Dataset Augmentation using Diffusion Models

Satya Sai Srinath, Debarshi Deka, Prem Abhinav Potta
University of Wisconsin - Madison
{sgnamburi, ddeka, potta}@wisc.edu

Abstract

*Our idea is to **use a latent space diffusion model such as Stable Diffusion to augment a standard image dataset with synthetic images***

- *This can help in increasing the size of the current datasets with mininal effort.*
- *This can also help in reducing class imbalances in a given dataset if there exists any.*
- *This aids in automation and standardization of prompt engineering, an active area of research which is not heavily explored for this particular domain.*

*The code is available at **this repository***

1. Introduction

Collecting data, annotating and cleaning is a tiresome task and it also requires domain expertise (eg: labeling medical data). This calls for the generation of synthetic data as it is easier to generate and maintain. Generative models like Dall-E2 [16] and Stable diffusion [18] have shown great promise in generating and transforming images.

With the advent of GANs, research community has tried to use these generative models to generate images and use it as an augmentation. But training GANs is very difficult and might easily go unstable during training [8, 9].

Another class of generative models that became prominent recently are diffusion models whose core idea is inspired from statistical physics. These models are promising enough to explore whether they can be used to generate quality datasets.

2. Related work

In this section, we deep dive into the preliminaries and works that was done previously in this field

2.1. Diffusion Models

Diffusion models are inspired by non-equilibrium thermodynamics. They define a Markov chain of diffusion steps to slowly add random noise to data and then learn to reverse the diffusion process to construct desired data samples from the noise. Unlike Variational Auto Encoders [12] or flow models [17], diffusion models are learned with a fixed procedure and the latent variable has high dimensionality (same as the original data). Several diffusion-based generative models have been proposed with similar ideas underneath, including diffusion probabilistic models in [22], noise-conditioned score network in [23], and denoising diffusion probabilistic models in [10].

2.2. Finetuning Diffusion Models

While the Vanilla Stable Diffusion produces reasonably good images, we went a step further and finetuned these models to gain better control on image generation and to understand whether the finetuned version produces better augmentations.

Textual-inversion [6] is a technique for capturing novel concepts from a small number of example images in a way that can later be used to control text-to-image pipelines. It does so by learning new 'words' in the embedding space of the pipeline's text encoder.

DreamBooth [19] is a method to personalize text-to-image models like stable diffusion given just a few (3-5) images of a subject. Then, the model synthesizes the subject into a different context to produce a brand-new image whilst maintaining the features.

2.3. Augmentation Methods

Data Augmentation is a widely used method when training machine learning models. When we have less amount of training data or the model is overfitting, one common approach is to increase the amount of data which can generalize the learning. Various data augmentation methods were summarized in figure 1

The basic image manipulations involve geometric transformations, color space transformations, photometric transformations, mixing images and random erasing which are the standard augmentation methods due to them being intuitive and easy to implement. While they improve accuracy compared to baselines, they are limited in capabilities.

Other data augmentation techniques based on deep learning involve feature space augmentation, adversarial train-

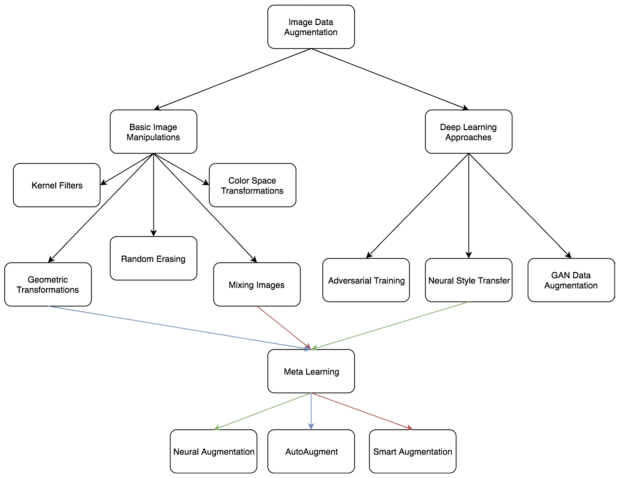

Figure 1. Different data augmentation methods as summarized in [21]

ing, Neural Style Transfer [7], GAN-based data augmentation which include Progressive GAN [11], Cycle GAN [24] and DCGANs [15]. Other techniques include Meta learning which involve neural augmentation, smart augmentation using an adaptive CNN to merge two images and AutoAugment [3] which is a reinforcement learning algorithm that searches for an optimal augmentation policy. It is reported in [5] and several other papers that using GANs as data augmentation resulted in better accuracy when compared to classic augmentations.

Recent research is shifted in using diffusion models for data augmentation as reported in [1], [20] and [13] but are mostly in medical domain and is almost a binary classification setting.

3. Proposed Approach

We plan to answer the following question: **"Can we use images generated by Diffusion Models to augment and improve the accuracy of a multi class classification?"**

In brief, the questions we addressed are:

- **Is adding synthetic data from diffusion models helpful in improving accuracy?**
- **What happens if we replace original data with synthetic data?**

3.1. Dataset curation

Our dataset consists of 3 parts

- Original set - This is a small version of Imagenet where we picked the classes that are present in CIFAR-10 and webscrapped images from images.cv/ . We refer this as $d_{original}$ and has a total of 7000 images with 700 per class.
- Synthetic set - This set is constructed by passing prompts to Stable Diffusion and it's finetuned versions. We refer this as $d_{synthetic}$ and has 7000 images with 700 per class. We generated 3 versions of synthetic dataset, $d_{synthetic(VanillaSD)}, d_{synthetic(Textual)}$ and $d_{synthetic(Dreambooth)}$ corresponding to the vanilla version, textual inversion and dreambooth as generation model.
- Test set - This comes from similar distribution compared to Original set and we refer this as d_{test} and has a total of 3000 images with 300 per class.

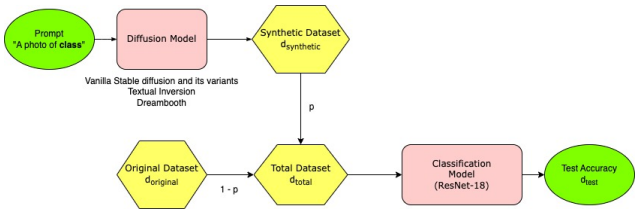

Figure 2. Flowchart describing various steps in our approach

We used ResNet-18 as classification backbone with SGD as optimizer. For experiments with random weight initialization, we trained the model for 200 epochs and with pretrained weights, we finetuned for 20 epochs. In finetuning, we considered both the settings where the gradients are propagated through the entire backbone and where the gradients are passed only to the last linear layer. Our approach is presented in figure 2

4. Experiments

In this section, we explain the methodology behind the experiments and the results.

Assume the training set be d_{train} and validation set be $d_{validate}$. This set can be constructed from original and synthetic set in the following way:

$$d_{total} = p * d_{synthetic} + (1-p) * d_{original}$$
$$d_{train} = val\_percent * d_{total}$$
$$d_{validate} = (1 - val\_percent) * d_{total}$$

where d_{total} is the combination of $d_{synthetic}$ and $d_{original}$ with proportion p:1-p and val_percent is set to 0.85. Note that $d_{synthetic}$ is one among $d_{synthetic(VanillaSD)}, d_{synthetic(Textual)}$ or $d_{synthetic(Dreambooth)}$.

p is varied from 0 to 1 with 0.25 increment and the accuracy results are in plotted in figure 3, 4 and 5. Class-wise accuracies are plotted in 6

As an illustration, when p = 0.25, $d_{synthetic}$ = 0.25*7000 i.e 1750 and $d_{original}$ = 0.75*7000 i.e 5250. Each class gets splitted equally i.e there will be 175 synthetic images and 525 original images totalling to 700 for each class. Note that in the figures, for legend "Only original", the dataset size is not same as 7000 i.e it helps us to understand whether adding additional synthetic images from a different distribution helps in improving accuracy.

4.1. FID computation

FID (Fréchet inception distance) is a metric used to assess the quality of generated images and was introduced in [9]. It helps in understanding the distribution gap between the original dataset and the synthetic dataset and the results are summarized in table 1:

We draw the following observations from the experiments:

- ImageNet pretrained weights are useful for feature extraction (from 3, 4 and 5) and it is as expected because the classes are very common. Even if we have a low quality dataset, the pretrained models are able to achieve great accuracy (refer when p=1 across 3, 4 and 5)
- Synthetic data improved the accuracy a few points (refer Only original vs Original+Synthetic for a fixed p from 3, 4 and 5) which suggests that this method can be potentially used as a means of data augmentation.
- Accuracy dropped when percentage of synthetic dataset is increased (as we go from p=0 to p=1) which suggests that the quality of synthetic dataset generation can be further improved.
- From figure 6, we can observe that the class accuracy drops significantly when the proportion of synthetic dataset is increased validating the previous observation that the quality of images can be improved. But this also gives information on which classes were hurt the most i.e classes for which more data or better generation method is needed which can also be observed from table 1.

5. Conclusions

We observed that replacing original data with synthetic data hurts the accuracy of the model which reveals that the synthetic data is not coming from the same distribution as the original dataset. We also observed that adding synthetic data helps in improving the accuracy of the classifier even though it is from a different distribution which suggests that this method can be used as a means of augmentation. We also observed that vanilla Stable diffusion might be a good starting point (from 1) and prompt engineering can be explored further.

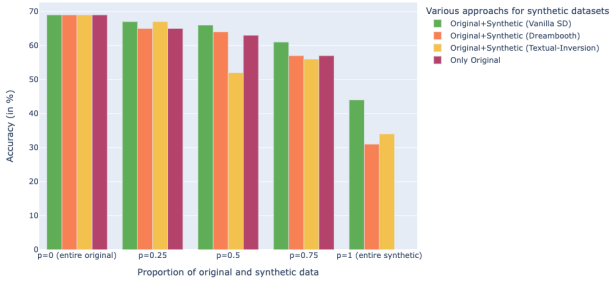

Figure 3. Classification accuracy of the model with varying synthetic datasets and proportions. Model is pretrained from scratch for 200 epochs

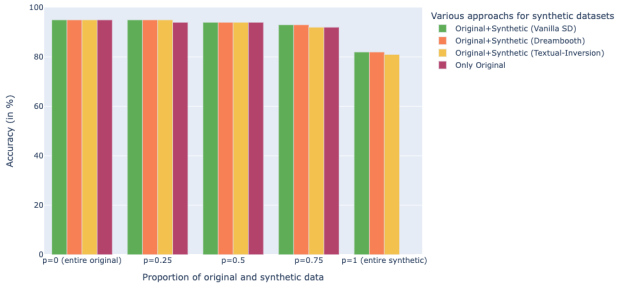

Figure 4. Classification accuracy of the model with varying synthetic datasets and proportions. Imagenet weights are taken and the backbone is finetuned for 20 epochs.

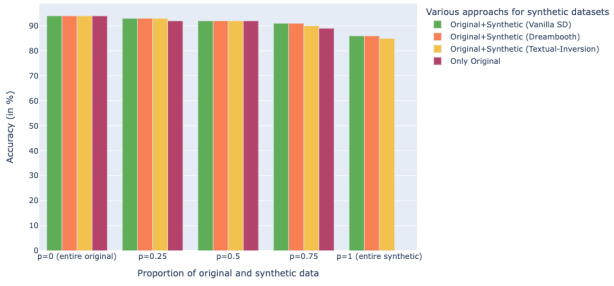

Figure 5. Classification accuracy of the model with varying synthetic datasets and proportions. Imagenet weights are taken and the backbone is frozen for 20 epochs.

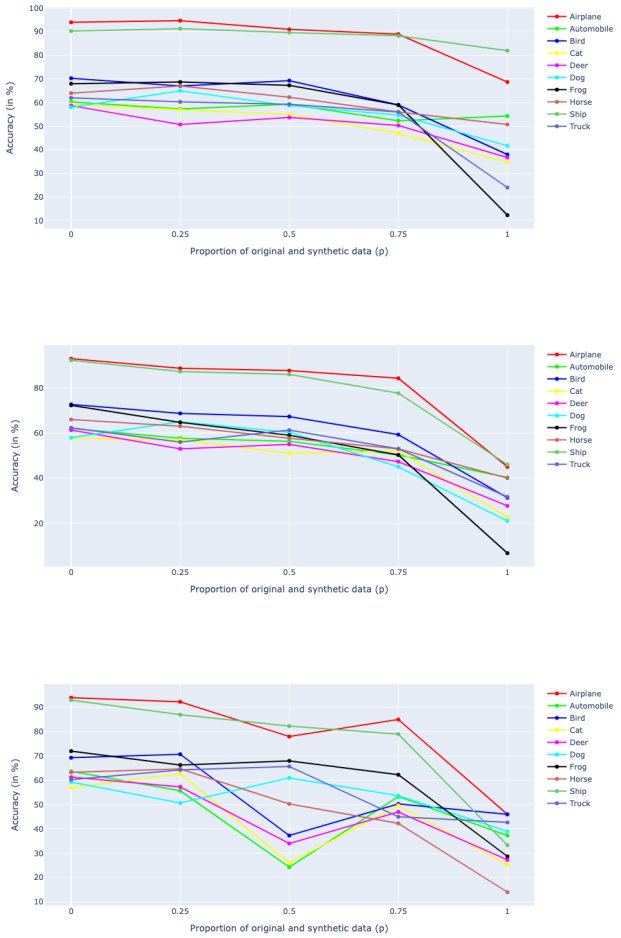

Figure 6. Individual class accuracy across three synthetic datasets (Vanilla SD, Dreambooth and Textual Inversion) and varying proportions. Training settings are same as in fig. 3

Table 1. FID comparison between original dataset and various synthetic datasets (lower the better)

	Vanilla SD ($d_{synthetic(VanillaSD)}$)	Dreambooth ($d_{synthetic(Dreambooth)}$)	Textual Inversion ($d_{synthetic(Textual)}$)
Airplane	87.2	**26.4**	31.2
Automobile	**54.3**	81.7	89.1
Bird	**61**	67.3	78.3
Cat	95.5	**92.6**	135.5
Deer	93.5	99.3	**84.8**
Dog	163.4	93.7	**93**
Frog	84.2	80.6	**67.1**
Horse	**64**	65.3	81.2
Ship	**44.9**	76.6	75.4
Truck	99.2	**77.1**	112.8
Overall	**39.9**	42.7	48.3

6. Explorations and Future work

We realized that prompt-engineering is the key to generate more realistic images which can serve as better augmentations. So, we explored some methods to create more meaningful prompts and use them as inputs for Diffusion models.

6.1. Generating prompts using GPT-3

GPT-3 [2] is an autoregressive language model which takes a prompt as input and produces text that continues the prompt. So, we formed few templates such as:

- Describe what a {} looks like?
- How can you identify a {}?
- What does a {} looks like?
- Describe an image from the internet of a {}?
- A caption of an image of {}:

where {} is replaced with each class. These templates are passed to GPT-3 and the output text (which described these classes) served as input for Stable diffusion.

Davinci-002 is used with temperature 0.99 and with max-token size of 20. While the output text describes about the class in a more detailed way, when passed to Stable Diffusion, we observed that the images generated are not ideal and suffer from issues as shown in 8

6.2. Experiments with RandAugment

We also applied RandAugment [4] and observed that it is not improving accuracy as compared to our approach.

6.3. Zero-shot CLIP

CLIP [14] is a recent model which is able to achieve state-of-the-art accuracies on classification tasks on various datasets. So, we performed zero-shot on our test set and observed an accuracy of 97%

7. Contributions

The rough splitup and percentages of the works done are as follows:

- Satya Sai Srinath (40%) - Experiments with Vanilla SD, Training and finetuning Resnet-18, Prompt engineering using GPT-3, Zero-shot CLIP and RandAugment, Report
- Debarshi Deka (30%) - Experiments with Textual Inversion and Dreambooth, Report
- Prem Abhinav Potta (30%) - Dataset curation, FID computation, Report

References

[1] Hazrat Ali, Shafaq Murad, and Zubair Shah. Spot the fake lungs: Generating synthetic medical images using neural diffusion models. *arXiv preprint arXiv:2211.00902*, 2022. 2

[2] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901, 2020. 4

[3] Ekin D Cubuk, Barret Zoph, Dandelion Mane, Vijay Vasudevan, and Quoc V Le. Autoaugment: Learning augmentation policies from data. *arXiv preprint arXiv:1805.09501*, 2018. 2

[4] Ekin D Cubuk, Barret Zoph, Jonathon Shlens, and Quoc V Le. Randaugment: Practical automated data augmentation with a reduced search space. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops*, pages 702–703, 2020. 4

[5] Maayan Frid-Adar, Idit Diamant, Eyal Klang, Michal Amitai, Jacob Goldberger, and Hayit Greenspan. Gan-based synthetic medical image augmentation for increased cnn performance in liver lesion classification. *Neurocomputing*, 321:321–331, 2018. 2

[6] Rinon Gal, Yuval Alaluf, Yuval Atzmon, Or Patashnik, Amit H Bermano, Gal Chechik, and Daniel Cohen-Or. An image is worth one word: Personalizing text-to-image generation using textual inversion. *arXiv preprint arXiv:2208.01618*, 2022. 1

[7] Leon A Gatys, Alexander S Ecker, and Matthias Bethge. A neural algorithm of artistic style. *arXiv preprint arXiv:1508.06576*, 2015. 2

[8] Ishaan Gulrajani, Faruk Ahmed, Martin Arjovsky, Vincent Dumoulin, and Aaron C Courville. Improved training of wasserstein gans. *Advances in neural information processing systems*, 30, 2017. 1

[9] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. *Advances in neural information processing systems*, 30, 2017. 1, 3

[10] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in Neural Information Processing Systems*, 33:6840–6851, 2020. 1

[11] Tero Karras, Timo Aila, Samuli Laine, and Jaakko Lehtinen. Progressive growing of gans for improved quality, stability, and variation. *arXiv preprint arXiv:1710.10196*, 2017. 2

[12] Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013. 1

[13] Walter HL Pinaya, Petru-Daniel Tudosiu, Jessica Dafflon, Pedro F Da Costa, Virginia Fernandez, Parashkev Nachev, Sebastien Ourselin, and M Jorge Cardoso. Brain imaging generation with latent diffusion models. In *MICCAI Workshop on Deep Generative Models*, pages 117–126. Springer, 2022. 2

[14] 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. 4

[15] Alec Radford, Luke Metz, and Soumith Chintala. Unsupervised representation learning with deep convolutional generative adversarial networks. *arXiv preprint arXiv:1511.06434*, 2015. 2

[16] Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. Hierarchical text-conditional image generation with clip latents. *arXiv preprint arXiv:2204.06125*, 2022. 1

[17] Danilo Rezende and Shakir Mohamed. Variational inference with normalizing flows. In *International conference on machine learning*, pages 1530–1538. PMLR, 2015. 1

[18] 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. 1

[19] Nataniel Ruiz, Yuanzhen Li, Varun Jampani, Yael Pritch, Michael Rubinstein, and Kfir Aberman. Dreambooth: Fine tuning text-to-image diffusion models for subject-driven generation. *arXiv preprint arXiv:2208.12242*, 2022. 1

[20] Luke W Sagers, James A Diao, Matthew Groh, Pranav Rajpurkar, Adewole S Adamson, and Arjun K Manrai. Improving dermatology classifiers across populations using images generated by large diffusion models. *arXiv preprint arXiv:2211.13352*, 2022. 2

[21] Connor Shorten and Taghi M Khoshgoftaar. A survey on image data augmentation for deep learning. *Journal of big data*, 6(1):1–48, 2019. 2

[22] Jascha Sohl-Dickstein, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. In *International Conference on Machine Learning*, pages 2256–2265. PMLR, 2015. 1

[23] Yang Song and Stefano Ermon. Generative modeling by estimating gradients of the data distribution. *Advances in Neural Information Processing Systems*, 32, 2019. 1

[24] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *Proceedings of the IEEE international conference on computer vision*, pages 2223–2232, 2017. 2

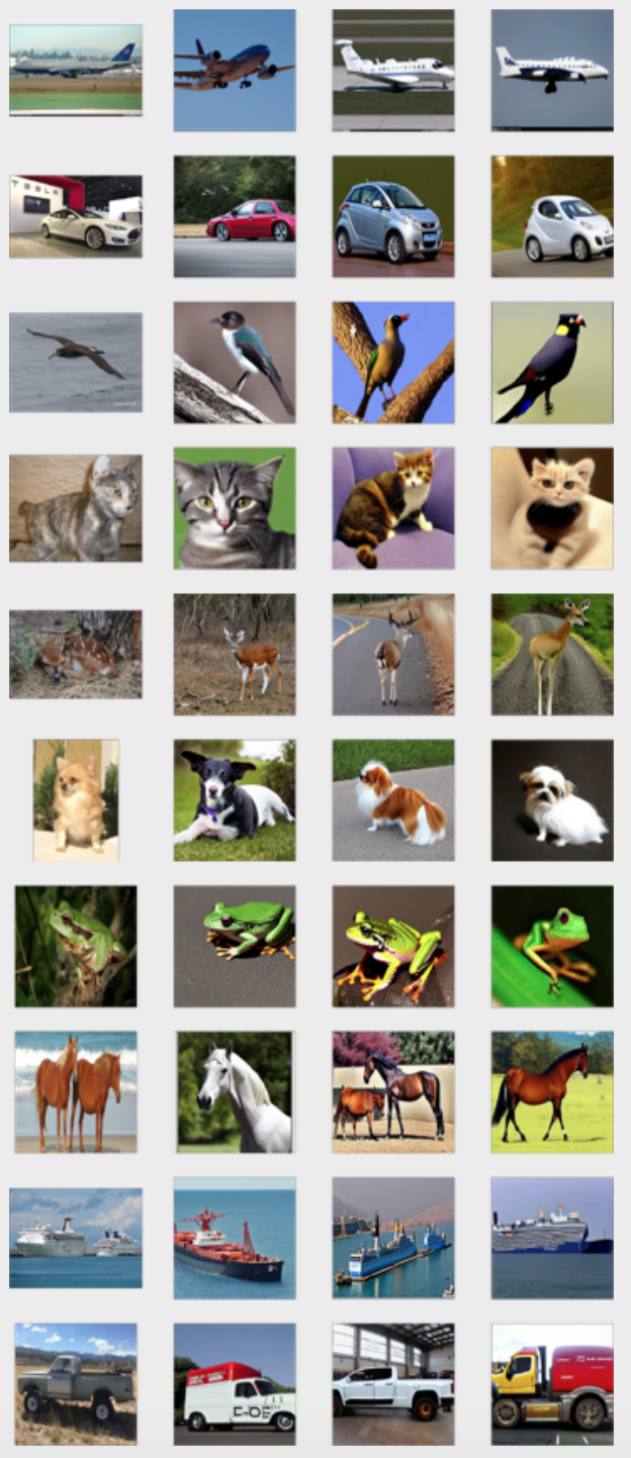

Figure 7. Example images from each class in our dataset. Rows correspond to classes and columns are original, vanilla stable diffusion, dreambooth and textual inversion. Images are picked at random.

Figure 8. Example images generated from vanilla stable diffusion model when following prompts are passed (from left to right): “A car on a road in front of mountainsThis image shows a car driving on a road.”, Middle image: “A car has four round, metal plates called tires.”, Right image: “A wild mustang galloping through the desert.” These prompts are generated from GPT-3