
Generating Astronomical Spectra from Photometry with Conditional Diffusion Models

Lars Doorenbos
University of Bern
Bern, Switzerland
lars.doorenbos@unibe.ch

Stefano Cavuoti
Astronomical Observatory of Capodimonte
Naples, Italy
stefano.cavuoti@inaf.it

Giuseppe Longo
University Federico II
Naples, Italy
giuseppe.longo@unina.it

Massimo Brescia
University Federico II
Naples, Italy
massimo.brescia@unina.it

Raphael Sznitman
University of Bern
Bern, Switzerland
raphael.sznitman@unibe.ch

Pablo Márquez-Neila
University of Bern
Bern, Switzerland
pablo.marquez@unibe.ch

Abstract

A trade-off between speed and information controls our understanding of astronomical objects. Fast-to-acquire photometric observations provide global properties, while costly and time-consuming spectroscopic measurements enable a better understanding of the physics governing their evolution. Here, we tackle this problem by generating spectra directly from photometry, through which we obtain an estimate of their intricacies from easily acquired images. This is done by using multi-modal conditional diffusion models, where the best out of the generated spectra is selected with a contrastive network. Initial experiments on minimally processed SDSS galaxy data show promising results.

1 Introduction

Modern digital multi-band astronomical surveys are producing large volumes of photometric data, and will do so even more in the future. In most cases the scientific exploitation of the photometric data requires a vast amount of ancillary spectroscopic data, which can better characterize the sources. The latter, unfortunately, are much more demanding in terms of telescope time and can be acquired only for a small fraction of the photometric counterparts. To quote an example, the Sloan Digital Sky Survey (SDSS, Ahumada et al. [2020]), which has one of the richest spectroscopic datasets to date, has observed more than one billion objects photometrically, while only about two million objects have been observed together with their spectra. The fact that multi-band photometry may be assimilated to low-resolution spectroscopy, motivates the search for methods capable of automatically guessing the shape of a spectrum, provided a significant number of examples. The possibility to use only photometric data to have a reliable guess of spectra would be useful for many applications, for instance to determine which objects are likely to be the most interesting, and need to be targeted for further investigations Masters et al. [2015].

For our approach to this problem we make use of the paradigm of multi-modal learning. Recently, many multi-modal learning methods have combined all sorts of modalities Bayouhd et al. [2022],

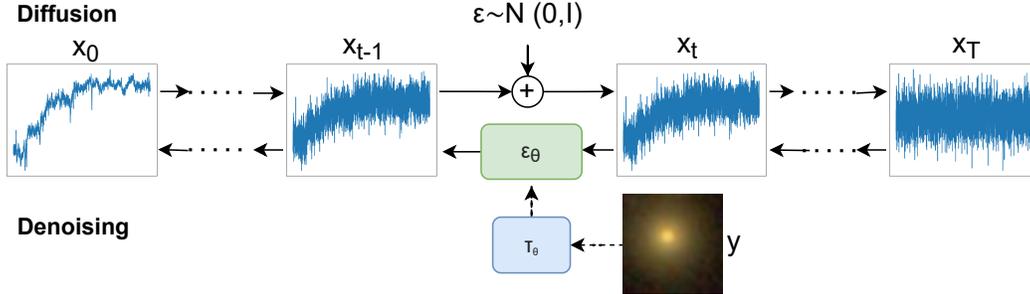


Figure 1: Training of the conditional diffusion model.

such as text-image Radford et al. [2021], text-video Fan et al. [2019], genome-image Taleb et al. [2022], and so on. In astronomy, Wu and Peek [2020] attempted to predict galaxy spectra from images by predicting the latent variables of a variational autoencoder (VAE). A follow-up study Holwerda et al. [2021] aimed to confirm its practical use.

In this work, we similarly attempt to generate galaxy spectra from images, by directly learning the relations between spectroscopic and photometric observations of the same object. In particular, we take inspiration from the text-image models CLIP Radford et al. [2021] and DALL-E Ramesh et al. [2021]. The images generated from text by the generative model DALL-E are ranked using the contrastive model CLIP to find the best fitting samples.¹ We apply this concept to generate spectra from multi-band images, and find the best match with a contrastive network.²

2 Method

Data We extracted 64x64 thumbnails of the three inner SDSS bands (g , r and i) for around 500 000 galaxies from the Sloan Digital Sky Survey Data Release 16 (SDSS DR16, Ahumada et al. [2020]), together with their corresponding spectra. As the SDSS pixel scale is 0.396 arcseconds per pixel, after random cropping, the images cover ± 22.85 arcseconds squared. Since the spectra covered different wavelength (λ) ranges, we discarded all spectra with a minimum $\log \lambda > 3.59$, or a maximum below 3.95. Then, we cropped every spectrum to the range between 3.59 and 3.95 and produced a uniform dataset of 3598-dimensional spectra. We normalized all spectra to the range $[-1, 1]$. In total, this resulted in 496599 image-spectra pairs, which we split into a training set of 491599 and a validation set of 5000 pairs.

2.1 Preliminaries

Denosing diffusion probabilistic models (DDPM) DDPMs, also known as diffusion models, are a class of generative models, that learn the data distribution through the combination of two parts: a diffusion and a denosing process Ho et al. [2020]. The diffusion process adds a certain amount of Gaussian noise to a sample, based on a set variance schedule β_1, \dots, β_T at a timestep $t \in [1, T]$. This gradually transforms a sample x_0 into a standard Gaussian distribution at x_T .

The denosing process aims to reverse this process: given t , it predicts the denosed version x_0 from x_t with an autoencoder neural network ϵ_θ . In practice, this is often learned by predicting the noise instead Ho et al. [2020], Rombach et al. [2022], with the simplified objective

$$L = \mathbb{E}_{x_0, t, \epsilon \sim \mathcal{N}(0, I)} (\| \epsilon - \epsilon_\theta(x_t, t) \|^2). \quad (1)$$

From these processes, we can create new samples by sampling from a Gaussian distribution, then using the diffusion process to go back from x_T to x_0 .

Conditional diffusion models (CDM) While vanilla diffusion models are capable of modelling and sampling from a data distribution, we are interested in obtaining samples for a specific object. This can be achieved by sampling from a conditional distribution instead. This is implemented by

¹ <https://openai.com/blog/dall-e> ² <https://github.com/LarsDoorenbos/generate-spectra>

adding a condition to the input of the noise prediction network. Given a condition y , we project it to a latent embedding using a learnable encoder τ_θ . This gives the loss function

$$L = \mathbb{E}_{x_0, y, t, \epsilon \sim \mathcal{N}(0, \mathbf{I})} (\| \epsilon - \epsilon_\theta(x_t, t, \tau_\theta(y)) \|^2). \quad (2)$$

We experimented with 2 forms of conditioning. The first projects $\tau_\theta(y)$ to the same dimensionality as x_0 , and concatenates them channel-wise. The second follows Rombach et al. [2022], and introduces the condition with cross-attention at multiple layers of the autoencoder.

Contrastive learning Contrastive learning is a framework for self-supervised learning. It optimizes a network by minimizing the latent distance between two views of the same object, while maximizing the distances to the latent codes of the other samples in the batch. This is done by minimizing the contrastive loss for mini-batches of size N Chen et al. [2020],

$$\ell_{i,j} = -\log \frac{\exp(\text{sim}(\mathbf{z}_i, \mathbf{z}_j)/\tau)}{\sum_{k=1}^{2N} \mathbf{1}_{[k \neq i]} \exp(\text{sim}(\mathbf{z}_i, \mathbf{z}_k)/\tau)}, \quad (3)$$

for views i and j , where \mathbf{z} represents their latent representation, τ the temperature, and $\text{sim}(\cdot, \cdot)$ the cosine similarity. In the present work, the two views of an object are its image and its spectrum.

2.2 Generating spectra from images

Combining the above, we train a CDM to learn the conditional distribution of spectra given an image. At inference time, we sample from this distribution a number of times in order to generate candidate spectra. Then, using the contrastive network, we compute the similarity between the latent representations of the generated spectra and that of the original image. The closest match is our best guess of the spectrum. An overview of the CDM training procedure is shown in Figure 1.

3 Experiments

Implementation details We normalize the spectra to the range $[-1, 1]$. All images are resized to 71x71, random cropped to 64x64 and randomly flipped horizontally and vertically with a probability of 0.5. We train the CDM using Adam Kingma and Ba [2015] with a learning rate of 10^{-4} , and a batch size of 256. We use 250 timesteps with a cosine variance schedule. Models are trained for 500k iterations with 2 NVIDIA GeForce RTX 3090s. The contrastive network is trained with Adam, a learning rate of 10^{-4} , a weight decay of 10^{-3} , and a batch size of 512.

Compared methods We compare our method against that of Wu and Peek [2020]. They first train the VAE of Portillo et al. [2020] to learn a distribution over spectra, followed by a network that predicts the latent code of a spectrum from the associated image. Moreover, we adapt ULISSE Doorenbos et al. [2022] for this use-case. ULISSE only uses photometry, and finds objects sharing similar properties by a similarity search in pretrained feature space. To obtain a guess of a spectrum, we find the nearest neighbour of an image in the feature space of a pretrained EfficientNet-b0, and use its spectrum as a guess for the spectrum of the query.

Results The quantitative results in Table 1 show that selecting the best spectrum with the contrastive network improves upon generating single spectra in mean squared error (MSE). We outperform ULISSE, but the method of Wu and Peek [2020] reaches a lower MSE than ours. The qualitative results shown in Figure 2 suggest this reduction in MSE is caused by it producing much smoother, less realistic samples. Metrics like the MSE are known to favor smooth samples Zhang et al. [2018], and are thus not a perfect metric for our goal. For instance, by taking the mean of the best matching samples, we can reduce the MSE, but the samples become interpolations of multiple realistic spectra.

The fifth row in Figure 2 shows an unfortunate consequence of the normalization to $[-1, 1]$. Our generated spectrum in column (c) generally lies around 0.25 lower than the actual spectrum in (b), although their shape is similar. Because normalization is done by $x_{norm} = \frac{x - x_{min}}{x_{max} - x_{min}}$, the extrema of a spectrum have a large impact on what values x_{norm} will contain. Hence, the relative size of the peak around $\log \lambda \pm 3.83$ would need to be estimated perfectly to get the whole spectrum in the right place. If done incorrectly, this has a very large negative impact on MSE, despite an otherwise reasonably well-estimated spectrum.

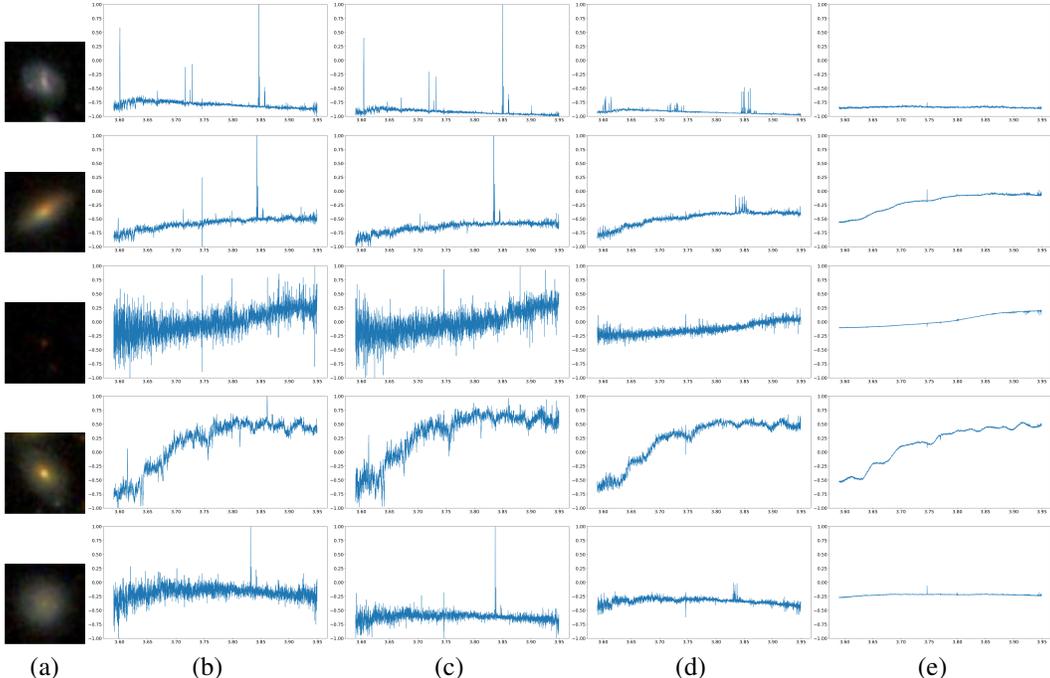


Figure 2: Generated spectra for the images in (a). In (b) we show the target (real) spectra, in (c) the best match according to our contrastive model out of 25 samples, in (d) the spectrum obtained by averaging the 5 best matches according to the contrastive model, out of 25 samples total, and in (e) the results with the method of Wu and Peek [2020]. While the MSE of (e) is the lowest, we believe (c) to be the most useful.

Table 1: Mean and standard deviation over 3 runs of the MSE on the validation set. Contrastive (n) uses the contrastive network to find the best match out of n samples, Contrastive+Mean (n/m) averages the best m out of n matches according to the contrastive network.

ϵ_θ	τ_θ	Conditioning	Selection	MSE
U-net Falk et al. [2019] (12.4m)	ResNet-18 He et al. [2016] (11.2m)	Concat	Single	0.179 ± 0.017
		Concat	Contrastive (5)	0.160 ± 0.010
		Concat	Contrastive (25)	0.154 ± 0.004
		Concat	Contrastive+mean (25/5)	0.105 ± 0.003
U-net Falk et al. [2019] (23.7m)	ResNet-18 He et al. [2016] (11.2m)	X-attention	Single	0.225 ± 0.014
		X-attention	Contrastive (5)	0.136 ± 0.008
		X-attention	Contrastive (25)	0.118 ± 0.010
		X-attention	Contrastive+Mean (25/5)	0.103 ± 0.008
ULISSE (5.3m)				0.168 ± 0.000
Wu and Peek [2020] (25.6m VAE + 6.8m hybrid-conv)				0.080 ± 0.000

4 Discussion & conclusion

Generating high-dimensional, noisy spectra is a difficult task, especially as we do not employ any further preprocessing, besides the normalization to $[-1, 1]$. Other works involving spectra and deep learning typically use a variety of preprocessing methods, such as low-pass median filtering or de-redshifting Muthukrishna et al. [2019], Portillo et al. [2020]. However, all preprocessing imposes some assumptions on the data. As we aim to show our method works on as general a case as possible, we keep the preprocessing minimal. Nonetheless, with task-specific knowledge, the results can likely be improved further. For example, the images can at times be extremely faint, thus a selection based on *e.g.* redshift would likely boost performance.

A direct consequence of our normalization is that we are effectively matching images with the correct *shape* of spectrum, while ignoring the intensities. This is done for well-behaved training, as done in other deep learning works involving spectra such as Tan et al. [2022]. However, this might destroy some important properties for certain use-cases.

Additionally, despite a wide hyperparameter search, we found the VAE architecture of Portillo et al. [2020] unable to generate high-quality reconstructions of our spectra, producing overly smooth, non-realistic samples, unlike the diffusion models. We believe this is caused by the higher dimensional spectra (3598 vs. 1000) and less preprocessing. As Wu and Peek [2020] rely on this VAE, their method will have this problem as well. Nonetheless, the fact that their MSE is lower shows a mismatch between the metric and our desired output. This discrepancy prevents proper evaluation of methods, and solving this is a crucial step in furthering this line of research.

All in all, our method is capable of producing reasonable estimates of spectra directly from photometric observations. With task-specific knowledge and pre-processing, we envision this to be an exceptionally useful tool for the selection of objects for spectroscopic follow-up.

5 Broader impact

We have applied diffusion models, a class of generative models, to an astronomical use-case. While generative models as a whole come with substantial ethical dilemmas, such as generating deepfakes or other harmful content, there are no such clear cases for our domain. Nonetheless, our application might lead to a better understanding of these models, and shed some light on how and where they can be used. On the brighter side, our generated spectra might allow for more targeted searches, thus reducing superfluous spectral acquisition, saving time, carbon and money.

Acknowledgments and Disclosure of Funding

This work was funded by the Swiss National Science Foundation (SNSF), research grant 200021_192285 “Image data validation for AI systems”.

References

- Romina Ahumada, Carlos Allende Prieto, Andrés Almeida, Friedrich Anders, and Scott F. et al. Anderson. The 16th Data Release of the Sloan Digital Sky Surveys: First Release from the APOGEE-2 Southern Survey and Full Release of eBOSS Spectra. *The Astrophysical Journal Supplement*, 249(1):3, July 2020. doi: 10.3847/1538-4365/ab929e.
- Khaled Bayouhd, Raja Knani, Fayçal Hamdaoui, and Abdellatif Mtibaa. A survey on deep multimodal learning for computer vision: advances, trends, applications, and datasets. *The Visual Computer*, 38(8):2939–2970, 2022.
- Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pages 1597–1607. PMLR, 2020.
- Lars Doorenbos, Olena Torbaniuk, Stefano Cavuoti, Maurizio Paolillo, Giuseppe Longo, Massimo Brescia, Raphael Sznitman, and Pablo Márquez-Neila. ulisse: A tool for one-shot sky exploration and its application for detection of active galactic nuclei. *Astronomy & Astrophysics*, 666:A171, 2022.
- Thorsten Falk, Dominic Mai, Robert Bensch, Özgün Çiçek, Ahmed Abdulkadir, Yassine Marrakchi, Anton Böhm, Jan Deubner, Zoe Jäckel, Katharina Seiwald, et al. U-net: deep learning for cell counting, detection, and morphometry. *Nature methods*, 16(1):67–70, 2019.
- Chenyou Fan, Xiaofan Zhang, Shu Zhang, Wensheng Wang, Chi Zhang, and Heng Huang. Heterogeneous memory enhanced multimodal attention model for video question answering. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 1999–2007, 2019.

- Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.
- Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in Neural Information Processing Systems*, 33:6840–6851, 2020.
- Benne W Holwerda, John F Wu, William C Keel, Jason Young, Ren Mullins, Joannah Hinz, KE Saavik Ford, Pauline Barmby, Rupali Chandar, Jeremy Bailin, et al. Predicting the spectrum of ugc 2885, rubin’s galaxy with machine learning. *The Astrophysical Journal*, 914(2):142, 2021.
- Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *International Conference on Learning Representations*, 2015.
- Daniel Masters, Peter Capak, Daniel Stern, Olivier Ilbert, Mara Salvato, Samuel Schmidt, Giuseppe Longo, Jason Rhodes, Stephane Paltani, Bahram Mobasher, Henk Hoekstra, Hendrik Hildebrandt, Jean Coupon, Charles Steinhardt, Josh Speagle, Andreas Faisst, Adam Kalinich, Mark Brodwin, Massimo Brescia, and Stefano Cavuoti. Mapping the Galaxy Color-Redshift Relation: Optimal Photometric Redshift Calibration Strategies for Cosmology Surveys. *Astrophysical Journal*, 813(1):53, November 2015. doi: 10.1088/0004-637X/813/1/53.
- Daniel Muthukrishna, David Parkinson, and Brad E Tucker. Dash: Deep learning for the automated spectral classification of supernovae and their hosts. *The Astrophysical Journal*, 885(1):85, 2019.
- Stephen KN Portillo, John K Parejko, Jorge R Vergara, and Andrew J Connolly. Dimensionality reduction of sdss spectra with variational autoencoders. *The Astronomical Journal*, 160(1):45, 2020.
- Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning*, pages 8748–8763. PMLR, 2021.
- Aditya Ramesh, Mikhail Pavlov, Gabriel Goh, Scott Gray, Chelsea Voss, Alec Radford, Mark Chen, and Ilya Sutskever. Zero-shot text-to-image generation. In *International Conference on Machine Learning*, pages 8821–8831. PMLR, 2021.
- Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10684–10695, 2022.
- Aiham Taleb, Matthias Kirchler, Remo Monti, and Christoph Lippert. Contig: Self-supervised multi-modal contrastive learning for medical imaging with genetics. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 20908–20921, 2022.
- Lei Tan, Ying Mei, Zhicun Liu, Yangping Luo, Hui Deng, Feng Wang, Linhua Deng, and Chao Liu. A robust hot subdwarfs identification method based on deep learning. *arXiv preprint arXiv:2201.08967*, 2022.
- John F Wu and Joshua EG Peek. Predicting galaxy spectra from images with hybrid convolutional neural networks. *arXiv preprint arXiv:2009.12318*, 2020.
- Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 586–595, 2018.

A Checklist

1. For all authors...
 - (a) Do the main claims made in the abstract and introduction accurately reflect the paper’s contributions and scope? [\[Yes\]](#)

- (b) Did you describe the limitations of your work? [\[Yes\]](#) See the results paragraph in 3 and 4, where we e.g. discuss that our normalization might be a limiting factor.
 - (c) Did you discuss any potential negative societal impacts of your work? [\[Yes\]](#) See Section 5
 - (d) Have you read the ethics review guidelines and ensured that your paper conforms to them? [\[Yes\]](#)
2. If you are including theoretical results...
- (a) Did you state the full set of assumptions of all theoretical results? [\[N/A\]](#)
 - (b) Did you include complete proofs of all theoretical results? [\[N/A\]](#)
3. If you ran experiments...
- (a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? [\[Yes\]](#)
 - (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [\[Yes\]](#)
 - (c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [\[Yes\]](#) Standard deviations are given in Table 1
 - (d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [\[Yes\]](#)
4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...
- (a) If your work uses existing assets, did you cite the creators? [\[Yes\]](#)
 - (b) Did you mention the license of the assets? [\[N/A\]](#)
 - (c) Did you include any new assets either in the supplemental material or as a URL? [\[Yes\]](#)
 - (d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? [\[N/A\]](#)
 - (e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [\[N/A\]](#)
5. If you used crowdsourcing or conducted research with human subjects...
- (a) Did you include the full text of instructions given to participants and screenshots, if applicable? [\[N/A\]](#)
 - (b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [\[N/A\]](#)
 - (c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [\[N/A\]](#)