and the success of the Galaxy Zoo Citizen Science Project revolutionised the morphological classification of galaxies (Lintott et al., 2008). The contributions from researchers have demonstrated the capability of supervised deep learning-based approaches in the task of galaxy morphology classification (Dieleman et al., 2015; Kim & Brunner, 2016; Gharat & Dandawate, 2022). While CNN based model remains the mainstream model for galaxy classification, (Lin et al., 2021) proposed the first vision transformer-based model applied to galaxy classification. While that model failed to outperform the ResNet-18 baseline, (Lin et al., 2021) found vision transformer-based model performed significantly better in classifying smaller and fainter galaxies.

While all aforementioned papers are focusing on different galaxy morphology classification tasks using different datasets, there are also several researchers specifically focusing on Galaxy10 SDSS and Galaxy10 DECals datasets, which are the datasets we used in the report. (Gharat & Dandawate, 2022) applied a VGG08 architecture to classify Galaxy10 SDSS dataset, and achieved an accuracy of 84%. (Ghadekar et al., 2023) proposed a CNN-based model and applied it to the Galaxy10 DECals dataset, achieving an accuracy of 84.04%.

In terms of self-supervised representation learning, (Hayat et al., 2021) proposed a CNN-based self-supervised architecture to learn representations from astronomical images. The pre-trained model was then fine-tuned on label data and applied to the three-class galaxy morphology classification. The model successfully outperformed the supervised baseline. However, the three-way classification was too trivial that the supervised baseline would already have a good performance. In two among three classes, the self-supervised model only outperforms the supervised baseline by 3% (from 95% to 98%) and 0% (100% to 100%). Furthermore, only images from Galaxy10 SDSS are used, the question of the ability to generalise well on different survey images remains to be investigated.

6. Conclusions and Future Directions

We have built two self-supervised models that classify galaxies according to their morphology, achieving a macro-F1 score of 0.812 on the Galaxy10 SDSS dataset. Additionally, both proposed models have outperformed the supervised ResNet-18 baseline by 17%. We have also investigated the fine-tuning performance of the pre-trained models on galaxy images from a distinct astronomy survey, Galaxy10 DECals, to test the cross-survey generalisation ability. Once again, the models outperformed the supervised ResNet-18 baseline by 10%. We found that the boost in performance brought by the self-supervised pretraining decreased when fine-tuning the model on the images from Galaxy10 DECals. The pre-trained models did not outperform the baseline when a small amount of data is used for fine-tuning, but remained effective and even provided a boost in performance of around 12% when the whole dataset is used for fine-tuning.

With the above summary of results, as shown in Figure 5 and Figure 6, we can conclude that the models learn robust feature representations for astronomy images in a self-supervised manner while generalising well across different astronomy surveys.

One future direction we will like to conduct is applying vision transformer-based self-supervised learning models to galaxy morphology classification. The current state-of-the-art vision transformer-based self-supervised learning model can outperform the CNN-based self-supervised learning because of the introduction of an attention mechanism in the transformer model (Caron et al., 2021). Another future direction we will like to try is masked Autoencoder instead of the normal Autoencoder. Proposed by (He et al., 2022), Mask Autoencoder is an effective vision-transformer-based Autoencoder architecture which has shown promising results in ImageNet classification tasks. We will like to investigate if these two state-of-the-art self-supervised models could achieve better performance on galaxy morphology classification.

7. Code and Data Availability

The code we used to conduct the experiments is available at: https://github.com/JunhuaL/MLP$_G$036. The data we used are available on https://astronn.readthedocs.io/en/latest/galaxy10sdss.html and https://zenodo.org/record/3565489.Y3vFKS-l0eY.

References

Galaxy10 SDSS Dataset. https://astronn.readthedocs.io/en/latest/galaxy10sdss.html, accessed 26 January 2023.

Anderson, Lauren, Aubourg, Eric, Bailey, Stephen, Bizyaev, Dmitry, Blanton, Michael, Bolton, Adam S, Brinkmann,

Jon, Brownstein, Joel R, Burden, Angela, Cuesta, Antonio J, et al. The clustering of galaxies in the sdss-iii baryon oscillation spectroscopic survey: baryon acoustic oscillations in the data release 9 spectroscopic galaxy sample. *Monthly Notices of the Royal Astronomical Society*, 427(4):3435–3467, 2012.

Barchi, P.H., de Carvalho, R.R., Rosa, R.R., Sautter, R.A., Soares-Santos, M., Marques, B.A.D., Clua, E., Gonçalves, T.S., de Sá-Freitas, C., and Moura, T.C. Machine and deep learning applied to galaxy morphology - a comparative study. *Astronomy and Computing*, 30:100334, 2020. ISSN 2213-1337. doi: https://doi.org/10.1016/j.ascom.2019.100334. URL https://www.sciencedirect.com/science/article/pii/S2213133719300757.

Caron, Mathilde, Touvron, Hugo, Misra, Ishan, Jégou, Hervé, Mairal, Julien, Bojanowski, Piotr, and Joulin, Armand. Emerging properties in self-supervised vision transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 9650–9660, 2021.

Cavanagh, Mitchell K, Bekki, Kenji, and Groves, Brent A. Morphological classification of galaxies with deep learning: comparing 3-way and 4-way cnns. *Monthly Notices of the Royal Astronomical Society*, 506(1):659–676, 2021.

Chen, Ting, Kornblith, Simon, Norouzi, Mohammad, and Hinton, Geoffrey. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pp. 1597–1607. PMLR, 2020.

Conselice, Christopher J. The evolution of galaxy structure over cosmic time. *Annual Review of Astronomy and Astrophysics*, 52(1):291–337, 2014. doi: 10.1146/annurev-astro-081913-040037. URL https://doi.org/10.1146/annurev-astro-081913-040037.

Dey, Arjun, Schlegel, David J, Lang, Dustin, Blum, Robert, Burleigh, Kaylan, Fan, Xiaohui, Findlay, Joseph R, Finkbeiner, Doug, Herrera, David, Juneau, Stéphanie, et al. Overview of the desi legacy imaging surveys. *The Astronomical Journal*, 157(5):168, 2019.

Dieleman, Sander, Willett, Kyle W, and Dambre, Joni. Rotation-invariant convolutional neural networks for galaxy morphology prediction. *Monthly notices of the royal astronomical society*, 450(2):1441–1459, 2015.

Ericsson, Linus, Gouk, Henry, and Hospedales, Timothy M. How well do self-supervised models transfer? In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5414–5423, June 2021.

Falcon et al., William. Pytorch lightning. *GitHub. Note: https://github.com/PyTorchLightning/pytorch-lightning*, 3, 2019.

Feigelson, Eric D and Babu, GJ. Statistical challenges in modern astronomy. *arXiv preprint astro-ph/0401404*, 2004.

Ghadekar, Premanand, Chanda, Kunal, Manmode, Sakshi, Rawate, Sanika, Chaudhary, Shivam, and Suryawanshi, Resham. Galaxy classification using deep learning. In *Advancements in Smart Computing and Information Security: First International Conference, ASCIS 2022, Rajkot, India, November 24–26, 2022, Revised Selected Papers, Part I*, pp. 3–13. Springer, 2023.

Gharat, Sarvesh and Dandawate, Yogesh. Galaxy classification: a deep learning approach for classifying sloan digital sky survey images. *Monthly Notices of the Royal Astronomical Society*, 511(4):5120–5124, 2022.

Hayat, Md Abul, Stein, George, Harrington, Peter, Lukić, Zarija, and Mustafa, Mustafa. Self-supervised representation learning for astronomical images. *The Astrophysical Journal Letters*, 911(2):L33, 2021.

He, Kaiming, Zhang, Xiangyu, Ren, Shaoqing, and Sun, Jian. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778, 2016.

He, Kaiming, Chen, Xinlei, Xie, Saining, Li, Yanghao, Dollár, Piotr, and Girshick, Ross. Masked autoencoders are scalable vision learners. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 16000–16009, 2022.

Kalvankar, Shreyas, Pandit, Hrushikesh, and Parwate, Pranav. Galaxy morphology classification using efficientnet architectures, 2020. URL https://arxiv.org/abs/2008.13611.

Kim, Edward J and Brunner, Robert J. Star-galaxy classification using deep convolutional neural networks. *Monthly Notices of the Royal Astronomical Society*, pp. stw2672, 2016.

Kingma, Diederik P and Ba, Jimmy. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

Lin, Joshua Yao-Yu, Liao, Song-Mao, Huang, Hung-Jin, Kuo, Wei-Ting, and Ou, Olivia Hsuan-Min. Galaxy morphological classification with efficient vision transformer. *arXiv preprint arXiv:2110.01024*, 2021.

Lintott, Chris J, Schawinski, Kevin, Slosar, Anže, Land, Kate, Bamford, Steven, Thomas, Daniel, Raddick, M Jordan, Nichol, Robert C, Szalay, Alex, Andreescu, Dan, et al. Galaxy zoo: morphologies derived from visual inspection of galaxies from the sloan digital sky survey. *Monthly Notices of the Royal Astronomical Society*, 389 (3):1179–1189, 2008.

Loshchilov, Ilya and Hutter, Frank. Sgdr: Stochastic gradient descent with warm restarts. *arXiv preprint arXiv:1608.03983*, 2016.

Sharma, Neha, Jain, Vibhor, and Mishra, Anju. An analysis of convolutional neural networks for image classification. *Procedia computer science*, 132:377–384, 2018.

Shorten, Connor and Khoshgoftaar, Taghi M. A survey on image data augmentation for deep learning. *Journal of big data*, 6(1):1–48, 2019.

Willett, Kyle W, Lintott, Chris J, Bamford, Steven P, Masters, Karen L, Simmons, Brooke D, Casteels, Kevin RV, Edmondson, Edward M, Fortson, Lucy F, Kaviraj, Sugata, Keel, William C, et al. Galaxy zoo 2: detailed morphological classifications for 304 122 galaxies from the sloan digital sky survey. *Monthly Notices of the Royal Astronomical Society*, 435(4):2835–2860, 2013.

York, Donald G, Adelman, J, Anderson Jr, John E, Anderson, Scott F, Annis, James, Bahcall, Neta A, Bakken, JA, Barkhouser, Robert, Bastian, Steven, Berman, Eileen, et al. The sloan digital sky survey: Technical summary. *The Astronomical Journal*, 120(3):1579, 2000.

Zhu, Xiao-Pan, Dai, Jia-Ming, Bian, Chun-Jiang, Chen, Yu, Chen, Shi, and Hu, Chen. Galaxy morphology classification with deep convolutional neural networks. *Astrophysics and Space Science*, 364(4), apr 2019. doi: 10.1007/s10509-019-3540-1. URL https://doi.org/10.1007%2Fs10509-019-3540-1.

8. Appendix

Model	Macro F1 Fine-tune on				Accuracies Fine-tune on			
	30	300	3000	21785	30	300	3000	21758
Baseline	0.129	0.198	0.422	0.637	0.192	0.255	0.517	0.801
SimCLR	0.189	0.262	0.530	0.812	0.211	0.372	0.676	0.891
Autoencoder	0.066	0.285	0.546	0.779	0.129	0.402	0.638	0.877

Table 7. Evaluation accuracies and macro F1 score for the different amounts of labelled data used for fine-tuning on Galaxy10 SDSS dataset.

Model	Macro F1 Fine-tune on				Accuracies Fine-tune on			
	30	300	3000	17736	30	300	3000	17736
Baseline	0.160	0.384	0.632	0.748	0.183	0.386	0.659	0.778
SimCLR	0.055	0.350	0.657	0.855	0.169	0.401	0.681	0.875
Autoencoder	0.126	0.166	0.611	0.845	0.169	0.233	0.632	0.856

Table 8. Evaluation accuracies and macro F1 score for the different amounts of labelled data used for fine-tuning on Galaxy10 DECals dataset.

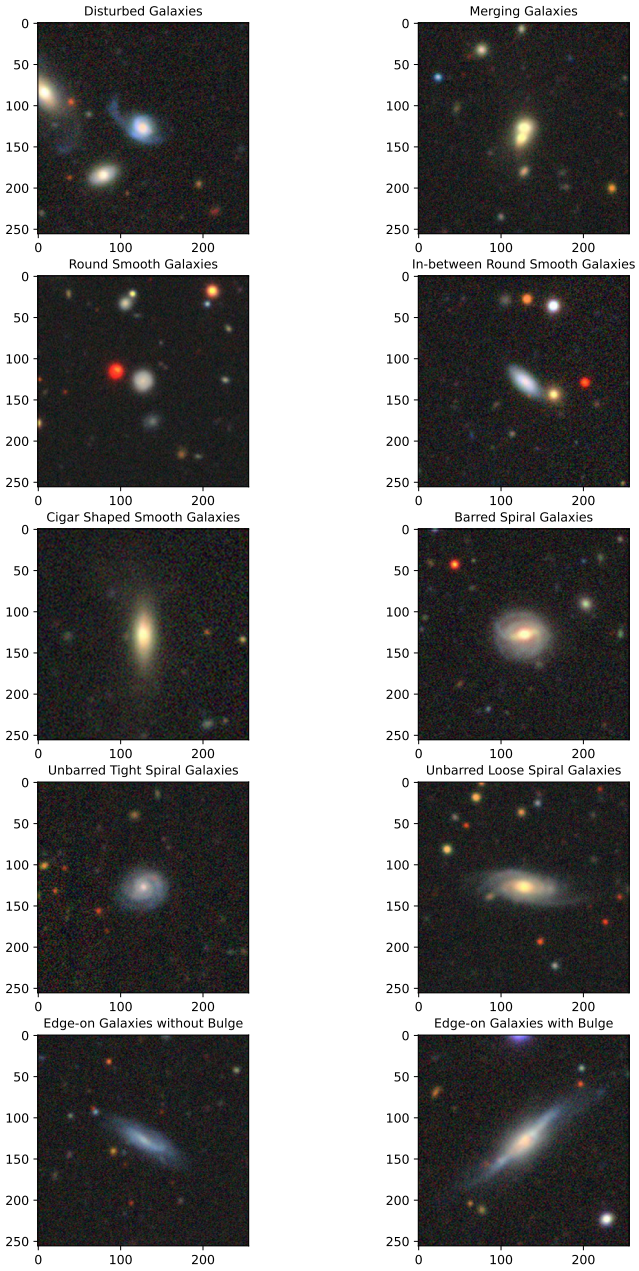

Figure 7. Images of each Galaxy Class in DECals

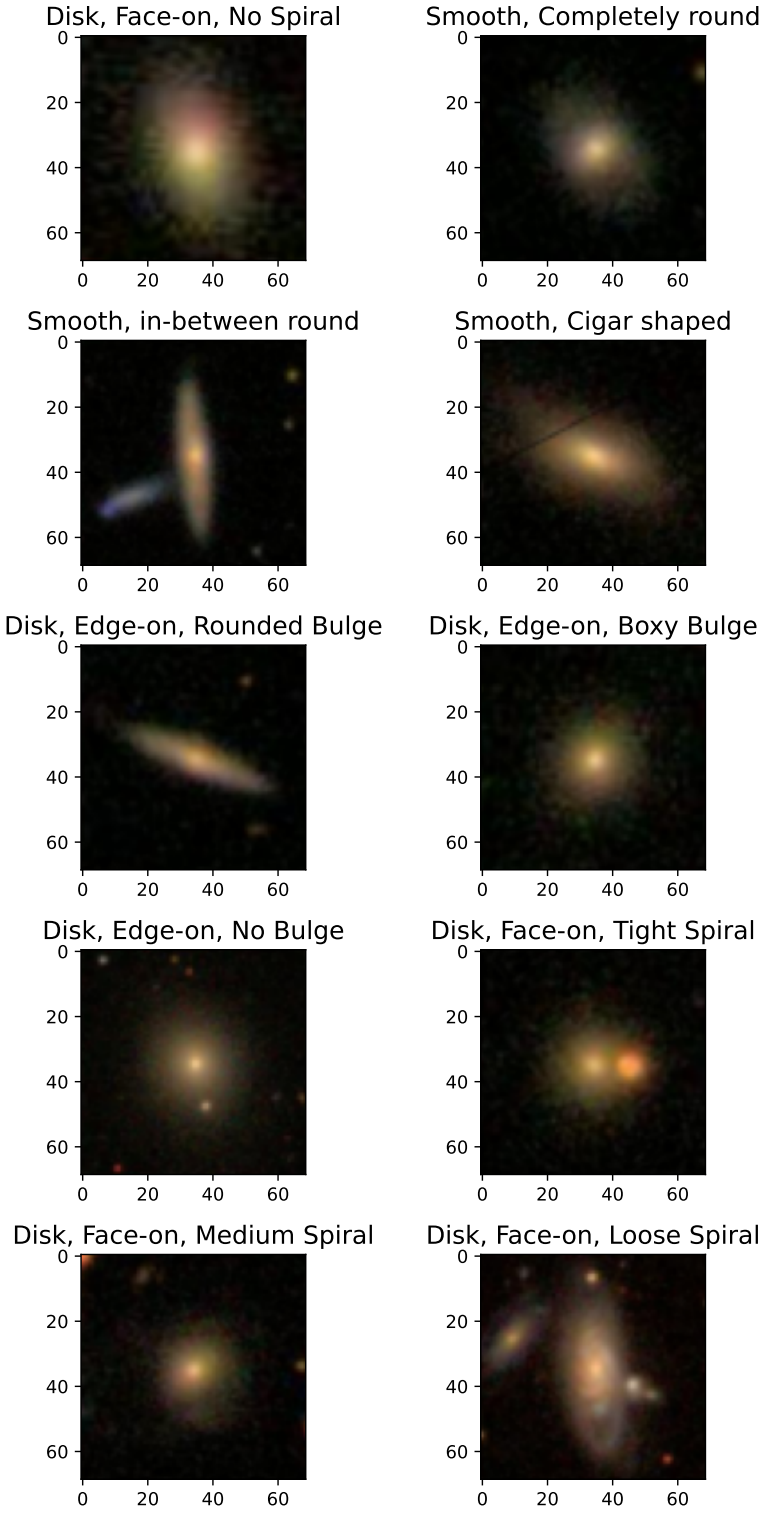

Figure 8. Images of each Galaxy Class