RAS Techniques and Instruments

RASTAI **2,** 264–277 (2023) https://doi.org/10.1093/rasti/rzad016
Advance Access publication 2023 May 26

Morphological classification of radio galaxies with Wasserstein generative adversarial network-supported augmentation

Lennart Rustige,[1,2] Janis Kummer ,[1,3★] Florian Griese ,[1,4,5] Kerstin Borras,[2,6] Marcus Brüggen ,[3] Patrick L. S. Connor,[1,7] Frank Gaede,[2] Gregor Kasieczka,[7] Tobias Knopp[4,5] and Peter Schleper[7]

[1]*Center for Data and Computing in Natural Sciences (CDCS), Notkestrasse 9, D-22607 Hamburg, Germany*
[2]*Deutsches Elektronen-Synchrotron DESY, Notkestrasse 85, D-22607 Hamburg, Germany*
[3]*Universität Hamburg, Hamburger Sternwarte, Gojenbergsweg 112, D-21029 Hamburg, Germany*
[4]*Section for Biomedical Imaging, University Medical Center Hamburg-Eppendorf, D-20246 Hamburg, Germany*
[5]*Institute for Biomedical Imaging, Hamburg University of Technology, D-21073 Hamburg, Germany*
[6]*Physics Institute III A, RWTH Aachen University, Templergraben 55, D-52062 Aachen, Germany*
[7]*Institut für Experimentalphysik, Universität Hamburg, Luruper Chaussee 149, D-22761 Hamburg, Germany*

Accepted 2023 May 19. Received 2023 April 25; in original form 2022 December 16

ABSTRACT

Machine learning techniques that perform morphological classification of astronomical sources often suffer from a scarcity of labelled training data. Here, we focus on the case of supervised deep learning models for the morphological classification of radio galaxies, which is particularly topical for the forthcoming large radio surveys. We demonstrate the use of generative models, specifically Wasserstein generative adversarial networks (wGANs), to generate data for different classes of radio galaxies. Further, we study the impact of augmenting the training data with images from our wGAN on three different classification architectures. We find that this technique makes it possible to improve models for the morphological classification of radio galaxies. A simple fully connected neural network benefits most from including generated images into the training set, with a considerable improvement of its classification accuracy. In addition, we find it is more difficult to improve complex classifiers. The classification performance of a convolutional neural network can be improved slightly. However, this is not the case for a vision transformer.

Key words: Machine Learning; – Data Methods – methods: data analysis – methods: statistical – radio continuum: galaxies – techniques: image processing.

1 INTRODUCTION

Radio galaxies are galaxies that emit a large fraction of their electromagnetic output in the radio band. The structures visible in radio wavelengths are typically larger than the structures visible in optical wavelengths. Radio galaxies are a class of active galactic nuclei (AGNs) and are powered by supermassive black holes at the centres of galaxies. The extended emission is produced by synchrotron radiation of highly relativistic particles accelerated by the AGN. Studying radio galaxies helps to understand the effects of massive black holes on their environment (see e.g. McNamara & Nulsen 2007). The jets of highly energetic particles emitted by giant radio galaxies potentially play a major role in the creation of cosmic magnetic fields (Vazza et al. 2023).

A lot of new radio sources will be discovered with the new generation of radio telescopes (e.g. *LOFAR*, *MeerKAT*, and in the future the *SKA*, Carilli et al. 2004; van Haarlem et al. 2013; Jonas & MeerKAT Team 2016). Processing the incoming data is one of the biggest challenges in radio astronomy. The cause is not only the enormous amount of data, but also the higher source density due to the improved sensitivity of the instruments. Novel techniques are required for this purpose. For instance, the *SKA* data challenges have demonstrated the difficulties of source finding for *SKA* data (Bonaldi et al. 2021). Deep learning has been used to automate processes in radio astronomical data reduction, e.g. in the automatic flagging of data (see e.g. Mosiane et al. 2017). Another example is the work by Mesarcik et al. (2020) who have used a variational autoencoder (VAE) in combination with other methods to automatically inspect data to diagnose system health for modern radio telescopes. Commonly large amounts of labelled training data are required for supervised algorithms, which are not always available.

Morphological classification of radio sources can be achieved by deep learning models trained on well-understood data sets. Alhassan et al. (2018), Aniyan & Thorat (2017), Maslej-Krešňáková et al. (2021), Samudre et al. (2021), and Tang et al. (2019) use convolutional neural networks (CNNs) trained on data from the Faint Images of the Radio Sky at Twenty-Centimeters (FIRST) survey (Becker et al. 1995) for the classification of radio galaxies. The architectures of the neural networks for classification are inspired by the AlexNet (Krizhevsky et al. 2012). For approaches in radio galaxy classification that use non-standard CNNs and other techniques, see e.g. Bowles et al. (2020), Lukic et al. (2019), Ma et al. (2019b),

★ E-mail: janis.kummer@desy.de

Ntwaetsile & Geach (2021), Sadeghi et al. (2021), Scaife & Porter (2021), and Wu et al. (2019).

In other areas of astronomy, similar morphological classification problems arise, e.g. for classification of optical galaxies (Lintott et al. 2008; Nair & Abraham 2010), and of gravitational lenses (Petrillo et al. 2017). Here, supervised methods of machine learning have been applied with some success, see e.g. Cheng et al. (2020), Huertas-Company & Lanusse (2023), Vavilova et al. (2021), and Walmsley et al. (2019).

However, the existing number of radio sources with morphological labels is limited [the MiraBest data set contains 1254 Fanaroff–Riley type I (FR I), FR II, and hybrid FR sources (Porter 2020)]. These class labels are typically extracted from catalogues created and curated manually by experts. Small data sets used in the training of deep learning models for galaxy classification can be enlarged by data augmentation (Maslej-Krešňáková et al. 2021), e.g. by applying random rotations and reflections to the images (classical augmentation). A different approach based on equivariance implements the symmetry constraints of the problem directly in the construction of the model (Bowles et al. 2021; Scaife & Porter 2021). This may help classifiers to understand symmetries without relying exclusively on augmentation and may be particularly useful for problems with sparse data.

In this work, we investigate a novel application of generative models to enhance the available training sets. For this augmentation technique, multiple neural networks are combined to learn the underlying distribution of a data set. We focus on the task of classifying different morphological types of radio galaxies. The morphological classification scheme by FR is fundamental for such applications (Fanaroff & Riley 1974). For the class FR I, the unique maximum of the radio emission resides in the centre of the source and the surface brightness decreases along the jets. For FR II sources, the two maxima of the radio emissions are located at the edges of the jets and the surface brightness in the centre is lower. As radio sources have a large variety of structures, we consider two more classes. Unresolved and point sources are contained in the compact class. The bent class consists of sources for which the angle between the jets differs significantly from 180 degrees. The two subtypes narrow-angle tail (NAT) and wide-angle tail (WAT) are further discriminated by the angle, but are fully subsumed in the bent class for this study. As in Alhassan et al. (2018) and Samudre et al. (2021), we study a four-class classification problem, including bent-tail and compact sources in addition to the classes FR I and FR II of Fanaroff & Riley (1974). Fig. 1 illustrates the considered classes (FR I, FR II, compact, and bent).

Other studies probe the use of generative models to create images of radio galaxies (Ma et al. 2018, 2019a; Bastien et al. 2021). These studies are based on VAEs. Generative adversarial networks (GANs) have been applied to astrophysical images in Schawinski et al. (2017). For a semisupervised GAN application to radio pulsars see Balakrishnan et al. (2021). In Hackstein et al. (2023), various evaluation metrics are used to compare different generative models trained on optical galaxy images.

In this study, we investigate whether different radio galaxy classifiers can be improved when training is supported by providing additional data generated with a Wasserstein generative adversarial network (wGAN). For similar approaches from different fields see e.g. Frid-Adar et al. (2018), Gowal et al. (2021), and Zhu et al. (2017). We extend our framework presented in Kummer et al. (2022) to handle larger ratios between real and generated images. Additional images are only generated when they are needed during training. As before, we start with a simple model, namely an FCN. In addition,

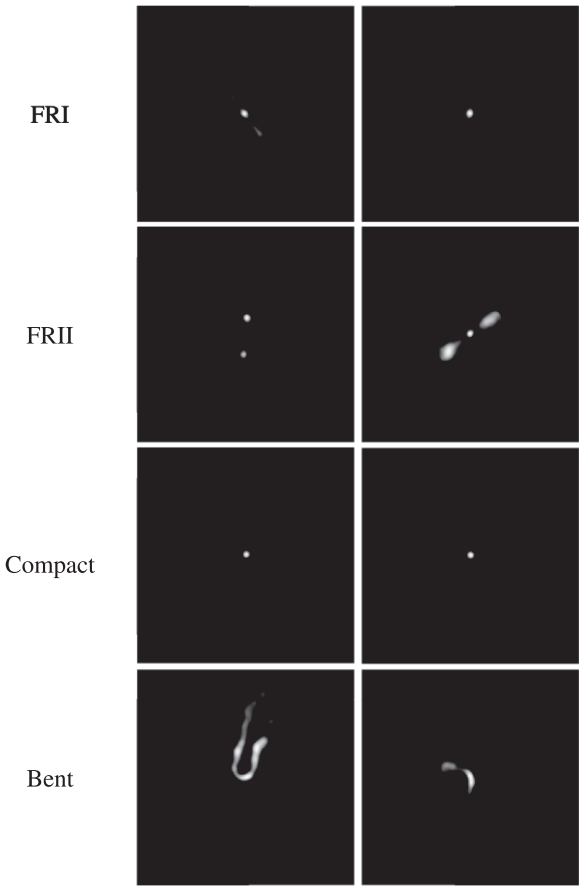


Figure 1. Class definition for FR I, FR II, compact, and bent. For the bent class we show an example of an NAT source in the left-hand panel and a WAT source in the right-hand panel.

we apply wGAN-supported augmentation to a CNN and a vision transformer (ViT, see Dosovitskiy et al. 2020).

The long-term goal is to use classification models to process incoming data from new radio telescopes. For this purpose, classification models need to generalize particularly well. A common problem in astronomy is the scarcity of labelled data in the face of large amounts of new data to process. This is a very different situation as for instance in particle physics, where simulations are highly fine-tuned and experiments are constantly repeated. In particular, for forthcoming radio surveys, and even for the majority of sources in FIRST, no morphological labels are available. As a result unsupervised, semisupervised, and self-supervised methods have gained attention without reaching the performance of supervised methods (Mostert et al. 2021; Slijepcevic et al. 2022a, b). The current classification scheme of radio galaxies and our physical interpretation will be challenged by new radio surveys. For instance, Mingo et al. (2019) detected a large population of low-luminosity FR II sources in the *LOFAR* Two-Metre Sky Survey (see Shimwell et al. 2019, 2022) that is not expected from the conventional FR distinction based on radio luminosity. Discoveries of rare morphologies can help to extend our understanding of radio sources, but are potentially prohibited by supervised learning techniques. Unsupervised methods such as self-organizing maps can be used efficiently to discover such rare morphologies (Mostert et al. 2021).

This paper is organized as follows: In Section 2, we introduce the data set used for training, validation, and testing. The generative

Table 1. Number of radio galaxy images per class in the train, validation, and test data sets.

	FR I	FR II	Compact	Bent	Total
5-Fold cross train	316	659	232	198	1405
5-Fold cross valid	79	165	59	50	353
Test	100	100	100	100	400
Total	495	924	391	348	2158
Relative frequency In total	0.23	0.43	0.18	0.16	1

model and its implementation are described in Section 3. The training procedure and the assessment of image quality are discussed in Section 4. The results of the comparison between only classical and classical plus wGAN-supported augmentation for different classifiers are presented in Section 5 before we conclude in Section 6.

2 DATA

We combine different catalogues (Gendre & Wall 2008; Gendre et al. 2010; Proctor 2011; Baldi et al. 2017; Capetti et al. 2017a, b; Miraghaei & Best 2017) that characterize radio sources from the FIRST survey to create a data set of 2158 radio galaxy images with morphological labels. The labelling in the catalogues is typically performed by experts by considering radio images and the corresponding optical counterparts. We group radio sources into four classes, namely FR I, FR II, compact and bent. The source coordinates are compared between catalogues to remove duplicates. Sources that appear with different labels are regarded as ambiguous and are removed entirely. More details on the acquisition of the data set can be found in Griese et al. (2023). The data set is published on zenodo (Griese et al. 2022) and on GitHub (https://github.com/floriangriese/RadioGalaxyDataset). The radio galaxy images of the FIRST survey are collected from the virtual observatory skyview.[1] We start from the original images with a size of (300 × 300) pixels. Then we adopt the preprocessing procedure from Aniyan & Thorat (2017). In particular, we set all pixel values below three times the local RMS noise to the value of this threshold. We apply classical augmentation to all images during training consisting of random rotations and reflections of the base image. This augmentation is done every time an image comes up in the training loop, so that the augmentation factor simply depends on the number of iterations of the training procedure. Consequently, classical augmentation retains the class imbalance present in the base image set. The augmented images are then cropped to the input size of our generative network, i.e. to (128 × 128) pixels. Subsequently, the pixel values are rescaled to the range $[-1, 1]$ to represent floating point grey-scale images.

We separated 100 sources per class from the data set for the final evaluation of our models. For validation purposes during training (e.g. choosing the best model), we use a 5-fold cross-validation. Therefore, we do not need a separate validation set. As a result, we lose less training data. In particular, we split the training set into five blocks and did five separate training runs. For each of these runs one of the five blocks was used as the validation set and the remaining four blocks represented the corresponding training set. The quantities per class and per split are shown in Table 1.

[1] https://skyview.gsfc.nasa.gov

3 WASSERSTEIN GAN

The ability to learn representations of underlying statistical distributions of data sets makes generative models a powerful tool for the creation of additional data points. In particular, sampling from those representations allows speeding up conventional simulation techniques significantly and may be useful for further subsequent treatments (Buhmann et al. 2021, 2022).

Three different categories of generative models are well-established: GANs, VAEs, and flow-based models. Diffusion models represent a relatively new development in this area. In this work, we focus on GANs. They consist of two neural networks: a generator G that generates fake images from a noise vector Z and the discriminator D that discriminates between real and fake images. This architecture was first introduced in Goodfellow et al. (2014) and Salimans et al. (2016). In a two-player minimax game, the generator learns to create fake images, which become less and less distinguishable from the real ones in the course of the training. The loss function for this set-up reads (Goodfellow et al. 2014; Salimans et al. 2016)

$$L = \min_G \max_D \mathbb{E}[\log D(x)] + \mathbb{E}[\log(1 - D(G(z)))], \tag{1}$$

where x represent real samples and $G(z) = \tilde{x}$ generated samples.

For this project, we employ a variant of the standard GAN set-up called wGAN that uses the Wasserstein-1 metric, also referred to as the Earth mover's distance, as main term in the loss function (Arjovsky et al. 2017). This loss function is calculated as

$$L = \sup_{f \in \mathrm{Lip}_1} \{\mathbb{E}[f(x)] - \mathbb{E}[f(\tilde{x})]\}, \tag{2}$$

where f denotes a 1-Lipschitz function that is learned during the training procedure. The discriminator of a standard GAN is transformed into a critic and is used to estimate the Wasserstein distance between real and generated images. Hence, the absolute value of the loss function is correlated with the image quality, resulting in the name change. Additionally, the training of wGANs is often more stable and more likely to converge than standard GAN set-ups. To approximate the Wasserstein-1 metric by use of a critic network, it has to be ensured that the 1-Lipschitz constraint is fulfilled. This is achieved by applying a gradient penalty term to the loss function as in Gulrajani et al. (2017):

$$L = \lambda \, \mathbb{E}[(\|\nabla_{\hat{x}} f(\hat{x})\|_2 - 1)^2] \tag{3}$$

for random samples $\hat{x} \sim \mathbb{P}_{\hat{x}}$.

Since we work with image data, it has proven to be the most promising approach to construct a wGAN set-up based on convolutional layers (Radford et al. 2015). The generator receives a noise tensor of size 100×1 and a class label y and, through multiple layers of two-dimensional (2D) transposed convolution operators, enlarges this to a 128 × 128 tensor, consistent with the dimensions of real images. The critic is given either real or generated images, as well as the class label y. The output of the critic is a single real value, which represents the belief of the critic for the image to be real. Generator and critic are trained intermittently, where the critic has five training cycles per training cycle of the generator. When training the generative model, it is necessary to apply classical augmentation such that the symmetries of the training set are also present in the generated data sets, and to avoid introducing a bias due to the limited number of training examples.

Morphologies of radio galaxies are diverse and result in very different images. Consequently, it is reasonable to condition the networks with the class label y such that a combination of image and class label is provided to the networks. In particular, this

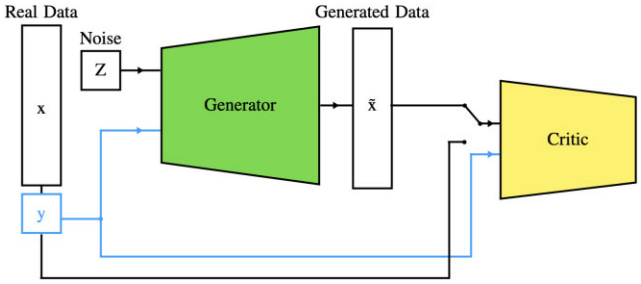


Figure 2. Schematic of the wGAN architecture, where y denotes the class label of real x or generated images $\tilde{x}$.

allows applying supervised learning techniques on the output of the generator. For our set-up, this is achieved for the generator by applying a 2D transposed convolution operator on a matrix of image dimensions filled with the class label. The transpose-convoluted layer is then concatenated to the first transpose-convoluted layer of the noise tensor. Batch normalization in 2D and ReLU (Rectified Linear Unit) activation functions are used. The concatenated tensor is then passed through five additional 2D transposed convolutions, where no normalization or activation is applied after the last layer. Instead, the individual pixel values are clipped to $[-1,1]$ for conversion to grey-scale. The critic is built analogously, but uses 2D convolutional layers, resulting in a single output node representing the critic score for image quality. Here, layer normalization and leaky ReLU functions are used except for the last layer. The leaky ReLU activation function is an attempt to avoid the 'dead neuron' phenomenon of the pure ReLU function, where any gradient information is lost if the input is negative. This makes the critic more stable against suboptimal starting points. The layer norm computes the normalization over the features instead of batches. A schematic of the wGAN set-up can be found in Fig. 2. For more details on the architectures see Tables B1 and B2.

4 RESULTS OF IMAGE GENERATION WITH A WGAN

4.1 Training

For each choice of training and validation data in the cross-validation procedure a wGAN training run is launched on the corresponding training set. The training is performed with a single NVIDIA A100 GPU provided by the Maxwell cluster at DESY for 40 000 generator iterations, i.e. weight updates. A batch size of 400 is chosen and one training run takes roughly 7 h to complete. The choice of the batch size did not have a strong impact on the performance of the model, so that we chose a size that still comfortably fits into the GPU's memory, while being large enough to fully profit from the computing speed-up of larger batches. The generator and critic weights are saved every 250 iterations, allowing scanning for the best training state later on, as described in the following section. Choosing such an iteration for every model and training run is necessary as wGAN training runs generally do not converge fully but rather fluctuate around an optimal value. This means that it is not instructive to simply use the final state of the model after training and instead other metrics need to be studied to choose an optimal working point. While comparing different model set-ups, we are only interested in the performance of these optimal working points. All models are implemented and trained in PyTorch (Paszke et al. 2019). For an overview of training details we refer to Table B5. The choice of hyperparameters is inspired by values obtained by Buhmann et al. (2021). With the exception of the learning rate, other hyperparameters have not been further optimized.

4.2 Evaluation of image quality

In this section, we present images created using the generator of the wGAN and examine the quality of the generated images in several ways.

4.2.1 Distribution-based comparison

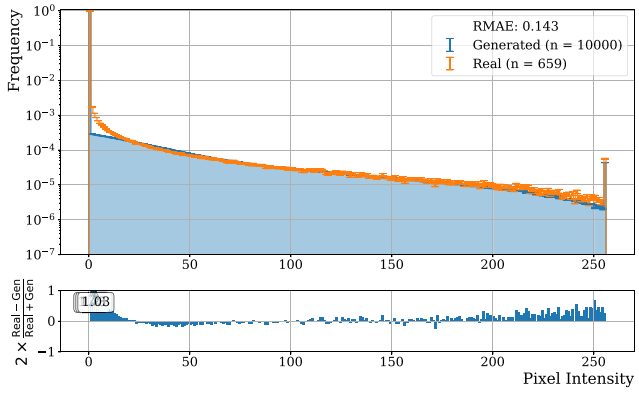
We define a set of distributions to compare generated images with the training data set, in order to determine the quality of generated images and thus to find the best performing training iteration. This includes normalized histograms of pixel intensities, the number of pixels with an intensity greater than zero, and of the sum of intensities. These histograms are compared for each class individually and the relative mean absolute error (RMAE) between the generated set of 10 000 images and the training set is computed. The RMAEs for the different distributions are summed up to yield a single figure-of-merit (FOM), where the wGAN training iteration with the lowest FOM value is used in the following as the best model. This procedure is followed for each of the four classes separately, i.e. we allow a different iteration of the generator training to yield the best model for each class. The chosen distributions are commonly used for images (e.g. photography), but it is important to note that they do not specifically contain information on the shape of the radio galaxies within these images. The choice of RMAE is based on its very fast computing time and robustness against empty bins while we acknowledge that other test metrics can be used.

Arbitrarily chosen examples of these distributions are shown in Fig. 3, where the distribution of the real images is shown in orange and the distribution of the generated images in blue. The uncertainty for each bin is given by the square-root of entries in that bin before normalization. The bottom panels in this figure show the per-bin divergence between the distributions, where absolute deviations larger than 1 are indicated by the corresponding value written in boxes. Here, only examples from the first cross-validation fold (of five) are shown.

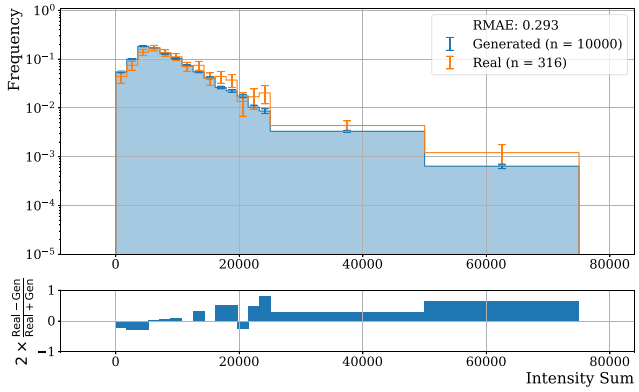
Overall, the distributions of the generated images tend to follow the distribution of the real images. Nevertheless, the generated images have difficulties in recreating very low, but non-zero, intensities. This can be seen for pixel values between 1 and 20 in Fig. 3a, which directly translates into underrepresenting the number of pixels with an intensity $I > 0$ in Fig. 3c.

4.2.2 Visual comparison

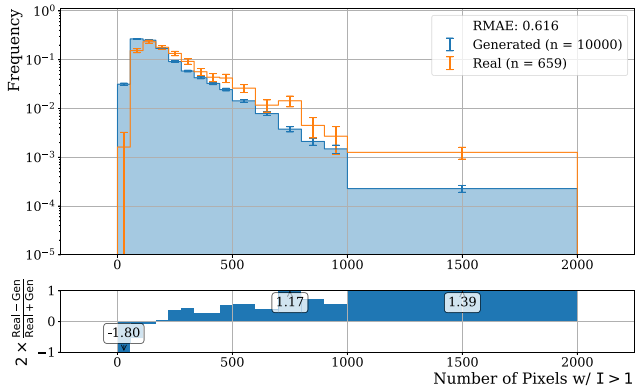
In order to get a visual idea of image quality, we generated a set of 5000 images per class and compared them to the full training data set over all cross-validation folds. The images are rotated so that their principal components are aligned. Subsequently, we compute the pixel-by-pixel difference for all possible pairs of real and generated images. All classes also include a few difficult to define sources with rather small spatial extension that are easy to emulate but do not show the generator's capability of reproducing the more interesting extended sources. Thus, we only consider images with an intensity sum of at least 15 000 (5000) for the extended (compact) radio galaxies. We show the resulting closest pairs for each class in Fig. 4. By eye, the generated images appear very similar to the analogue



(a) Pixel intensities of FRII sources.



(b) Sum of pixel intensities of FRI sources.



(c) Number of activated pixels of FRII sources.

Figure 3. Examples of image quality measures comparing histograms of real (orange) and generated (blue) images for the generator training iteration with the lowest combined RMAE for the corresponding class. The per-bin relative error is shown in the bottom of each panel. Histograms shown here are chosen arbitrarily from the first cross-validation fold.

real images, indicating a good performance of the generator set-up in terms of fidelity. In addition, the diversity of the generated data is crucial for the study in Section 5. To also get an impression of this diversity we show a random set of generated images in Appendix A.

4.2.3 Classifier-based comparison

Next, we use a CNN trained solely on the data set of real images to assess the image quality further. We compare the performance of the same classifier evaluated on the real test set and a set of generated images. The architecture of the CNN used for this experiment is summarized in Table B3 and the hyperparameters in Table B5. A comparison of the confusion matrices on both sets tests for any bias introduced by the image generation. In particular, we evaluate the conditioning on the class labels. In the top panel of Fig. 5, we show the confusion matrix of the classifier on the real test set. Comparing this to the confusion matrix obtained by the same classifier on a set of generated images on the bottom panel of Fig. 5, we find that the class conditioning of the generated images works overall quite well. However, confusion for images of the class FR I with the predicted classes FR II is enhanced on the generated test set. The classification performance of the Compact class is decreased on the generated test set, where particularly the misidentification of true compact class images as FR II images is increased. This might be due to the fact that some FR II-like sources resemble a combination of two compact sources. Confusion for true bent class images predicted to be of the FR I class is slightly reduced. The confusion between FR I and bent-tail sources is expected to be large as these classes contain sources that have faint, smeared out radio structures. In contrast, FR II and compact sources typically share sharp margins.

5 RESULTS OF CLASSIFIER TRAINING USING WGAN-SUPPORTED AUGMENTATION

We assess the new approach of supplementing the training set with generated images by comparing the performance of different classifiers (each trained on different setups with increasing amount of generated data). Our benchmark is the performance of the classifier trained on the original training set. We test the performance of the classifier trained with the original training set plus simulated images by the generator of the wGAN against this benchmark. We start with a FCN (see Table B4). Subsequently, we increase the complexity of the classifier by training a CNN (see Table B3). Finally, we apply our framework to a state-of-the-art classifier, namely the ViT (Dosovitskiy et al. 2020). Inspired by the performance of transformers in natural language processing, like BERT (Devlin et al. 2018) and GPT (Radford et al. 2018, 2019; Brown et al. 2020), ViTs are frequently used in computer vision tasks e.g. classification, object detection, and segmentation (Khan et al. 2022; Shamshad et al. 2022; Ulhaq et al. 2022). The self-attention mechanism enables learning long-range relationships between items within a sequence. Further, the architecture provides a scalability to high-complexity models (Khan et al. 2022). As the transformer assumes less prior knowledge than a CNN based model, it requires more training data, Thus, the transformer models are typically pre-trained on large-scale data sets to learn more general representations and afterwards the learned representations are fine-tuned to the task with limited data (Khan et al. 2022). In our case, we use the default ViT-B_16 ViT configuration with pre-trained weights from the ImagetNet21k data set[2] with a resetted head layer. The wGAN-generated images with pixel sizes 128 x 128 are zero-padded up to 224 × 224 pixels to fit the pre-trained model input size. As an attention based model, the ViT splits the image into fixed-size patches processed by the transformer encoder.

We generate images on the fly, i.e. each time a generated image is loaded it is newly generated. The images are generated such that the

[2]For the adopted ViT implementation see https://github.com/lucidrains/vit-pytorch and for the corresponding weights see https://github.com/google-research/vision_transformer.

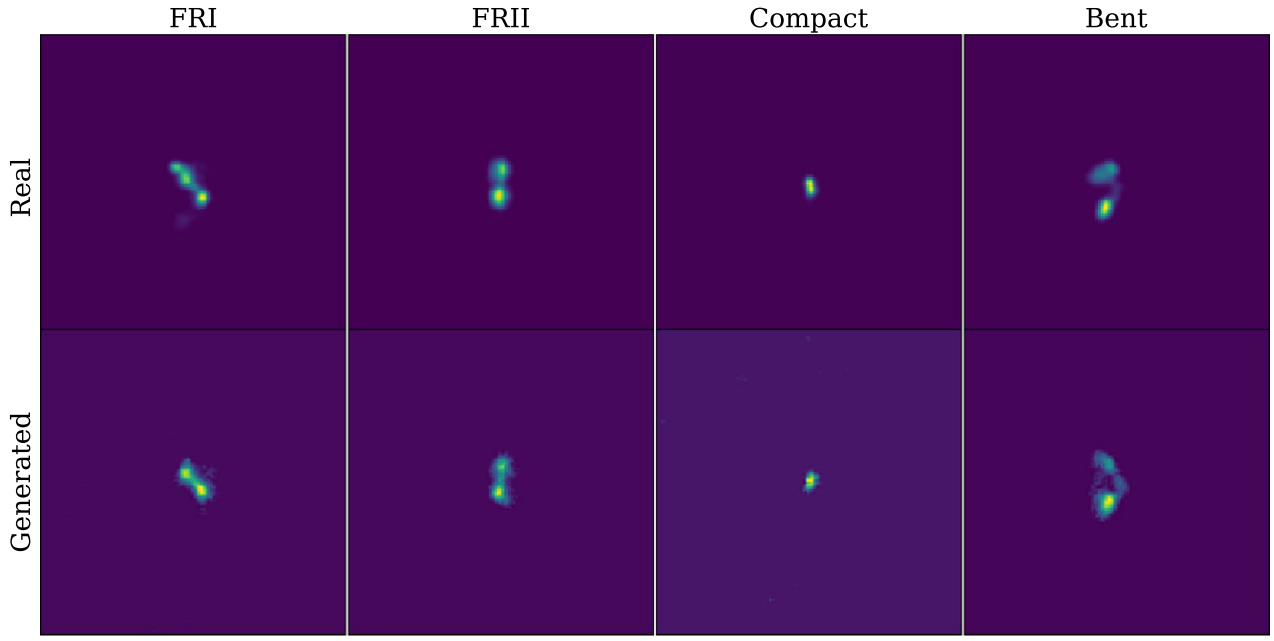


Figure 4. Closest matching pairs in terms of a pixel-wise difference between generated and real images for each class. The set of real images is the full training data set over all cross-validation folds; the generated data set consists of 5000 images per class. Images are aligned according to the first principal component.

resulting data set is balanced. As a loss function, cross-entropy loss is implemented, weighted for the imbalanced data only runs. The training set-ups are not optimized to reach maximal classification accuracies. The goal of this study is only to compare classical augmentation with wGAN-supported augmentation for each of the classifiers. We do not compare performance between the three classifiers in detail either. For further training details see Table B5.

5.1 Evaluation metrics

To compare the overall performance among different training setups and to determine the best training iteration of a classifier training run (see Fig. 6), we use the multiclass Brier score (Brier 1950). The Brier score is essentially the mean squared error of the predicted probabilities of a classifier for all classes. This has the advantage that also the certainty of the classifier's decision is considered, which winner-takes-all FOMs such as accuracy do not take into account. For each set-up, i.e. for a given ratio between the number of generated i_g and real images i_r, denoted $\lambda = i_g/i_r$, we have five models due to the 5-fold cross-validation.

The final evaluation is performed on an independent test set that contains real data only. We use the most commonly applied metric in radio astronomy publications: multiclass accuracy. In order to estimate statistical fluctuations, we average the performance metrics over the five best models of each cross-validation fold.

5.2 Accuracy

The multiclass accuracy, i.e. number of correct classifications over number of all classifications, on the test data set is shown in Fig. 7 for the three different classifiers investigated here. The results are shown for different training scenarios, where the number of generated images used to augment the training data set (represented by λ) is varied. The blue markers (uncertainty bars) represent the mean (standard deviation) of the obtained results over all cross-validation folds for the augmented training data sets and the horizontal orange line (area) show the corresponding result for the real data only case.

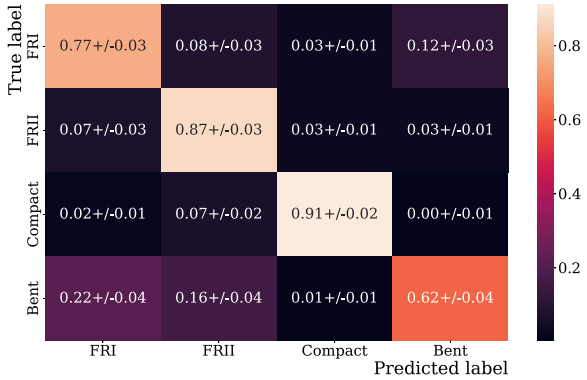
Fig. 7a presents the results for the FCN, which yields an improvement in accuracy of (17.5 ± 4.7) per cent over the baseline set-up at $\lambda = 2$. All augmented training set-ups outperform the real data only case which reaches an accuracy of (58.7 ± 1.8) per cent.

The highest obtained average for the CNN classifier is reached for $\lambda = 3$, as can be seen in Fig. 7b, which is (3.0 ± 1.8) per cent higher than the real data only baseline at (78.9 ± 1.1) per cent. The highest obtained average using wGAN augmented training data for the ViT classifier is reached at $\lambda = 2$, see Fig. 7c, which is (0.7 ± 2.0) per cent lower than the real data only baseline at (80.6 ± 0.9) per cent. For additional performance analyses per class we refer to Appendix C.

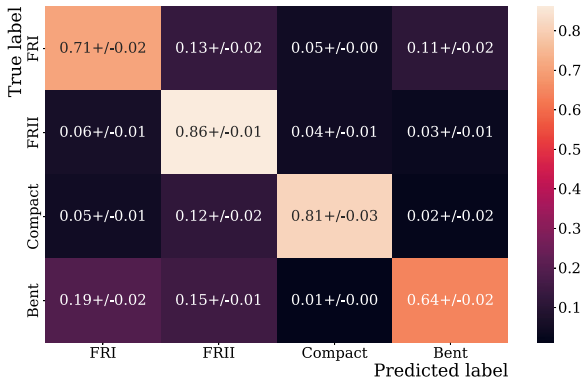
6 DISCUSSION AND CONCLUSION

The approach used for the study presented here, utilizing a wGAN, is novel to the field of radio astronomy. We are able to generate highly realistic images of radio sources of the four different radio galaxy classes. For this, we rely on the good agreement between the image metric distributions, such as the pixel intensity histogram, between real and generated images, as well as the good agreement between the confusion of a CNN classifier trained only on real data obtained on a real data only test set and a generated data only test set. Particularly the latter, provides confidence for the class conditioning of the generator.

Following a visual inspection, we note that the generated images tend to have sharper edges, i.e. low-intensity pixels directly next to high-intensity pixels. This is not the case for real images, which are smeared due to detector resolution effects. Resolving these issues would yield even more realistic generated images.



(a) Real test sample.



(b) Generated sample.

Figure 5. Confusion matrices on the real only test data set (above) and a data set of 4000 generated images, where each of the generators pertaining to a cross-validation fold contributes 200 images per class (5 generators × 4 classes × 200 images). Matrices are row-wise normalized and both data sets are class balanced. The values represent the mean and standard deviation of the confusion matrices obtained from each of the classifiers trained on the cross-validation folds with classical augmentation only.

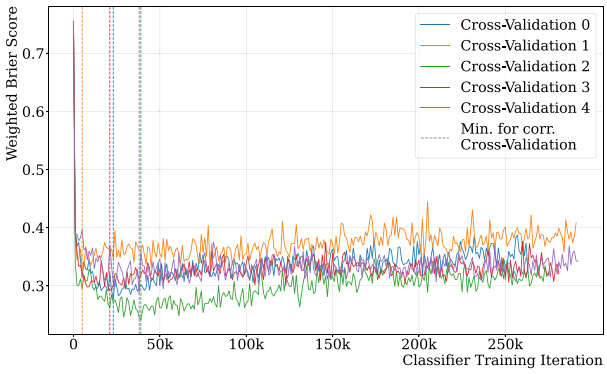
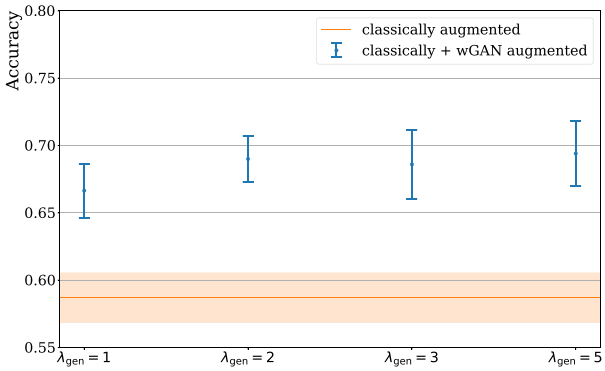
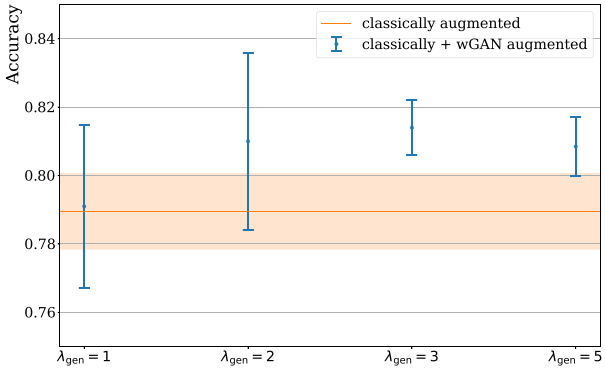


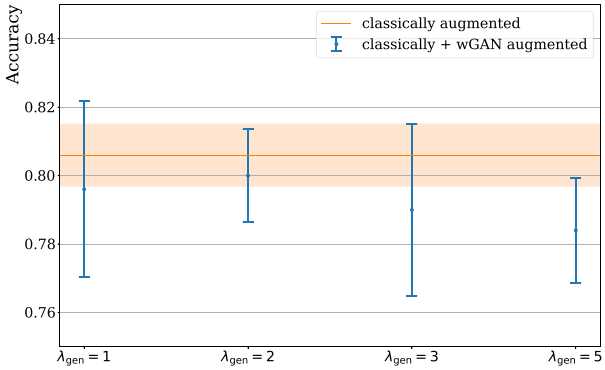
Figure 6. Weighted Brier score on the corresponding cross-validation validation sets per classifier training iteration. The iteration resulting in the minimum score is indicated by a dashed vertical line.



(a) FCN



(b) CNN



(c) ViT

Figure 7. Accuracy on the test data set for the three different classifier architectures for different training scenarios, where the amount of generated images used to augment the training data set (represented by λ) is varied. The blue markers (uncertainty bars) represent the mean (standard deviation) of the obtained results over all cross-validation folds for the classically + wGAN augmented training data sets and the horizontal orange line (area) show the corresponding result for the only classically augmented case.

However, we do not observe issues known from other state-of-the-art generative networks in radio astronomy. VAE-based models suffer from different noise levels between generated and training data or pseudo-textures and pseudo-structures (see e.g. Bastien et al. 2021). The results of this study therefore constitute a major improvement in generated image quality.

This high quality of the images allows us to use them to improve the training of an external classifier, called wGAN-supported augmentation here. This represents an extension to realistic data of the studies done in Butter et al. (2021), which showed that statistical information contained in a simplistic toy training data set can be augmented using generative models. Another extension of this study to more realistic data in the field of particle physics is given in Bieringer et al. (2022).

We find in agreement with these studies that generated images individually contain less information than real data. An additional test presented in Appendix D shows that the performance of a classifier worsens if the amount of real training data is reduced and replaced by generated data. However, the statistical power of the training set can be increased by the inclusion of generated data.

Here, we are able to show that adding generated images to the training data set does clearly improve the classifier performance on a real data only test set for the FCN classifier, where the largest improvement of (17.5 ± 4.7) per cent over the baseline set-up is reached for $\lambda = 2$, meaning a training data set consisting of all real images plus twice as many generated images. Additionally, similar improvements are seen for all other λ values that have been tested.

For the considerably more complex CNN classifier, the improvement is not so consistent and already the baseline performance is far better than even the enhanced performance of the FCN classifier. However, we do obtain a maximal improvement of (3.0 ± 1.8) per cent for $\lambda = 3$, which also represents the overall highest accuracy for any of the set-ups investigated here.

Finally, for the most complex classifier architecture, the ViT, we are not able to show a conclusive improvement of the classifier performance, so that we might expect a dependency of the ability of generated images to add useful information to the training data set on the baseline performance (often connected to the complexity) of the classifier in question. A naïve interpretation could be that the better performing architectures are simply more sensitive to even small differences between the real and generated images. Additionally, the robustness of the ViT might be an issue because it was pre-trained with natural images and only fine-tuned with radio galaxy images due to the limited data sample size.

Further, we considered a three-class classification problem with extended sources only. We found that the overall accuracy is reduced as compact sources are easier to classify. More importantly, the significance of the improvement by including generated images in the training is not enhanced as the variations in the cross-validation tend to increase as well.

The best overall accuracy is obtained by using the CNN and wGAN augmented training data, but only by a small margin. Yet, we have shown that wGAN augmentation works in principle (similar to the goal in Butter et al. 2021, as noted above) and can significantly improve a somewhat simpler algorithm. This can be useful for applications of classification algorithms in resource-constrained environments, i.e. disk-space and inference time restrictions.

Our generative model is able to generate large sets of radio galaxy images of different morphologies very quickly. A batch of 100 images can be generated on a NVIDIA V100 GPU in $\sim$0.1 s and in $\sim$4.5 s on CPU. Therefore, our wGAN can play an important role in the simulation and analysis of large radio surveys. Future work involving much larger training sets from the *LOFAR* telescope will explore this further. Moreover, wGAN-generated images can be used to validate new interferometric machine-learning algorithms (see e.g. Schmidt et al. 2022). To this end, we provide the model and weights with documentation at https://github.com/floriangriese/wGAN-supported-augmentation.

ACKNOWLEDGEMENTS

This work was supported by Universität Hamburg UHH, Deutsches Elektronen-Synchrotron DESY, Technische Universität Hamburg TUHH, and HamburgX grant LFF-HHX-03 to the Center for Data and Computing in Natural Sciences (CDCS) from the Hamburg Ministry of Science, Research, Equalities and Districts. This project benefits greatly from the exchange with particle physicists with a vast experience in using generative models for calorimeter simulations and was supported in part through the Maxwell computational resources operated at DESY. We acknowledge financial support from the Open Access Publication Fund of Universität Hamburg.

DATA AVAILABILITY

The code of all models trained for this work is publicly available on GitHub at the following address: https://github.com/floriangriese/wGAN-supported-augmentation.

The data set is available on Zenodo at: https://doi.org/10.5281/zenodo.7120632 and code for data loading on GitHub at: https://github.com/floriangriese/RadioGalaxyDataset. If you use this data set, please cite Griese et al. (2023).

REFERENCES

Alhassan W., Taylor A. R., Vaccari M., 2018, MNRAS, 480, 2085
Aniyan A. K., Thorat K., 2017, ApJS, 230, 20
Arjovsky M., Chintala S., Bottou L., 2017, preprint (arXiv:1701.07875)
Balakrishnan V., Champion D., Barr E., Kramer M., Sengar R., Bailes M., 2021, MNRAS, 505, 1180
Baldi R. D., Capetti A., Massaro F., 2017, A&A, 609, A1
Bastien D. J., Scaife A. M. M., Tang H., Bowles M., Porter F., 2021, MNRAS, 503, 3351
Becker R. H., White R. L., Helfand D. J., 1995, ApJ, 450, 559
Bieringer S. et al., 2022, J. Intrum., 17, P09028
Bonaldi A. et al., 2021, MNRAS, 500, 3821
Bowles M., Scaife A. M. M., Porter F., Tang H., Bastien D. J., 2020, MNRAS, 501, 4579
Bowles M., Bromley M., Allen M., Scaife A., 2021, preprint (arXiv:2111.04742)
Brier G. W., 1950, Mon. Weather Rev., 78, 1
Brown T. B. et al., 2020, preprint (arXiv:2005.14165)
Buhmann E., Diefenbacher S., Eren E., Gaede F., Kasieczka G., Korol A., Krüger K., 2021, Comput. Softw. Big Sci., 5, 13
Buhmann E. et al., 2022, Mach. Learn. Sci. Tech., 3, 025014
Butter A., Diefenbacher S., Kasieczka G., Nachman B., Plehn T., 2021, SciPost Phys., 10, 139
Capetti A., Massaro F., Baldi R. D., 2017a, A&A, 598, A49
Capetti A., Massaro F., Baldi R. D., 2017b, A&A, 601, A81
Carilli C., Furlanetto S., Briggs F., Jarvis M., Rawlings S., Falcke H., 2004, New Astron. Rev., 48, 1029
Cheng T.-Y. et al., 2020, MNRAS, 493, 4209
Devlin J., Chang M.-W., Lee K., Toutanova K., 2018, preprint (arXiv:1810.04805)
Dosovitskiy A. et al., 2020, preprint (arXiv:2010.11929)
Fanaroff B. L., Riley J. M., 1974, MNRAS, 167, 31P
Frid-Adar M., Klang E., Amitai M., Goldberger J., Greenspan H., 2018, preprint (arXiv:1801.02385)
Gendre M. A., Wall J. V., 2008, MNRAS, 390, 819

Gendre M. A., Best P. N., Wall J. V., 2010, MNRAS, 404, 1719
Goodfellow I. J., Pouget-Abadie J., Mirza M., Xu B., Warde-Farley D., Ozair S., Courville A., Bengio Y., 2014, preprint (arXiv:1406.2661)
Gowal S., Rebuffi S.-A., Wiles O., Stimberg F., Calian D. A., Mann T., 2021, preprint (arXiv:2110.09468)
Griese F., Kummer J., Rustige L., 2022, floriangriese/RadioGalaxyDataset: v0.1.1, Zenodo, available at: https://doi.org/10.5281/zenodo.7120632
Griese F., Kummer J., Connor P. L., Brüggen M., Rustige L., 2023, Data in Brief, 47, 108974
Gulrajani I., Ahmed F., Arjovsky M., Dumoulin V., Courville A., 2017, preprint (arXiv:1704.00028)
Hackstein S., Kinakh V., Bailer C., Melchior M., 2023, Astron. Comput., 42, 100685
Huertas-Company M., Lanusse F., 2023, PASA, 40, e001
Jonas J., MeerKAT Team, 2016, Proc. MeerKAT Sci.: On the Pathway to the SKA–PoS(MeerKAT2016), Vol. 277, The MeerKAT Telescope. SISSA, Trieste, p. 1
Khan S., Naseer M., Hayat M., Zamir S. W., Khan F. S., Shah M., 2022, ACM Comput. Surv., 54, 1
Krizhevsky A., Sutskever I., Hinton G. E., 2012, Proc. 25th Int. Conf. Neural Inf. Process. Syst. (NIPS'12) Vol. 1, ImageNet Classification with Deep Convolutional Neural Networks. Curran Associates Inc., NY, USA, p. 1097
Kummer J. et al., 2022, in Demmler D.,Krupka D., Federrath H., eds, INFORMATIK 2022, Lecture Notes in Informatics (LNI) - Proceedings. Gesellschaft für Informatik, Bonn, p. 469
Lintott C. J. et al., 2008, MNRAS, 389, 1179
Lukic V., Brüggen M., Mingo B., Croston J. H., Kasieczka G., Best P. N., 2019, MNRAS, 487, 1729
Ma Z., Zhu J., Li W., Xu H., 2018, in Proc. 25th IEEE Int. Conf. Image Process. (ICIP), Radio Galaxy Morphology Generation using Residual Convolutional Autoencoder and Gaussian Mixture Models. IEEE, Los Alamitos, CA, p. 3044
Ma Z., Zhu J., Zhu Y., Xu H., 2019a, Proc. 15th Int. Conf. Comput. Intell. Secur. IEEE, Los Alamitos, CA, p. 151
Ma Z. et al., 2019b, ApJS, 240, 34
McNamara B. R., Nulsen P. E. J., 2007, ARA&A, 45, 117
Maslej-Krešňáková V., El Bouchefry K., Butka P., 2021, MNRAS, 505, 1464
Mesarcik M., Boonstra A.-J., Meijer C., Jansen W., Ranguelova E., van Nieuwpoort R. V., 2020, MNRAS, 496, 1517
Mingo B. et al., 2019, MNRAS, 488, 2701
Miraghaei H., Best P. N., 2017, MNRAS, 466, 4346
Mosiane O., Oozeer N., Aniyan A., Bassett B. A., 2017, IOP Conf. Ser.: Mater. Sci. Eng. Vol. 198, Radio Frequency Interference Detection Using Machine Learning. IOP Publishing, Bristol, England, p. 012012
Mostert R. I. J. et al., 2021, A&A, 645, A89
Nair P. B., Abraham R. G., 2010, ApJS, 186, 427
Ntwaetsile K., Geach J. E., 2021, MNRAS, 502, 3417
Paszke A. et al., 2019, in Wallach H., Larochelle H., Beygelzimer A., d'Alché-Buc F., Fox E., Garnett R., eds, Advances in Neural Information Processing Systems 32. Curran Associates, Inc., NY, USA, p. 8024
Petrillo C. E. et al., 2017, MNRAS, 472, 1129
Porter F., 2020, MiraBest Batched Dataset. Zenodo, available at: https://doi.org/10.5281/zenodo.4288837
Proctor D. D., 2011, ApJS, 194, 31
Radford A., Metz L., Chintala S., 2015, preprint (arXiv:1511.06434)
Radford A., Narasimhan K., Salimans T., Sutskever I., 2018, Improving Language Understanding by Generative Pre-Training.
Radford A., Wu J., Child R., Luan D., Amodei D., Sutskever I., 2019, OpenAI blog, 1, 9
Sadeghi M., Javaherian M., Miraghaei H., 2021, AJ, 161, 94
Salimans T., Goodfellow I., Zaremba W., Cheung V., Radford A., Chen X., 2016, preprint (arXiv:1606.03498)
Samudre A., George L. T., Bansal M., Wadadekar Y., 2021, MNRAS, 509, 2269
Scaife A. M. M., Porter F., 2021, MNRAS, 503, 2369
Schawinski K., Zhang C., Zhang H., Fowler L., Santhanam G. K., 2017, MNRAS, 467, L110
Schmidt K., Geyer F., Fröse S., Blomenkamp P. S., Brüggen M., de Gasperin F., Elsässer D., Rhode W., 2022, A&A, 664, A134
Shamshad F., Khan S., Zamir S. W., Khan M. H., Hayat M., Khan F. S., Fu H., 2023 preprint (arXiv:2201.09873)
Shimwell T. W. et al., 2019, A&A, 622, A1
Shimwell T. W. et al., 2022, A&A, 659, A1
Slijepcevic I. V., Scaife A. M. M., Walmsley M., Bowles M., Wong O. I., Shabala S. S., Tang H., 2022a, MNRAS, 514, 2599
Slijepcevic I. V., Scaife A. M. M., Walmsley M., Bowles M., 2022b, Proc. 39th Int. Conf. Mach. Learn. (ICML 2022), Machine Learning for Astrophysics. JMLR, Inc. and Microtome Publishing, United States
Tang H., Scaife A. M. M., Leahy J. P., 2019, MNRAS, 488, 3358
Ulhaq A., Akhtar N., Pogrebna G., Mian A., 2022, preprint (arXiv:2209.05700)
van Haarlem M. et al., 2013, A&A, 556, A2
Vavilova I. B., Dobrycheva D. V., Vasylenko M. Y., Elyiv A. A., Melnyk O. V., Khramtsov V., 2021, A&A, 648, A122
Vazza F., Wittor D., Di Federico L., Brüggen M., Brienza M., Brunetti G., Brighenti F., Pasini T., 2023, A&A, 669, A50
Walmsley M. et al., 2019, MNRAS, 491, 1554
Wu C. et al., 2019, MNRAS, 482, 1211
Zhu X., Liu Y., Qin Z., Li J., 2017, preprint (arXiv:1711.00648)

APPENDIX A: GENERATED AND REAL IMAGES

Here, we show a random sample of 24 real images in Fig. A1 and 24 generated images in Fig. A2 in order to give a visual impression of the diversity of the data.

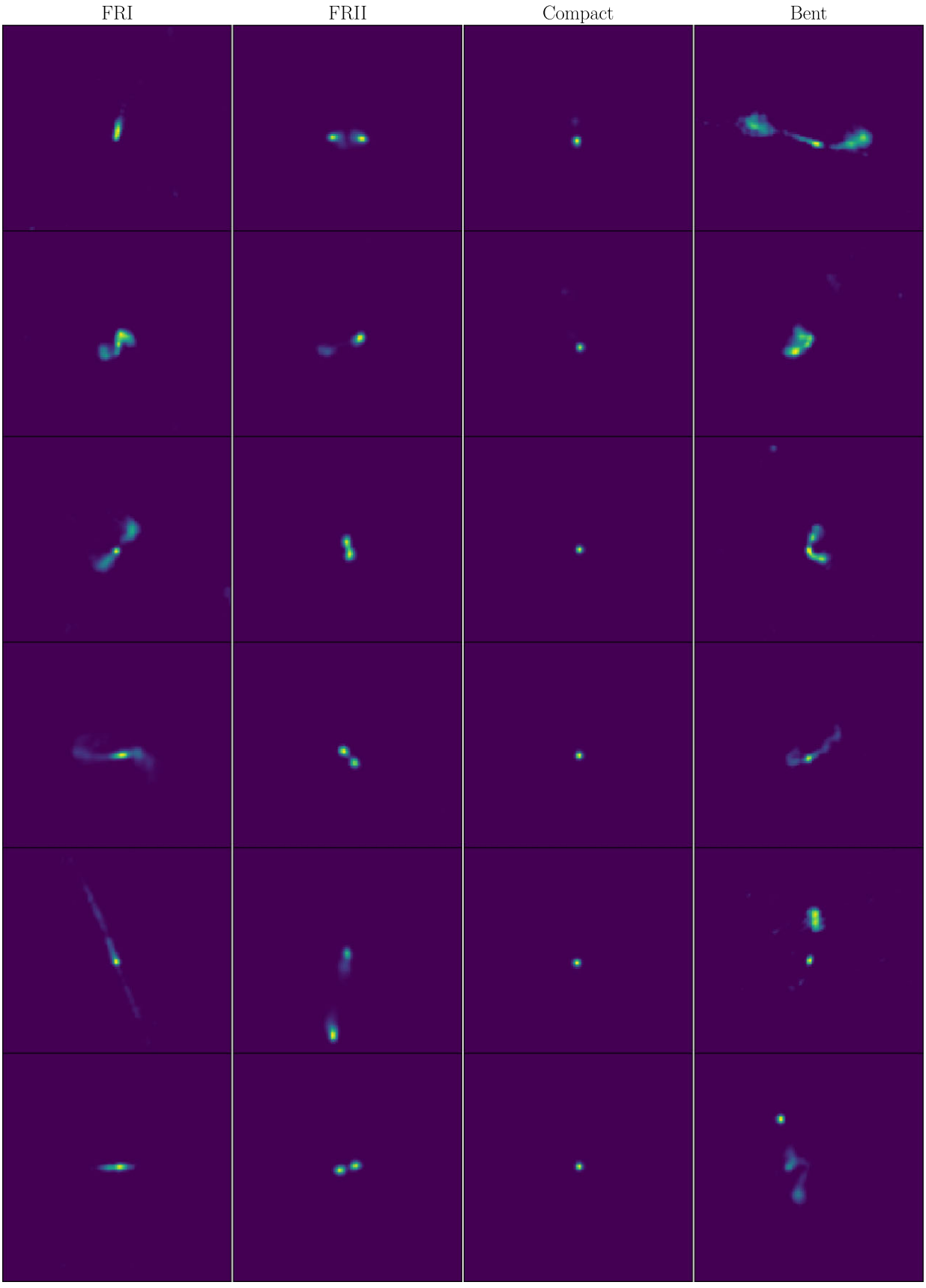


Figure A1. Real examples for each class.

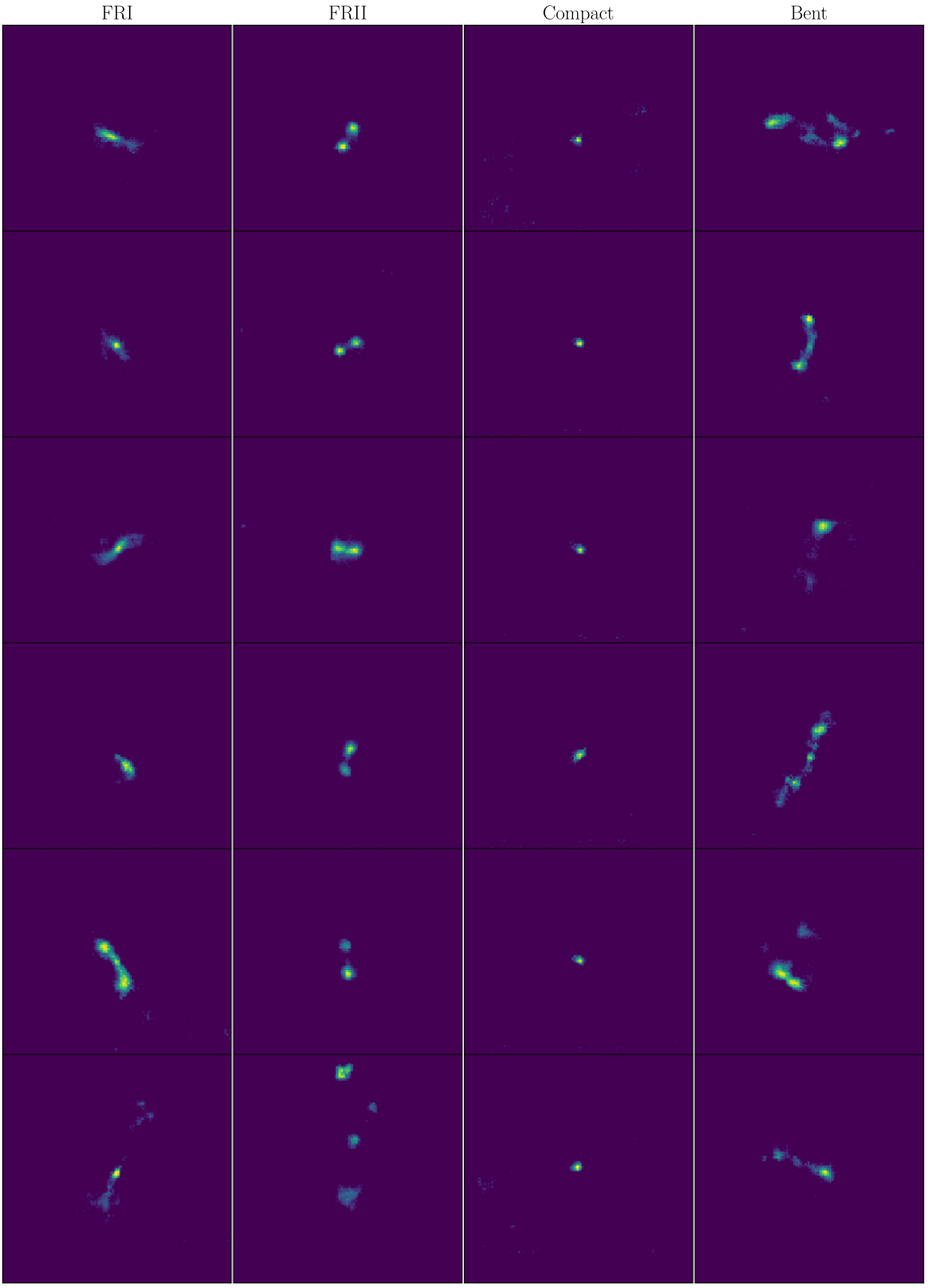


Figure A2. Generated examples for each class.

APPENDIX B: CLASSIFIER ARCHITECTURES

Detailed information about the architecture of the implemented models is given in this appendix. The structure and the corresponding number of parameters for the critic of the wGAN is given in Table B1, and for the generator in Table B2. Detailed information about the CNN is given in Table B3 and for the FCN in Table B4. In Table B5, we summarize the hyperparameters of all model trainings we conducted for this study.

Table B1. Parameters of the wGAN critic.

Layer	Name	Kernel size	Stride	Input channels	Depth	Activation	Regularizer	Parameters
1	Conv1	4 × 4	2	1	32	Leaky ReLU	Layer norm	512
2	Conv2	4 × 4	2	4	32	Leaky ReLU	Layer norm	2048
3	Conv3	4 × 4	2	64	128	Leaky ReLU	Layer norm	133 120
4	Conv4	4 × 4	2	64	256	Leaky ReLU	Layer norm	524 800
5	Conv5	4 × 4	2	256	512	Leaky ReLU	Layer norm	2097 280
6	Conv6	4 × 4	2	512	1024	Leaky ReLU	Layer norm	8388 640
7	Conv7	4 × 4	1	1024	1	–	–	16 384
Total parameters:								11 162 784

Table B2. Parameters of the wGAN generator.

Layer	Name	Kernel size	Stride	Input channels	Depth	Activation	Regularizer	Parameters
1	ConvT1	4 × 4	1	100	512	ReLU	Batch norm	820 224
2	ConvT2	4 × 4	1	4	512	ReLU	Batch norm	33 792
3	ConvT3	4 × 4	2	1024	512	ReLU	Batch norm	8389 632
4	ConvT4	4 × 4	2	512	256	ReLU	Batch norm	2097 664
5	ConvT5	4 × 4	2	256	128	ReLU	Batch norm	524 544
6	ConvT6	4 × 4	2	128	64	ReLU	Batch norm	131 200
7	ConvT7	4 × 4	2	64	1	–	–	1024
Total parameters:								11 998 080

Table B3. Parameters of the CNN classifier.

Layer	Name	Kernel size	Stride	Input channels	Depth	Activation	Regularizer	Parameters
1	Conv1	3 × 3	2	1	8	Leaky ReLU	Layer norm	8264
2	Conv2	3 × 3	2	8	16	Leaky ReLU	Layer norm	3200
3	Conv3	3 × 3	2	16	32	Leaky ReLU	Layer norm	5120
4	Conv4	3 × 3	2	32	32	Leaky ReLU	Layer norm	9344
5	Conv5	2 × 2	1	32	16	Leaky ReLU	–	2048
6	Fully connected 1			7 × 7 × 16	100	Leaky ReLU	–	78 500
7	Fully connected 2			100	4	ReLU	–	404
8	Softmax							
Total parameters:								106 880

Table B4. Parameters of the FCN classifier.

Layer	Name	Input channels	Depth	Activation	Parameters
1	Fully connected 1	128 × 128	250	Leaky ReLU	4096 250
2	Fully connected 2	250	250	Leaky ReLU	62 750
3	Fully connected 3	250	250	Leaky ReLU	62 750
4	Fully connected 4	250	250	Leaky ReLU	62 750
5	Fully connected 5	250	4		1004
6	Softmax				
Total parameters:					4285 504

Table B5. Hyperparameters of the trainings.

	Batch size	Learning rate	Optimizer	β_1	β_2	Momentum	Training time	Iterations
wGAN	400	0.0001	Adam	0	0.9	–	≈7 h	40 000
Classifier (FCN & CNN)	250	0.001	Adam	0.9	0.999	–	72 h	≈300 000
ViT	32	0.03	SGD	–	–	0.9	≈8 h	10 000

APPENDIX C: CLASS-WISE PERFORMANCE

In this section, we demonstrate the performance per class of the classifiers studied in Section 5. In particular, we show the class-wise precision in Fig. C1, recall in Fig. C2, and $F1$ score in Fig. C3.

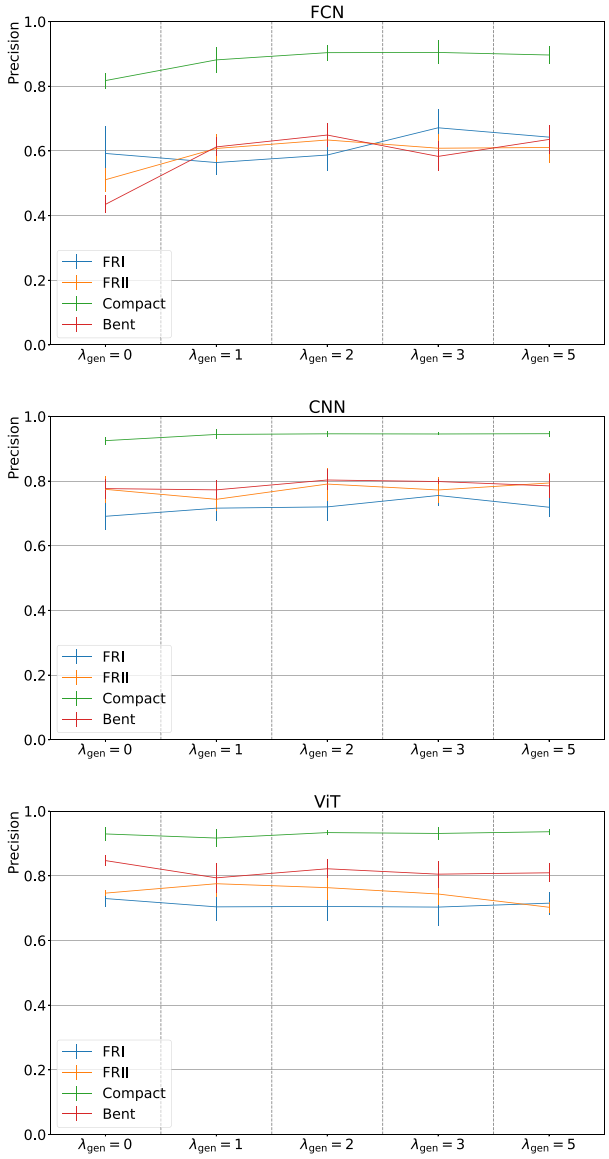


Figure C1. Precision for each class on the test data set for the three different classifier architectures for different training scenarios. The markers (uncertainty bars) represent the mean (standard deviation) of the obtained results over all cross-validation folds for the classically + wGAN augmented training data sets.

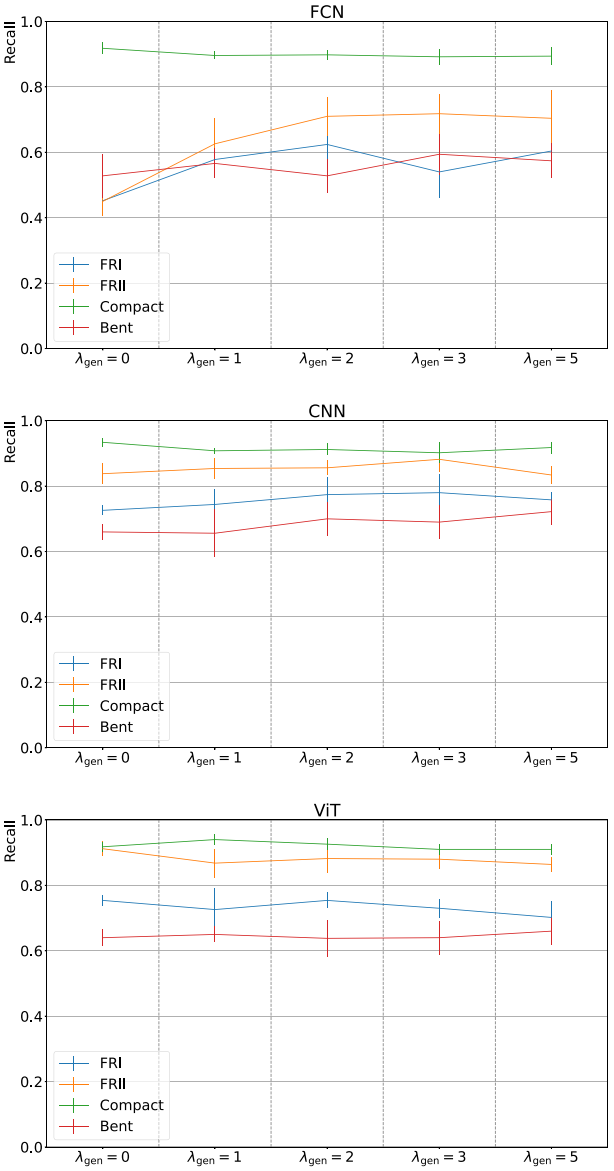


Figure C2. Recall for each class on the test data set for the three different classifier architectures for different training scenarios.

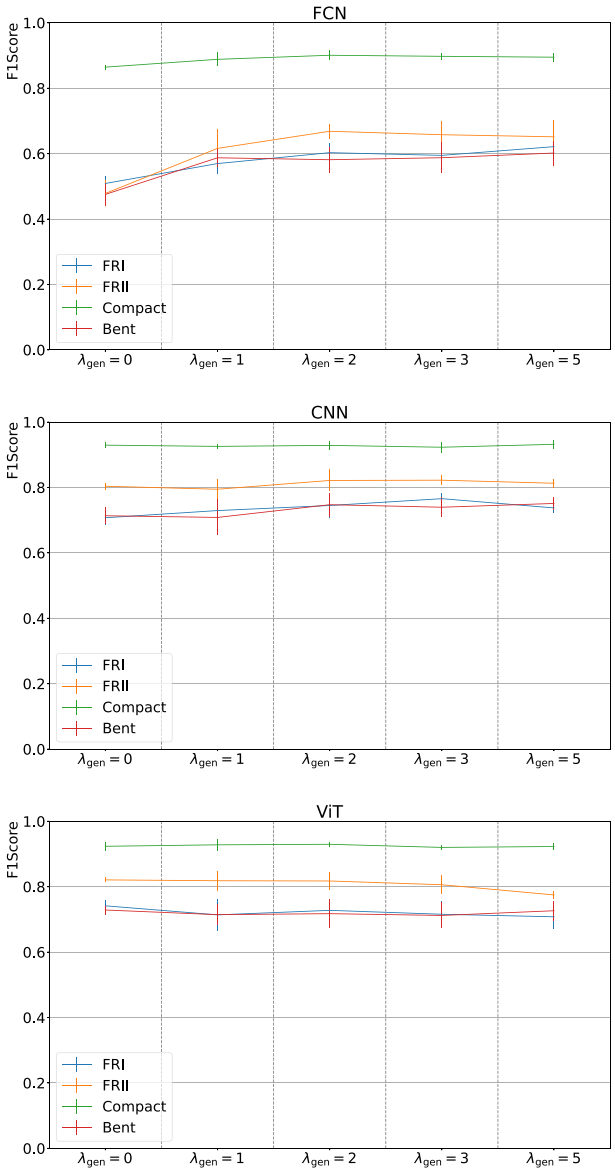


Figure C3. $F1$ score for each class on the test data set for the three different classifier architectures for different training scenarios.

APPENDIX D: ADDITIONAL TEST

Here, we present an additional test to compare the information content of real and generated images during classifier training. The classifier architecture for this test is the CNN introduced in Table B3. We train the CNN on different compositions of the original training set and a batch of generated images of the same size. We observe that the classifier performance worsens gradually as we remove real images and add generated images to keep the size of the training set fixed (see Fig. D1). From this experiment we can confidently conclude that the generated images are less informative compared to real images. Note, that we had to exclude some runs with low amount of real data due to the inability to classify the compact sources correctly along with the extended sources.

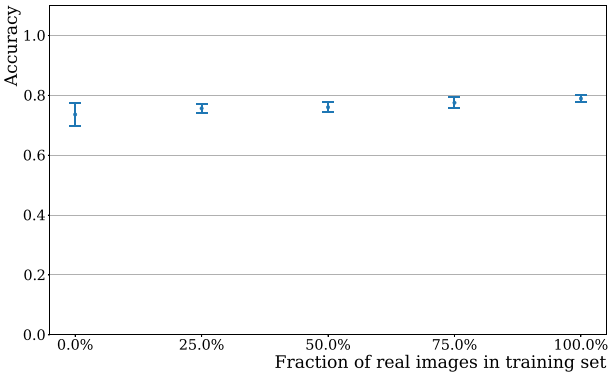


Figure D1. Accuracy on the test set achieved by the best CNN model trained on combined, i.e. generated + real, data sets with varying fractions of real images. The overall number of images in the training set corresponds to the full real-only training set and class imbalance is kept. Classical augmentation is used on both types of images during training.

This paper has been typeset from a TEX/LATEX file prepared by the author.