

Generative deep fields: arbitrarily sized, random synthetic astronomical images through deep learning

Michael J. Smith^{1,2} and James E. Geach^{1,2}  ^{1,2}★

¹Centre for Astrophysics Research, School of Physics, Astronomy & Mathematics, University of Hertfordshire, Hatfield AL10 9AB, UK

²Centre of Data Innovation Research, School of Physics, Astronomy & Mathematics, University of Hertfordshire, Hatfield AL10 9AB, UK

Accepted 2019 October 1. Received 2019 September 23; in original form 2019 April 29

ABSTRACT

Generative adversarial networks (GANs) are a class of artificial neural network that can produce realistic, but artificial, images that resemble those in a training set. In typical GAN architectures these images are small, but a variant known as Spatial GANs (SGANs) can generate arbitrarily large images, provided training images exhibit some level of periodicity. Deep extragalactic imaging surveys meet this criteria due to the cosmological tenet of isotropy. Here we train an SGAN to generate images resembling the iconic *Hubble Space Telescope* eXtreme Deep Field (XDF). We show that the properties of ‘galaxies’ in generated images have a high level of fidelity with galaxies in the real XDF in terms of abundance, morphology, magnitude distributions, and colours. As a demonstration we have generated a 7.6-billion pixel ‘generative deep field’ spanning 1.45 deg. The technique can be generalized to any appropriate imaging training set, offering a new purely data-driven approach for producing realistic mock surveys and synthetic data at scale, in astrophysics and beyond.

Key words: methods: miscellaneous – methods: statistical – surveys.

1 INTRODUCTION

Synthetic, or mock, data play an important role in the interpretation of observations, as it provide a means to test a theoretical framework, as a tool to explore biases or systematics in data analysis, or to help design future experiments. In astrophysics, like many fields, a standard route to generating synthetic data is to use an analytic or semi-analytic model, or numerical simulation, to generate synthetic data that mimic the observations (e.g. Cole et al. 1998; Obreschkow et al. 2009; Mandelbaum et al. 2012). The most common form of observation in astrophysics is digital imaging, and deep extragalactic imaging surveys have transformed our understanding of the Universe (Williams et al. 1996; Scoville et al. 2007).

Current methods to generate synthetic deep fields include the projection of volume- and resolution-limited hydrodynamical cosmological simulations (e.g. Vogelsberger et al. 2014; Schaye et al. 2015) into light-cones (Snyder et al. 2017), requiring a treatment for the transport of radiation through the volume and modelling of a particular instrument response (Jonsson 2006; Trayford et al. 2017). Alternatively, mock deep fields can be created by taking an input catalogue containing the positions and properties of fake galaxies and applying models to describe their light profiles (Bertin 2009; Dobke et al. 2010; Bergé et al. 2013; Peterson et al. 2015; Rowe et al. 2015). Mock blank fields can be created entirely parametrically, as in the case of the Ultra Fast Image Generator

(UFIG; Bergé et al. 2013), and these have the advantage of having a very well-defined input model. However, analytic and semi-analytic models require explicit encoding of astrophysical properties such as light profiles, and are therefore limited in realism. This is important when considering objects in surveys that are poorly described in this way, for example interacting/merging or high-redshift galaxies. Therefore, if the physical model is incomplete or flawed or too simplistic, simulations produced by the model may not be representative of real data. Furthermore, additional processing is required to produce a realistic synthetic observation, for example an understanding of the noise properties and convolution with a point spread function. Letting a machine infer both the astrophysical *and* the instrumental properties of a data set sidesteps these issues: with a large representative data set it is possible to use empirical data to construct new, realistic, but synthetic observations, at the expense of model transparency.

Generative adversarial networks (GANs) are a type of deep learning algorithm that can generate new samples from a probability distribution learnt from a representative training set (Goodfellow et al. 2014; Radford, Metz & Chintala 2016; Salimans et al. 2016). The adversarial aspect of the algorithm refers to the use of two neural networks – a generator G and discriminator D – that compete during training. G tries to estimate the probability distribution of the input data by producing samples that aim to trick D , which is estimating the probability that the generated sample came from the training set. Training is a ‘minimax’ game where G is trying to maximize the likelihood that D predicts the generated samples are from the real data. The generator transforms a ‘latent’ vector z , into an output

* E-mail: j.geach@herts.ac.uk

$G(z)$. Meanwhile, the discriminator takes either the output of the generator $G(z)$, or a real data example x , and transforms this input into an output, $D(G(z))$ or $D(x)$. The output can be thought of as the probability that $G(z)$ is indistinguishable from x . The networks are trained through gradient descent (Robbins & Monro 1951) until $G(z)$'s distribution closely matches the distribution of x . After training, the generator can produce convincing images resembling those in the training set. The current state of the art is capable of producing very convincing imagery (Karras et al. 2017; Karras, Laine & Aila 2018; Brock, Donahue & Simonyan 2019; Karnewar, Wang & Iyengar 2019). These generated images can be thought of as random draws from the probability distribution estimated by the generator that describes the distribution from which pixels in the training image were sampled.

One limitation of the GAN technique is its instability during training (Kodalg et al. 2017; Roth et al. 2017). We mitigate this somewhat by incorporating the relativistic discriminator introduced in Jolicoeur-Martineau (2019) (see Section 2.3). Also, a GAN model is inferred completely on training data. Therefore the trained model is only as accurate as the given data, and preferably a lot of data is required for training. Model interpretability can also be an issue with large neural networks.

In this work, we present a method exploiting GANs to generate realistic, but random, extragalactic deep field images of arbitrary size. In Section 2, we describe the ‘Spatial GAN’ variant, and explain how it is trained to generate fake images. In Section 3, we describe our results, comparing ‘galaxies’ detected in the fake images to galaxies detected in the training image. We discuss and summarize the results and highlight limitations and scope for improvement and future work in Section 4. Magnitudes are all quoted on the AB system.

2 METHOD

2.1 Spatial GANs

A Spatial GAN (SGAN; Jetchev, Bergmann & Vollgraf 2016) is a fully convolutional GAN that uses variably sized 2D latent vector arrays as the generator input z . This is in contrast to a standard GAN, which uses a 1D latent vector. In a standard GAN a dense neural layer is used to connect and reshape the latent vector so that a convolutional layer can operate on it. This lack of a fully convolutional architecture means that a standard GAN can only produce images of a single, fixed shape. An SGAN replaces all dense neural layers in both its generator and its discriminator with convolutional layers. This allows input of variably sized image–latent vector array pairs. In an SGAN the latent vector arrays are upsampled using deconvolving layers in the generator, and both generated and real images are downsampled in the discriminator via convolving layers. Since z can be varied, an SGAN can be used to create an image of any size, even one much larger than seen in the training set. If the training images exhibit periodicity, the SGAN’s generator will also learn to exhibit the same periodicity (Jetchev et al. 2016).

Note that the presence of periodicity in the training data is not a requisite for producing arbitrarily sized output images, but it is a requisite for producing realistic output images. An SGAN trained on an image of a cluster of galaxies, for example, will only produce output images that contain galaxy clusters, regardless of size. However, cosmic isotropy means that SGANs trained on deep ‘blank field’ extragalactic images can in principle produce output images of arbitrary size that resemble the real Universe.

The important caveat and limitation is that the generated images will only contain objects similar to those present in the training image; if the training image is too small to contain a representative sampling of the galaxy population (e.g. the rare massive clusters), then these types of object will not appear in the output. The SGAN cannot extrapolate in this manner, and this should be an important consideration when training and using such networks.

2.2 Training set

We use a training set comprised of the *F814W*, *F775W*, and *F660W* bands of the *Hubble Space Telescope* eXtreme Deep Field (XDF; Illingworth et al. 2013), with all images aligned and sampled on the same 60-mas grid.¹ The only other preprocessing is a channel-wise clip at the 99.99th percentile. Unlike many GAN approaches that use training images scaled to an 8-bit depth per channel, we train using the full floating point dynamic range of the data, with 32-bit depth per channel. Training images are sampled from the full XDF image by cropping the image at random positions with crop sizes of 64, 128, or 256 pixels, with the size fixed for each batch. Corresponding noise arrays of sizes 4, 8, or 16 pixels are sampled from a Gaussian distribution with a zero mean and unity variance, and passed to the generator. Each noise array has a channel axis of size 50.

2.3 Architecture and training

The generator is comprised of one initial deconvolutional layer with a stride of 2 and a kernel size of 4. Three deconvolutional sets are then applied, each comprising of one layer with a stride of 2, followed by three layers with strides of 1. Each layer in these sets have kernel sizes of 4. These layers have an exponential linear unit (ELU) activation (Clevert, Unterthiner & Hochreiter 2016). The generator’s output layer is also deconvolutional, with a stride of 1, a kernel size of 3, and three filters. The output layer has a sigmoid activation function,

$$S(x) = (1 + e^{-x})^{-1}. \quad (1)$$

The discriminator comprises of five initial convolutional layers, each with a stride of 2, and a kernel size of 4. All initial layers have a leaky rectified linear unit (ReLU) activation (Maas, Hannun & Ng 2013). 2D global average pooling is applied to the final convolutional layer, and a dense layer is used to connect the global average pool to a binary classification output.

The discriminator uses a relativistic average loss (Jolicoeur-Martineau 2019). A relativistic discriminator estimates whether the incoming data are more realistic than a random sample of the opposing type. To understand why this is important, consider that in a standard GAN a perfect generator will cause the discriminator to define all incoming data as ‘real’. However, it is known that exactly half of an incoming batch is real data. Therefore, if the generator is producing flawless fakes, the discriminator should assume that each sample has a 50 per cent probability of being real (Jolicoeur-Martineau 2019). To take this prior knowledge into account, the output function of a non-relativistic discriminator is modified from $D(x) = S(C(x))$, where $C(x)$ is the output of the final layer with no applied activation function, to $D_R(x)$

$$D_R(x) = \begin{cases} S(C(x) - \mathbb{E}_{x_f \sim \mathcal{Q}} C(x_f)) & \text{for real } x, \\ S(C(x) - \mathbb{E}_{x_r \sim \mathcal{P}} C(x_r)) & \text{for generated } x, \end{cases} \quad (2)$$

¹<https://archive.stsci.edu/prepds/xdmf>

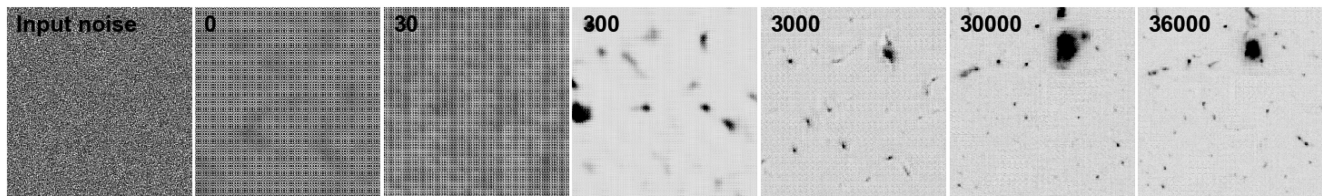


Figure 1. Evolution of SGAN training. The images show a 256×256 pixel generated image for the $F775W$ channel after 0, 30, 300, 3000, 30000, and 36000 epochs using the same input latent noise vector, z . The distribution of input (Gaussian) noise before training is shown in the first panel. Each image is linearly scaled with a 0.5–99.5 per cent percentile clip. Structure becomes apparent after a few hundred training epochs and is continuously refined as training progresses. The total training time to 36000 epochs is 1280 h wall time on an NVIDIA Tesla K40c GPU.

where S is the sigmoid activation function. The second parts of the real and generated $D_R(x)$ are effectively the average discriminator value for fake images, x_f , and real images, x_r , respectively. The use of a relativistic discriminator leads to stable training on a difficult data set of large images, where a standard discriminator fails (Jolicœur-Martineau 2019). The discriminator is packed to stabilize training (Lin et al. 2018). To pack the discriminator, two images from the same class are concatenated along their channel axes and fed into the discriminator as a single sample. Packing the discriminator in this way reduces the possibility of mode collapse (Lin et al. 2018). The Adam optimizer (Kingma & Ba 2015) is used, with a learning rate of 0.0002. The learning rate was determined through a manual search as the maximum rate that yields stable training.

The SGAN is trained on an NVIDIA Tesla K40c GPU for 36000 epochs of 30 batches with a batch size of 128. Each epoch requires approximately 120 s, depending on the batch crop sizes. The evolution of the generated images for a fixed latent noise vector z is shown in Fig. 1, showing the emergence of amorphous structure and then refinement into structures resembling galaxies with increasing training time.

3 RESULTS

A self-ensemble of 21 GANs is created by taking every tenth epoch’s model from a range of 200 epochs. A convolutional neural network (CNN) ensemble is a combination of different CNNs trained from different weight initializations on the same data. A self-ensemble is an ensemble of CNNs trained from the same initialization, but taken at different points in the training cycle. Ensembles of CNNs can produce significantly more accurate predictions when compared to a single CNN (Nanni, Ghidoni & Brahnam 2018; Paul et al. 2018). This increase in accuracy has been shown by Wang, Zhang & van de Weijer (2016) and Mordido, Yang & Meinel (2018) to also be present in GAN ensembles.

Four outputs were taken from each of these 21 GANs, resulting in 84 total XDF simulations. To generate a simulation set, each of the trained generators is fed a z vector that has shape $[b, 202, 202, 50]$, where $b = 4$ is the batch size. The generators upsample z to a shape $[b, 3232, 3232, 3]$, matching the shape of the training image. The generation time for single realization of the XDF is 15 s for all three bands on an Intel Xeon CPU E5-2650 v3. Using the same CPU it would take 1.5 h wall time to generate a 3-billion pixel image similar to one produced by the LSST. In principle, the generated image can be of arbitrary contiguous size by using seamless tessellation. Seamless tessellation can be achieved with SGAN by having adjacent tiles a and b share a portion of boundary z values. This creates a shared area where $G(z_a)$ and $G(z_b)$ have an equal output. A seamless tile is made by cropping at the midpoint of the shared noise and concatenating $G(z_a)$ and $G(z_b)$. To

illustrate, we have generated a contiguous 87040×87040 pixel ($1.45^\circ \times 1.45^\circ$) version of the XDF that can be examined online at <http://star.herts.ac.uk/~jgeach/gdf.html>.

Fig. 2 presents an example of a generated field in comparison to the real XDF, combining the $F814W$, $F775W$, and $F606W$ bands into an RGB colour composite. We show the full field, spanning approximately 3 arcmin, and a zoom-in of a 45 arcsec region for closer inspection. The generated image is reasonably convincing as an extragalactic deep field from the point of view of cursory visual inspection, although like many GAN-generated images, on closer inspection there are artefacts (such as low surface brightness features, pixelization, and discontinuities not seen in the real image) that betray the counterfeit. These artefacts are likely caused by a lack of resolution in the GAN generator, and could be remedied through the addition of more layers, or through the use of a larger convolutional kernel. Both of these approaches would increase the receptive field. The GAN also cannot reproduce some of the larger galaxies with the same level of detail as the real image. Nevertheless, the generated image does contain a mixture of early- and late-type galaxies amongst a more numerous field of background sources. The early and late types have the correct colour and morphology, i.e. classic red/elliptical and blue/spiral, respectively, and background sources have the clumpy and disturbed morphology typically seen in the high-redshift galaxy population. Fig. 3 presents single-band ($F775W$) thumbnail images of a sample of 100 randomly chosen sources with $F775W$ AB magnitudes in the range 21–26 mag. 50 sources are selected from the real image and 50 are selected from the generated images, but the thumbnails have been randomly shuffled in the figure to demonstrate to the reader the visual similarity of individual galaxies in the generated and real data.

To test whether the similarity between the generated and real image is more than superficial, we extract ‘galaxies’ from the generated images and compare to those in the real XDF. We use the source finder SEXTRACTOR (Bertin & Arnouts 1996) to detect galaxies in the combined $F606W+F775W+F814W$ (pixelwise-mean) images, measuring the corresponding source flux in the individual bands. Sources are identified with the criteria that five contiguous pixels have a signal $\geq 5\sigma$ above the local background. The same detection procedure is applied to the real and generated images, and the photometric zero-points for the generated images are identical to the real data. Fig. 4 compares the magnitude distribution of sources detected in the real field and ensemble of generated fields measured in each of the $F606W$, $F775W$, and $F814W$ bands. The generated and real distributions bear close statistical resemblance. The absolute difference in the generated and real median magnitudes in each of the $F606W$, $F775W$, and $F814W$ bands is within 0.02 mag, and the generated images produce the correct abundance of galaxies across the full magnitude range in every band. The p -values given by a two-sided Kolmogorov–Smirnov test for a magnitude limit of

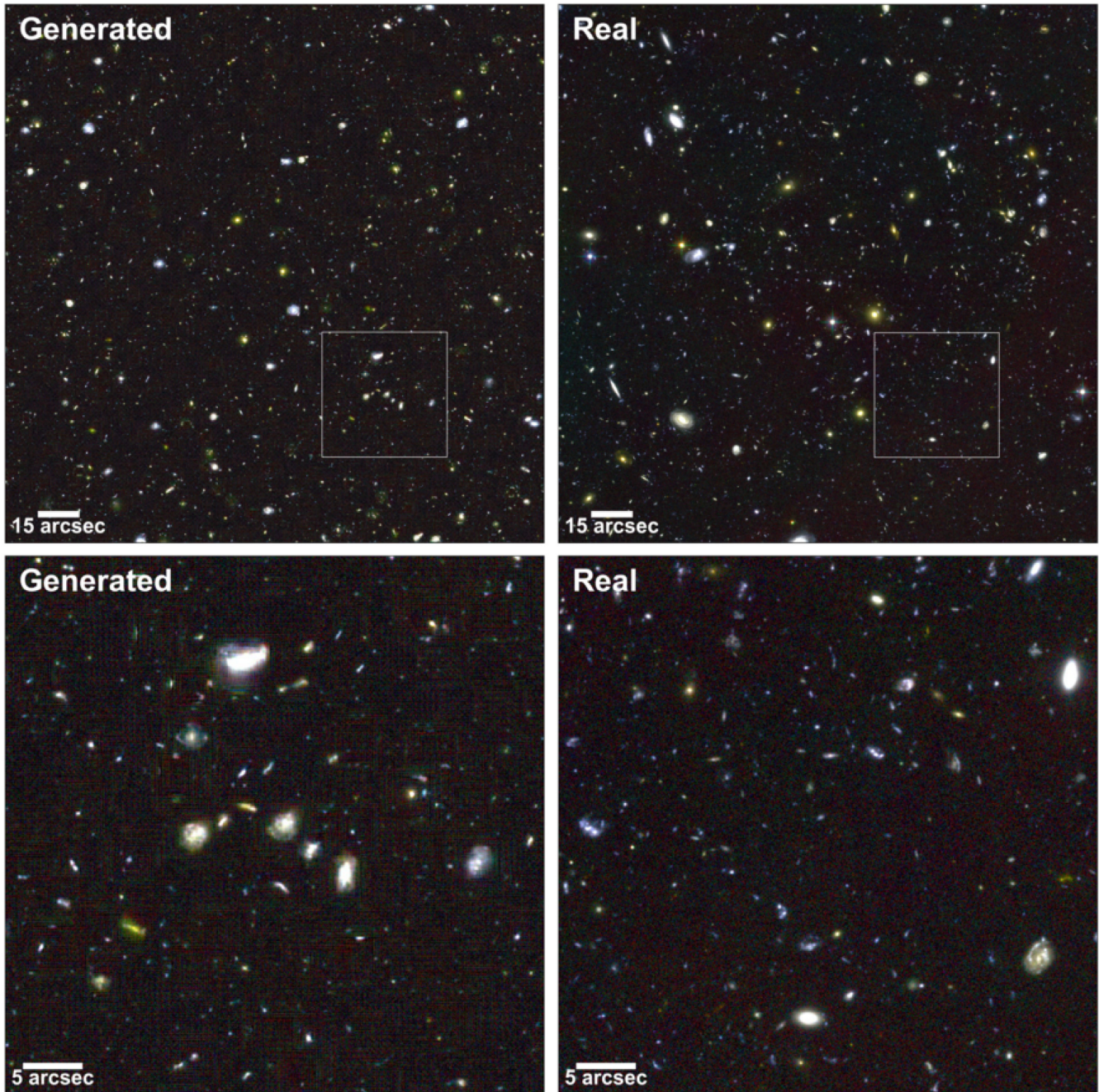


Figure 2. Example of a generated deep field. The top left panel shows a generated deep field with the RGB channels corresponding to the $F814W$, $F775W$, and $F606W$ bands. The size of the generated image was set to 3.2 arcmin, spanning the full XDF (top right), which has a $60 \text{ mas pixel}^{-1}$ scale. The lower panels show 45 arcsec zoom-ins of the generated and real fields.

28 mag (approximately the completeness limit of the catalogue for our detection criteria) are 0.16, 0.63, and 0.75, indicating that we cannot say with confidence that the distributions are drawn from different parent populations, and are therefore statistically similar.

4 DISCUSSION AND CONCLUSIONS

While the SGAN can produce arbitrarily sized simulations of the XDF, there are clear limitations of this technique. The primary limitation is that the generated images are of course totally biased

to the training set. The consequence is that a generated image much larger than the training data will not be cosmologically representative. This is simply because the XDF probes a volume too small to contain rare objects such as, for example, clusters of galaxies and the generator cannot produce examples of objects it has not seen. This problem is simple to alleviate by using a much larger and representative training set. In the real Universe, the positions of galaxies are correlated across a wide range of angular scales, due to the presence of cosmological large-scale structure. Indeed, due to gravitational lensing, even the shapes of galaxies can be

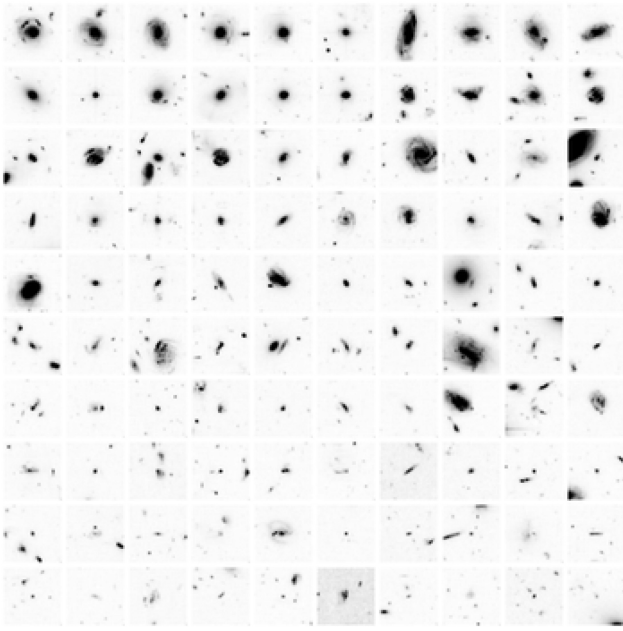


Figure 3. Thumbnail images of real and generated ‘galaxies’. Each image is 7.2 arcsec across and shows the $F775W$ channel on an identical linear grey-scale for 100 sources selected in the range 21–26 mag. 50 targets are selected from each of the real and generated catalogues and displayed in random order in row-wise ascending order of magnitude.

correlated from cosmic shear, and this will not be well captured by the SGAN.

Another limitation is that this clustering information is not present in the generated images, although correlations on the scales of the crop size will be somewhat preserved.

GANs also have some architectural drawbacks. It can be difficult to train a GAN stably; we were not able to achieve stable training with more than three photometric bands. This issue could be resolved with more regularization: adding dropout (Srivastava et al. 2014), or spectral normalization (Miyato et al. 2018) could allow one to generate additional bands, although at the cost of a longer training time, and a larger memory footprint during training.

An advantage of this technique is that it is empirically driven since the data itself is used as the model. By training directly on imaging data the SGAN simultaneously encodes information about the instrumental response as well as the ‘galaxy’ population, thus circumventing the need to make modelling assumptions about either, as is the case in other approaches. The corollary to this is that in the current approach we cannot disentangle the instrument response from the underlying galaxy population, although one could envision approaches to tackle this challenge. For example, a method similar to that used by Schawinski et al. (2017), where a GAN is used to remove noise in astronomical imagery, could be employed. Specifically, a GAN with a UNet-like (Ronneberger, Fischer & Brox 2015) generator would learn a transformation between an image with an entangled instrument response, and one without. Also, since the model is entirely data driven, it is highly dependent on the quality and quantity of the training data.

Recent advances in GAN training techniques have resulted in impressively high fidelity image output (Karras et al. 2017; Karras et al. 2018; Zhang et al. 2018; Brock et al. 2019; Karnewar et al. 2019). Similar methods could be implemented to improve the quality of the generated deep fields. The same methods could also reduce the occurrence of artefacts. Training on a larger set of imaging data, with an increased batch size (Brock et al. 2019), may also produce more representative simulations on a deeper network. The addition of a GAN architecture that allows for control over the output, such as InfoGAN (Chen et al. 2016) or Conditional GAN (Mirza & Osindero 2014), could also be useful when creating mock surveys because they would allow control over the frequency of particular objects of interest, or over the make-up of background noise.

This method could be used to generate at scale entirely artificial, but realistic, image realizations for the design, development, and exploitation of new surveys. For example, one could assemble large training sets for instance segmentation and classification of galaxies. In the early stages of a new survey, relatively small amounts of data could be collected, but then expanded to a level useful for training deep learning models using the generative method described here. Segmentation and classification algorithms could then be trained on the generated data, and then applied to new data, allowing far faster deployment of astronomical deep learning algorithms than would

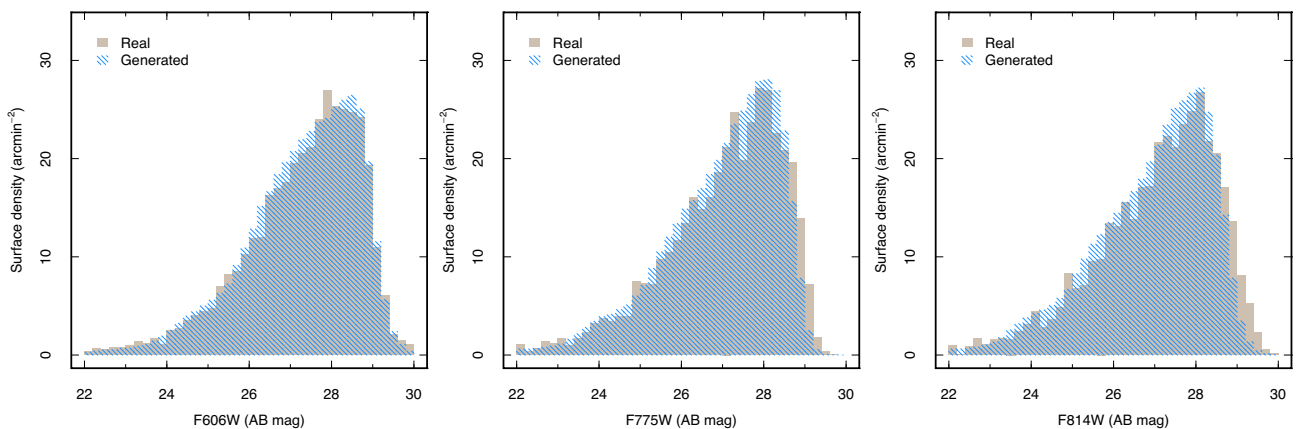


Figure 4. The photometric properties of galaxies detected in the real XDF and ensemble of generated images. The histograms show the magnitude distributions of the galaxies detected in the band-merged image and measured in each of the $F606W$, $F775W$, and $F814W$ bands. The photometric zero-points applied to the generated data is identical to the zero-points calibrated for the real data.

otherwise be possible, potentially accelerating the exploitation of new survey data.

ACKNOWLEDGEMENTS

MJS acknowledges the hospitality of Queen's University during much of this work. JEG is supported by the Royal Society. We gratefully acknowledge the support of NVIDIA Corporation with the donation of the Tesla K40c Graphics Processing Unit used for this research, which also made use of the University of Hertfordshire high-performance computing facility (<http://uhhpc.herts.ac.uk>). The Spatial Generative Adversarial Network code can be obtained at <https://github.com/Smith42/XDF-GAN>. The 7.6-billion pixel version of the generated eXtreme Deep Field can be viewed at <http://star.herts.ac.uk/~jgeach/gdf.html>.

REFERENCES

- Bergé J., Gamper L., Réfrégier A., Amara A., 2013, *Astron. Comput.*, 1, 23
- Bertin E., 2009, *Mem. Soc. Astron. Ital.*, 80, 422
- Bertin E., Arnouts S., 1996, *A&AS*, 117, 393
- Brock A., Donahue J., Simonyan K., 2019, 7th International Conference on Learning Representations, ICLR 2019, Conference Track Proceedings. New Orleans, LA, USA
- Chen X., Duan Y., Houthoofd R., Schulman J., Sutskever I., Abbeel P., 2016, in Lee D., Sugiyama M., Luxburg U., Guyon I., Garnett R., eds, Conference on Neural Information Processing Systems, NIPS 2016, Vol. 29. Curran Associates, Inc, Red Hook, NY, USA, p. 2172
- Clevert D., Unterthiner T., Hochreiter S., 2016, 4th International Conference on Learning Representations, ICLR 2016, Conference Track Proceedings. San Juan, Puerto Rico
- Cole S., Hatton S., Weinberg D. H., Frenk C. S., 1998, *MNRAS*, 300, 945
- Dobke B. M., Johnston D. E., Massey R., High F. W., Ferry M., Rhodes J., Vanderveld R. A., 2010, *PASP*, 122, 947
- Goodfellow I., Pouget-Abadie J., Mirza M., Xu B., Warde-Farley D., Ozair S., Courville A., Bengio Y., 2014, in Ghahramani Z., Welling M., Cortes C., Lawrence N., Weinberger K., eds, Conference on Neural Information Processing Systems, NIPS 2014, Vol. 27. Curran Associates, Inc, Red Hook, NY, USA, p. 2672
- Illingworth G. D. et al., 2013, *ApJS*, 209, 6
- Jetchev N., Bergmann U., Vollgraf R., 2016, CoRR, abs/1611.08207
- Jolicœur-Martineau A., 2019, 7th International Conference on Learning Representations, ICLR 2019, Conference Track Proceedings. New Orleans, LA, USA
- Jonsson P., 2006, *MNRAS*, 372, 2
- Karnewar A., Wang O., Iyengar R. S., 2019, CoRR, abs/1903.06048
- Karras T., Aila T., Laine S., Lehtinen J., 2017, CoRR, abs/1710.10196
- Karras T., Laine S., Aila T., 2018, CoRR, abs/1812.04948
- Kingma D. P., Ba J., 2015, 3rd International Conference on Learning Representations, ICLR 2015, Conference Track Proceedings. San Diego, CA, USA
- Kodali N., Abernethy J. D., Hays J., Kira Z., 2017, CoRR, abs/1705.07215
- Lin Z., Khetan A., Fanti G., Oh S., 2018, in Bengio S., Wallach H., Larochelle H., Grauman K., Cesa-Bianchi N., Garnett R., eds, Conference on Neural Information Processing Systems, NIPS 2018, Vol. 31. Curran Associates, Inc, Red Hook, NY, USA, p. 1498
- Maas A. L., Hannun A. Y., Ng A. Y., 2013, Proc. International Conference on Machine Learning, Vol. 30. p. 3
- Mandelbaum R., Hirata C. M., Leauthaud A., Massey R. J., Rhodes J., 2012, *MNRAS*, 420, 1518
- Mirza M., Osindero S., 2014, CoRR, abs/1411.1784
- Miyato T., Kataoka T., Koyama M., Yoshida Y., 2018, 6th International Conference on Learning Representations, ICLR 2018, Conference Track Proceedings. Vancouver, BC, Canada
- Mordido G., Yang H., Meinel C., 2018, CoRR, abs/1807.11346
- Nanni L., Ghidoni S., Brahmam S., 2018, *Appl. Comput. Informatics*, in press
- Obreschkow D., Klöckner H.-R., Heywood I., Levrier F., Rawlings S., 2009, *ApJ*, 703, 1890
- Paul R., Hall L., Goldgof D., Schabath M., Gillies R., 2018, 2018 International Joint Conference on Neural Networks (IJCNN). IEEE
- Peterson J. R. et al., 2015, *ApJS*, 218, 14
- Radford A., Metz L., Chintala S., 2016, 4th International Conference on Learning Representations, ICLR 2016., Conference Track Proceedings. San Juan, Puerto Rico
- Robbins H., Monro S., 1951, *Ann. Math. Stat.*, 22, 400
- Ronneberger O., Fischer P., Brox T., 2015, Medical Image Computing and Computer-Assisted Intervention - MICCAI 2015-18th International Conference Proceedings, Part III. Springer, Berlin, p. 234
- Roth K., Lucchi A., Nowozin S., Hofmann T., 2017, in Guyon I., Luxburg U., Bengio S., Wallach H., Fergus R., Vishwanathan S., Garnett R., eds, Conference on Neural Information Processing Systems, NIPS 2017. Curran Associates, Inc., Red Hook, NY, USA, p. 2018
- Rowe B. T. P. et al., 2015, *Astron. Comput.*, 10, 121
- Salimans T., Goodfellow I. J., Zaremba W., Cheung V., Radford A., Chen X., 2016, CoRR, abs/1606.03498
- Schawinski K., Zhang C., Zhang H., Fowler L., Santhanam G. K., 2017, *MNRAS*, 467, L110
- Schaye J. et al., 2015, *MNRAS*, 446, 521
- Scoville N. et al., 2007, *ApJS*, 172, 1
- Snyder G. F., Lotz J. M., Rodriguez-Gomez V., Guimarães R. d. S., Torrey P., Hernquist L., 2017, *MNRAS*, 468, 207
- Srivastava N., Hinton G. E., Krizhevsky A., Sutskever I., Salakhutdinov R., 2014, *J. Mach. Learn. Res.*, 15, 1929
- Trayford J. W. et al., 2017, *MNRAS*, 470, 771
- Vogelsberger M. et al., 2014, *MNRAS*, 444, 1518
- Wang Y., Zhang L., van de Weijer J., 2016, CoRR, abs/1612.00991
- Williams R. E. et al., 1996, *AJ*, 112, 1335
- Zhang H., Goodfellow I. J., Metaxas D. N., Odena A., 2018, CoRR, abs/1805.08318

This paper has been typeset from a $\text{\TeX}/\text{\LaTeX}$ file prepared by the author.