
DeepBench: Simulated Benchmark Datasets for Astronomical Analysis

Ashia Lewis^{*1} Margaret Voetberg^{*1} Brian Nord¹²³ Craig Jones⁴ Renée Hložek⁵ Aleksandra Ćiprijanović¹
Gabriel Perdue¹

Abstract

The astronomy community is experiencing a lack of benchmark datasets tailored towards machine learning and computer vision problems. The overall goal of this software is to fill this need. We introduce the python library `DeepBench`, which is designed to provide a method of producing highly reproducible datasets at varying levels of complexity, size, and content. The software includes simulation of basic geometric shapes and astronomical structures, such as stars and elliptical galaxies, as well as tools to collect and store the dataset for consumption by a machine learning algorithm. We also present a trained ResNet-50 model as an illustration of the expected use of the software as a benchmarking tool for different architectures' suitability to scientifically motivated problems. We envision this tool to be useful in a suite of contexts at the intersection of astronomy and machine learning. For example, this could be useful for those new to machine learning principles and software, as a way to build their skills and tools with a toy-model data set that looks like astronomical data. Also, experts can use this tool to build simple data sets that allow them to check their models. Finally, the geometric/polygon images can be used as a highly simplified version of astronomical objects: this could be used for addressing a spectrum of problems from object classification to deblending. The GitHub repository containing the source code for `DeepBench` can be found here: <https://github.com/deepskies/DeepBench>.

^{*}Equal contribution ¹Fermi National Accelerator Laboratory, P.O. Box 500, Batavia, IL 60510, USA ²University of Chicago, Chicago, IL 60637, USA ³Kavli Institute for Cosmological Physics, University of Chicago, Chicago, IL 60637, USA ⁴Whiting School of Engineering, Johns Hopkins University, Baltimore, MD 21218 ⁵Dunlap Institute for Astronomy and Astrophysics, University of Toronto, Toronto ON M5S 3H4, Canada. Correspondence to: Ashia Lewis <atlewis@fnal.gov>, Margaret Voetberg <margiev@fnal.gov>.

1. Motivation

`DeepBench` was designed to have three major elements. First, it is intended to be extremely easy to use, especially for those new to machine learning and even computing. Second, it can be used to produce computer vision data sets with varying levels of complexity to help the diagnose machine learning model errors – for users of any level of experience in machine learning. Third, it is intended as a generator of benchmark data sets

1.1. Beginner-Friendly Software

One of the main motivations of this project was the production of a tool to gently introduce new practitioners of machine learning to techniques for feature engineering and model tuning commonly encountered in the field, using datasets that are fast to integrate with out-of-the-box machine learning frameworks. This is accomplished by including methods to catalog all produced images and provide a description of all generated objects included in the images.

To avoid unnecessary complexity in execution and use of the program, the software was designed with the philosophy of being easy enough for someone with minimal programming experience to use. `DeepBench` is packaged with multiple examples of usage and different types of configuration templates for producing default datasets, along with making available multiple modes of execution – either as a package used in conjunction with jupyter-style notebooks or through command line interfaces.

The range of objects also lends itself toward flexibility for beginners – from datasets that are “easy” to learn (such as balanced binary classification of distinct classes) to more complicated challenges like regression or simulation: this simultaneously makes `DeepBench` useful as a learning/teaching tool, and as a vehicle to help climb a ladder of complexity for experts in testing their methods.

1.2. Diagnosis for Experiments and Models

The software is also intended for use as a diagnosis tool to help users identify model weaknesses and assess issues during machine learning model training. `DeepBench` makes it easy to incrementally increase and decrease the data ob-

ject complexity (e.g., remove noise, decrease the number of classes, increase the class balance), which is helpful when seeking weak points in a machine learning model architecture or training schedule.

More explicitly, in the case of a task like gravitational lensing classification, it would be possible to move from a dataset containing real images, to a simulated dataset containing a similar distribution of objects as the training set, and incrementally remove elements such as noise from simulated images.

1.3. Transfer Learning and Network Benchmarking

The astronomy community is long familiar with short-term data challenges that use benchmark data sets – e.g. the GREAT challenges for weak gravitational lensing (Bridle et al., 2009), the finding challenges for strong lensing (Metcalfe et al., 2019), and the supernova classification PLAS-TICC challenge (Hložek et al., 2020). However, there do not yet exist benchmark data sets that permit persistent, long-term model comparisons and testings that are particularly relevant for astronomy and physics.

Most persistent benchmark datasets for computer vision – e.g., MNIST (Deng, 2012), CIFAR (Krizhevsky et al.) – are not designed with the intent of developing models for astronomy research, as their focus tends to be replicating human vision and identification techniques. This leads to 1) a lack of accuracy baselines for networks that are applicable for astronomy problems, and 2) a lack of pre-trained weights for large architectures that can be used in transfer and semi-supervised learning problems.

DeepBench aims to help fill this need by providing the capability to generate simulated datasets that contain distributions of objects that more closely match the type of datasets used in more realistic studies: studies relying on pre-built architectures need not retrain all weights from scratch. Training techniques that use transfer learning often have drastically decreased training run times, as well as provide out-of-the-box weights that can perform on related astronomy problems.

2. Related Work

Benchmarking and dataset generation is heavily used in the field of Machine Learning. Works that are most closely related to the work described here include SHAPES (Wu et al., 2016) for its use of collections of geometric objects as a benchmark with varying levels of complexity, and deepLenstronomy (Morgan et al., 2021) for its packaging of the strong lensing framework lenstronomy (Bridle et al., 2009) into tools with ease of use and reproducibility included in their design.

3. Methods

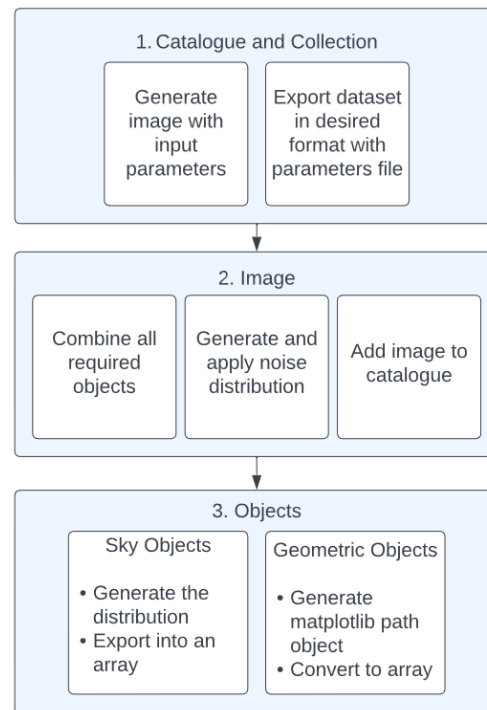


Figure 1. The DeepBench workflow. Dataset parameters (type of objects in each image, qualities of each object), are passed by the Catalogue module (1) by the user. These are used the Image module (2) to call and construct each Object (3) that will be included in the final image. This module also combines called objects into one image, and “image level” noise is applied, choosing from either Gaussian or Poisson distributions. The parameters of each object within the image are saved by Catalogue (1), along with the image itself.

3.1. Modules

3.1.1. GEOMETRIC OBJECTS

Geometric objects are all generated by using the `matplotlib` library (Hunter, 2007), and are able to either be left as solid two-dimensional shapes or of an outline with a varying thickness. Shapes include rectangles, n -sided regular polygons, arcs, straight lines, ellipses, and circles. Shapes are limited to two-color rendering, such that all shapes are composed of 0 and 1 values within an array. They can be combined in multiple ways using the “Image” module to produce composite images with both different geometric objects and astronomical objects.

3.1.2. ASTRONOMICAL OBJECTS

Astronomical objects offer simplified renderings of common profiles found in astronomical data sets. These objects can be either 1D or 2D models, depending on the object type. They can also have parameters that are either randomized or manually specified by the user via a configuration file.

One-Dimensional Models The *N-Body* object is currently the sole one-dimensional model available in the alpha release. The N-Body model is output as a set of NumPy (Harris et al., 2020) arrays containing coordinates representing the path of the points produced, along with the kinetic and potential energy produced over the specified duration.

Two-Dimensional Models Two-dimensional models are representations of astronomical objects commonly found in data sets used for galaxy morphology classification. All objects also come with the option to append various levels of Gaussian and Poisson noise, and are output as NumPy arrays. The following are the objects currently available via alpha release.

Star The *Star* object is created using the Moffat distribution provided by the AstroPy (Astropy Collaboration et al., 2018) modeling library.

Elliptical Galaxy The (*Elliptical*) *Galaxy* object is created using the Sérsic profile provided by the AstroPy modelling library.

Spiral Galaxy The profile of the *Spiral Galaxy* object is created by simulating the function used to produce a logarithmic spiral (Ringermacher & Mead, 2009), along with reproducing the following relationship:

$$r(\phi) = A/\log[B * \tan(\phi/2N)] \quad (1)$$

Where A defines the scale of the galaxy, B controls spiral pitch, N is the 'winding number', with defines the tightness of the arms, and ϕ is the angle of parallelism.

3.1.3. IMAGE

The image module allows users to concatenate various shape or astronomical objects within a matplotlib meshgrid object in order to simulate the profiles and shape distributions commonly seen in images used in more complex astronomical data sets.

Three distinct image types are available - sky images, lensing images, and geometric images. Sky images are composed of any combination of user-specified galaxy and star objects, while lensing images are randomized combinations of arc and star objects. Lastly, geometric images are can-

vased assortments of any of the individual geometric shape objects available.

3.1.4. CATALOGUE

The Catalogue module allows users to specify the size, contents, and output directory of a data set composed of DeepBench's available image types or individual object images.

Catalogues can either be entirely randomized, with parameters of the included image being randomly chosen, or have parameters specified by the user at various levels of granularity.

The only required argument in the creation of Catalogue objects are the output directory and image type. Within the Catalogue module also exists a Collection class, which concatenates various catalogues of varying image types into one data set.

Catalogues and collections can be output in the formats .jpeg, .npy, or .h5.

4. Example Output

4.1. Geometric Shape Image

With the exception of the image dimensions, geometric images are also provided randomized default values for each of the available parameters in every available shape. The default image dimensions are defined as 256-by-256 pixels. Like astronomical objects, geometric shapes can be produced as individual objects or as a plot of concatenated shapes. Although each geometric object has distinct parameters necessary for the creation of a shape, the most commonly held configurable variables for each shape are the center, radius, and angle. See Figure 2.

4.2. Astronomical Object Images

All parameters of the astronomical objects are randomized at default with the exception of the image dimensions variable, which is set to a default value of 28-by-28 pixels. The noise levels for the point-spread and Gaussian noise can also be specified for every astronomical image type.

Elliptical Galaxy Image Configurable parameters for the Elliptical Galaxy object include the following: image dimensions, center of the object, amplitude, and radius, along with the ellipticity and theta value (see Figure 6).

Spiral Galaxy Image Configurable parameters to produce the Spiral Galaxy object include the following: image dimensions, center of the object, elliptic core size, number of arms, radius, and amplitude (see Figure 3).

Star Image Configurable parameters for Stars include: image dimensions, noise, radius, center, and amplitude (see Figure 4).

Lensing Image Configurable parameters for the Lensing object are composed of the available variables used in each of its separate objects, namely the arc geometric object and star astronomical object (see Figure 5).

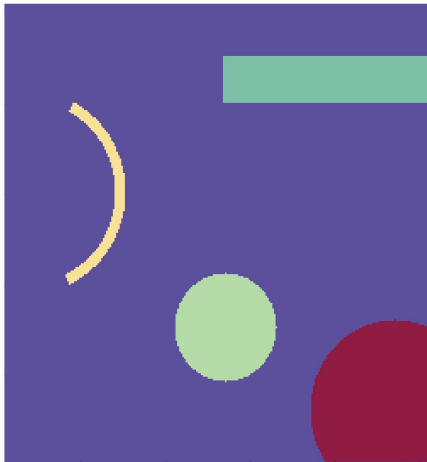


Figure 2. A completely randomized collection of geometric shapes concatenated onto a canvas object. Each shape is colored differently for illustrative purposes.

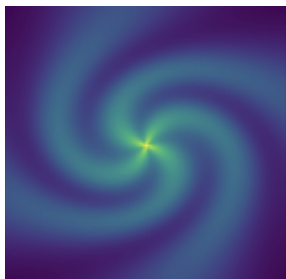


Figure 3. A spiral galaxy with the number of arms defined as four and the Gaussian noise set to 0.0.

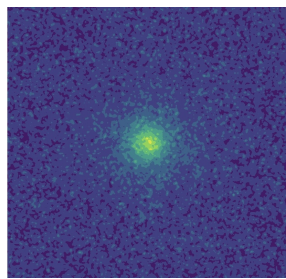


Figure 4. A star with a noise level defined as 0.50. All other parameters are randomized by default.

5. Executed Benchmarks

In the interest of providing executed benchmarks, multiple ResNet-50 models (He et al., 2015) were trained for five different classification tasks. The ResNet-50 architecture was selected for both its high performance for image classification tasks, overall popularity, and existing benchmark runs on many other computer vision datasets.

The architectures were trained using the pyTorch framework (Paszke et al., 2019), using three different

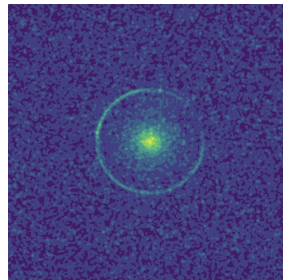


Figure 5. A lensing image with the arc’s endpoint defined as 150.0 degrees.

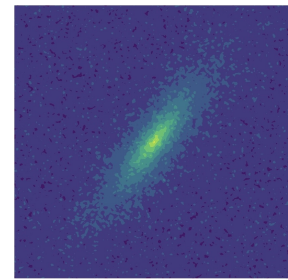


Figure 6. An elliptical galaxy image using totally randomized default parameters.

DeepBench variants: (1) a dataset containing “Sky” images, containing multiple structures per image; (2) a dataset containing geometric shapes, with multiple shapes per image; (3) individual astronomical structures, with one star, galaxy, or spiral galaxy per image. Each dataset contained 10,000 total images, split 60/20/20 between training, validation, and testing distributions, respectively.

All models were trained with a batch size of 64, learning rate of 0.524, and weight decay rate of 0.00125.

Table 1. Summary of results for the benchmark trained ResNet models.

Dataset	Accuracy
Sky Images	95.7
Geometric Images	97.3
Astronomy Objects	97.1

For the transfer learning trial, a ResNet-50 model without prior weights was trained on the dataset of randomly produced sky images used in the previous classification task, while another model was produced without any defined weights. The comparison dataset used for the transfer learning was composed of a collection of 5000 images from the GalaxyZoo2 (Willett et al., 2013) dataset. The included class labels were cigar-shaped smooth, ring-shaped, merging, and spiral, with each class having approximately equal representation. All layers of the model trained on the DeepBench dataset, with the exception of the input and output layers, were then transferred to a separate model used to classify the GalaxyZoo2 images. The model without pre-defined weights was solely trained on GalaxyZoo2 images.

The transfer-learned model was able to reach the accuracy achieved by the non-transfer learning model after approximately 10,000 iterations, as seen in Figure 7.

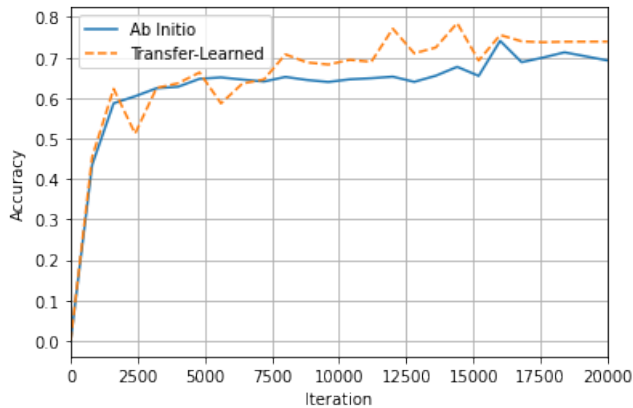


Figure 7. Summary of the training accuracy for the ab initio data model (blue, solid) and the transfer-learned model (orange, dotted) for the transfer learning trial.

Table 2 presents the accuracies of each approach after 20,000 iterations. Although training did not reach convergence, the significant difference in produced accuracy becomes clear early on.

Table 2. Summary of results producing during the transfer-learning trial for 16,000 iterations.

Approach	Accuracy
Transfer-Learned Model	73.1
Ab Initio Model	69.3

6. Conclusion and Outlook

DeepBench fulfills the demand for simple, seamlessly-integrated benchmark datasets for common astronomy-focused machine learning tasks. Due to its focus, it lends itself towards being a base for transfer-learning problems in so that higher accuracies can be reached in shorter training times. We envision pre-trained models based on DeepBench being used across a wide range of experiments and projects. We hope that using this method of benchmarking allows for easier communication and comparison of results, and encouraging collaboration between different research projects working on similar problems.

By designing a system that is simultaneously configurable yet undemanding, DeepBench’s straight-forward command-line tool allows users to quickly produce benchmarks personalized to fit their particular testing needs. Interfaces are provided for objects that range in complexity from simple geometric shapes to more complex simulations of common astronomical profiles, with no defined limit to the amount of data that can be produced. The straightforward nature also encourages new practitioners, with simple cases for classification and regression available.

With the beta release of DeepBench, more attention will be paid to expanding the number of objects available in the one- and three-dimensional modelling modules.

7. Acknowledgements

We acknowledge the Deep Skies Lab as a community of multi-domain experts and collaborators who’ve facilitated an environment of open discussion, idea-generation, and collaboration. This community was important for the development of this project.

This manuscript has been authored by Fermi Research Alliance, LLC under Contract No. DE-AC02-07CH11359 with the U.S. Department of Energy, Office of Science, Office of High Energy Physics.

References

Astropy Collaboration, Price-Whelan, A. M., Sipőcz, B. M., Günther, H. M., Lim, P. L., Crawford, S. M., Conseil, S., Shupe, D. L., Craig, M. W., Dencheva, N., Ginsburg, A., VanderPlas, J. T., Bradley, L. D., Pérez-Suárez, D., de Val-Borro, M., Aldcroft, T. L., Cruz, K. L., Robitaille, T. P., Tollerud, E. J., Ardelean, C., Babej, T., Bach, Y. P., Baccetti, M., Bakanov, A. V., Bamford, S. P., Barentsen, G., Barmby, P., Baumbach, A., Berry, K. L., Biscani, F., Boquien, M., Bostroem, K. A., Bouma, L. G., Brammer, G. B., Bray, E. M., Breytenbach, H., Buddelmeijer, H., Burke, D. J., Calderone, G., Cano Rodríguez, J. L., Cara, M., Cardoso, J. V. M., Cheedella, S., Copin, Y., Corrales, L., Crichton, D., D’Avella, D., Deil, C., Depagne, É., Dietrich, J. P., Donath, A., Droettboom, M., Earl, N., Erben, T., Fabbro, S., Ferreira, L. A., Finethy, T., Fox, R. T., Garrison, L. H., Gibbons, S. L. J., Goldstein, D. A., Gommers, R., Greco, J. P., Greenfield, P., Groener, A. M., Grollier, F., Hagen, A., Hirst, P., Homeier, D., Horton, A. J., Hosseinzadeh, G., Hu, L., Hunkeler, J. S., Ivezić, Ž., Jain, A., Jenness, T., Kanarek, G., Kendrew, S., Kern, N. S., Kerzendorf, W. E., Khvalko, A., King, J., Kirkby, D., Kulkarni, A. M., Kumar, A., Lee, A., Lenz, D., Littlefair, S. P., Ma, Z., Macleod, D. M., Mastroiello, M., McCully, C., Montagnac, S., Morris, B. M., Mueller, M., Mumford, S. J., Muna, D., Murphy, N. A., Nelson, S., Nguyen, G. H., Ninan, J. P., Nöthe, M., Ogaz, S., Oh, S., Parejko, J. K., Parley, N., Pascual, S., Patil, R., Patil, A. A., Plunkett, A. L., Prochaska, J. X., Rastogi, T., Reddy Janga, V., Sabater, J., Sakurikar, P., Seifert, M., Sherbert, L. E., Sherwood-Taylor, H., Shih, A. Y., Sick, J., Silbiger, M. T., Singanamalla, S., Singer, L. P., Sladen, P. H., Sooley, K. A., Sornarajah, S., Streicher, O., Teuben, P., Thomas, S. W., Tremblay, G. R., Turner, J. E. H., Terrón, V., van Kerkwijk, M. H., de la Vega, A., Watkins, L. L., Weaver, B. A., Whitmore, J. B., Woillez,

- J., Zabalza, V., and Astropy Contributors. The Astropy Project: Building an Open-science Project and Status of the v2.0 Core Package. , 156(3):123, September 2018. doi: 10.3847/1538-3881/aabc4f.
- Bridle, S., Gill, M., Heavens, A., Heymans, C., High, F. W., Hoekstra, H., Jarvis, M., Kirk, D., Kitching, T., Kneib, J.-P., Kuijken, K., Shawe-Taylor, J., Lagatutta, D., Mandelbaum, R., Massey, R., Mellier, Y., Moghaddam, B., Moudren, Y., Nakajima, R., Paulin-Henriksson, S., Pires, S., Rassat, A., Amara, A., Refregier, A., Rhodes, J., Schrabback, T., Semboloni, E., Shmakova, M., van Waerbeke, L., Witherick, D., Voigt, L., Wittman, D., Applegate, D., Balan, S. T., Berge, J., Bernstein, G., Dahle, H., and Erben, T. Handbook for the GREAT08 challenge: An image analysis competition for cosmological lensing. *The Annals of Applied Statistics*, 3(1), mar 2009. doi: 10.1214/08-aoas222. URL <https://doi.org/10.1214%2F08-aoas222>.
- Deng, L. The mnist database of handwritten digit images for machine learning research. *IEEE Signal Processing Magazine*, 29(6):141–142, 2012.
- Harris, C. R., Millman, K. J., van der Walt, S. J., Gommers, R., Virtanen, P., Cournapeau, D., Wieser, E., Taylor, J., Berg, S., Smith, N. J., Kern, R., Picus, M., Hoyer, S., van Kerkwijk, M. H., Brett, M., Haldane, A., del Río, J. F., Wiebe, M., Peterson, P., Gérard-Marchant, P., Sheppard, K., Reddy, T., Weckesser, W., Abbasi, H., Gohlke, C., and Oliphant, T. E. Array programming with NumPy. *Nature*, 585(7825):357–362, September 2020. doi: 10.1038/s41586-020-2649-2. URL <https://doi.org/10.1038/s41586-020-2649-2>.
- He, K., Zhang, X., Ren, S., and Sun, J. Deep residual learning for image recognition. *CoRR*, abs/1512.03385, 2015. URL <http://arxiv.org/abs/1512.03385>.
- Hložek, R., Ponder, K. A., Malz, A. I., Dai, M., Narayan, G., Ishida, E. E. O., Allam, T., Bahmanyar, A., Biswas, R., Galbany, L., Jha, S. W., Jones, D. O., Kessler, R., Lochner, M., Mahabal, A. A., Mandel, K. S., Martínez-Galarza, J. R., McEwen, J. D., Muthukrishna, D., Peiris, H. V., Peters, C. M., and Setzer, C. N. Results of the photometric lsst astronomical time-series classification challenge (plasticc), 2020. URL <https://arxiv.org/abs/2012.12392>.
- Hunter, J. D. Matplotlib: A 2d graphics environment. *Computing in Science & Engineering*, 9(3):90–95, 2007. doi: 10.1109/MCSE.2007.55.
- Krizhevsky, A., Nair, V., and Hinton, G. Cifar-10 (canadian institute for advanced research). URL <http://www.cs.toronto.edu/~kriz/cifar.html>.
- Metcalf, R. B., Meneghetti, M., Avestruz, C., Bellagamba, F., Bom, C. R., Bertin, E., Cabanac, R., Courbin, F., Davies, A., Decenciè re, E., Flamary, R., Gavazzi, R., Geiger, M., Hartley, P., Huertas-Company, M., Jackson, N., Jacobs, C., Jullo, E., Kneib, J.-P., Koopmans, L. V. E., Lanusse, F., Li, C.-L., Ma, Q., Makler, M., Li, N., Lightman, M., Petrillo, C. E., Serjeant, S., Schäfer, C., Sonnenfeld, A., Tagore, A., Tortora, C., Tuccillo, D., Valentín, M. B., Velasco-Forero, S., Kleijn, G. A. V., and VERNARDOS, G. The strong gravitational lens finding challenge. *Astronomy & Astrophysics*, 625:A119, may 2019. doi: 10.1051/0004-6361/201832797. URL <https://doi.org/10.1051%2F0004-6361%2F201832797>.
- Morgan, R., Nord, B., Birrer, S., Lin, J. Y.-Y., and Poh, J. deepLenstronomy: A dataset simulation package for strong gravitational lensing. *Journal of Open Source Software*, 6(58):2854, 2021. doi: 10.21105/joss.02854. URL <https://doi.org/10.21105/joss.02854>.
- Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., Desmaison, A., Kopf, A., Yang, E., DeVito, Z., Raison, M., Tejani, A., Chilamkurthy, S., Steiner, B., Fang, L., Bai, J., and Chintala, S. Pytorch: An imperative style, high-performance deep learning library. In Wallach, H., Laroche, H., Beygelzimer, A., d'Alché-Buc, F., Fox, E., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems 32*, pp. 8024–8035. Curran Associates, Inc., 2019. URL <http://papers.neurips.cc/paper/9015-pytorch-an-imperative-style-high-performance-deep-learning-library.pdf>.
- Ringermacher, H. I. and Mead, L. R. A new formula describing the scaffold structure of spiral galaxies. *Monthly Notices of the Royal Astronomical Society*, 397(1):164–171, jul 2009. doi: 10.1111/j.1365-2966.2009.14950.x. URL <https://doi.org/10.1111%2Fj.1365-2966.2009.14950.x>.
- Willett, K. W., Lintott, C. J., Bamford, S. P., Masters, K. L., Simmons, B. D., Casteels, K. R. V., Edmondson, E. M., Fortson, L. F., Kaviraj, S., Keel, W. C., Melvin, T., Nichol, R. C., Raddick, M. J., Schawinski, K., Simpson, R. J., Skibba, R. A., Smith, A. M., and Thomas, D. 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, sep 2013. doi: 10.1093/mnras/stt1458. URL <https://doi.org/10.1093%2Fmnras%2Fstt1458>.
- Wu, Q., Teney, D., Wang, P., Shen, C., Dick, A. R., and van den Hengel, A. Visual question answering: A survey

of methods and datasets. *CoRR*, abs/1607.05910, 2016.

URL <http://arxiv.org/abs/1607.05910>.