TEXT-TO-IMAGE DIFFUSION MODELS ARE ZERO-SHOT CLASSIFIERS

Anonymous authors
Paper under double-blind review

ABSTRACT

Text-to-image diffusion models have demonstrated remarkable generative capabilities, suggesting they learn informative representations of image-text data. However, their abilities are not fully understood and they have not been thoroughly explored on downstream tasks. We investigate diffusion models by proposing a method for evaluating them as zero-shot classifiers. The key idea is using a diffusion model's ability to denoise a noised image given a textual description of a label as a proxy for that label's likelihood. We apply our method to Imagen, using it to probe fine-grain aspects of Imagen's knowledge and comparing it with CLIP's zero-shot abilities. Imagen performs competitively with CLIP on a wide range of zero-shot image classification datasets. Additionally, it is more robust than CLIP and can successfully perform attribute binding while CLIP does not. Although generative pre-training is common in NLP, visual foundation models often use other methods such as contrastive learning. Based on our findings, we argue that generative pre-training should be explored as a compelling alternative for visual and vision-language problems.

1 INTRODUCTION

Generative text-to-image models based on denoising diffusion probablistic models (Ho et al., 2020) such as Imagen (Saharia et al., 2022a), Dalle-2 (Ramesh et al., 2022), and Stable Diffusion (Rombach et al., 2022) have demonstrated excellent abilities in generating high-resolution images and generalizing to diverse text prompts. Their strong performance suggests that they learn effective representations of image-text data. However, their ability to transfer to downstream discriminative tasks and how they compare to other pre-trained image models has not been explored thoroughly.

We investigate these questions by transferring the Imagen diffusion model to discriminative tasks. Specifically, we propose a method for using text-to-image diffusion models as zero-shot image classifiers. While Burgert et al. (2022) have explored using Stable Diffusion for zero-shot referring segmentation and Bar et al. (2022) have explored using inpainting models for few-shot pixel-level tasks, to our knowledge zero-shot classification with diffusion models has not been studied previously.

Our method essentially runs Imagen as a generative classifier (Ng & Jordan, 2001), using a reweighted version of Imagen's variational lower bound to score images since diffusion models do not produce exact likelihoods. First, our method makes a text prompt for each class (e.g. "a photo of a cat."). Then it scores input the image conditioned on each text prompt, measuring how helpful each prompt is for denoising the image averaged over different noise levels. The class corresponding to the prompt with the best score is predicted. This classification procedure requires denoising with Imagen many times for every class (with different noise levels), so it is computationally expensive. To make it usable in practice, we present improvements that increase the method's sample efficiency by up to 1000x, such as pruning away obviously-incorrect classes early.

We compare Imagen against CLIP[1] (Radford et al., 2021), a widely used model for zero-shot image-text tasks trained with contrastive learning. First, we demonstrate that Imagen has strong zero-shot classification accuracies (competitive with CLIP) on several diverse vision datasets. Next, we show that Imagen performs robustly and achieves SOTA results (>50% error reduction over CLIP) on images with texture-shape conflicting cues (Geirhos et al., 2018) that have shown to

[1] We use the largest public CLIP model

confound pre-trained convolutional supervised models. An important use of our classification method is in quantitatively studying fine-grained aspects of what diffusion models know (as opposed to qualitatively examining model generations). We showcase this by studying attribute binding in Imagen, and find that, unlike CLIP, it can successfully bind attributes in some settings. Together, our study of Imagen suggests that text-to-image diffusion models learn powerful representations that can effectively be used for tasks beyond image generation.

2 ZERO-SHOT CLASSIFICATION USING IMAGEN

We seek to demonstrate the knowledge transfer capabilities of text-to-image diffusion models using the setting of zero-shot classification.

Imagen as a Generative Classifier We begin with a dataset, $\wr(\boldsymbol{x}^1, y^1), \ldots, (\boldsymbol{x}^n, y^n)\int \subseteq \mathbb{R}^{d_1 \times d_2} \times [\mathsf{y}_K]$ of n images where each image belongs to one of K classes $[\mathsf{y}_K] := \{\mathsf{y}_1, \mathsf{y}_2, \cdots, \mathsf{y}_K\}$. Given an image $\boldsymbol{x}$, our goal is to predict the most likely class assignment

$$\tilde{y} = \arg\max_{\mathsf{y}_k} p(y = \mathsf{y}_k|\boldsymbol{x}) = \arg\max_{\mathsf{y}_k} p(\boldsymbol{x}|y = \mathsf{y}_k) \cdot p(y = \mathsf{y}_k) = \arg\max_{\mathsf{y}_k} \ \log p(\boldsymbol{x}|y = \mathsf{y}_k).$$

where we assume a uniform prior $p(y_i = \mathsf{y}_k) = \frac{1}{k}$ that can be dropped from the $\arg\max$.[2] A generative classifier (Ng & Jordan, 2001) uses a conditional generative model with parameters θ to estimate the likelihood as $p_\theta(\boldsymbol{x}|y = \mathsf{y}_k)$.

Imagen is conditioned on text prompts rather than class labels. Thus we convert each label, y_k, to text using a mapping T with a dataset-specific template (e.g. $\mathsf{y}_k \rightarrow$ `A photo of a` y_k). Furthermore, diffusion models do not produce exact log-likelihoods (i.e. we cannot compute $\log p_\theta(\boldsymbol{x}|y = \mathsf{y}_k)$ directly). Our key idea for a solution is to use the diffusion model's variational lower bound (VLB) as a proxy. In particular, we use $\mathcal{L}_{\mathsf{Diffusion}}$, the portion of the VLB corresponding to denoising images, as Imagen is not trained with the other loss terms. See Appendix A for more detailed background on diffusion models and their training. The predicted class is:

$$\begin{aligned} \tilde{y} &= \arg\max_{\mathsf{y}_k} \ \log p_\theta(\boldsymbol{x}|y = \mathsf{y}_k) \approx \arg\min_{\mathsf{y}_k} \ \mathcal{L}_{\mathsf{Diffusion}}(\boldsymbol{x}, \mathsf{y}_k) \\ &= \arg\min_{\mathsf{y}_k \in [\mathsf{y}_K]} \ \mathbb{E}_{\epsilon,t}\Big[\boldsymbol{w}_t \|\boldsymbol{x} - \tilde{\boldsymbol{x}}_{\boldsymbol{\theta}}\big(\boldsymbol{x}_t, \boldsymbol{c}_{\boldsymbol{\phi}}(\mathrm{T}(\mathsf{y}_k)), t\big)\|_2^2\Big] \end{aligned} \tag{1}$$

Estimating the Expectation: We approximate the expectation in Equation (1) using a Monte-Carlo estimatation. At each step, we sample a $t \sim \mathcal{U}([0, 1])$ and then $\boldsymbol{x}_t$ according to the forward diffusion process (Equation (2)): $\boldsymbol{x}_t \sim q(\boldsymbol{x}_t|\boldsymbol{x}_0)$. Next, we denoise this noisy image using Imagen (i.e. we use Imagen to predict $\boldsymbol{x}$ from $\boldsymbol{x}_t$), obtaining $\hat{\boldsymbol{x}} = \tilde{\boldsymbol{x}}_{\boldsymbol{\theta}}\big(\boldsymbol{x}_t, \boldsymbol{c}_{\boldsymbol{\phi}}(\mathrm{T}(\mathsf{y}_k)), t\big)$. We predict the class with the lowest average weighted squared error $\boldsymbol{w}_t \|\boldsymbol{x} - \hat{\boldsymbol{x}}\|_2^2$ across steps.

The choice of weighting function, $\boldsymbol{w}_t$, is crucial to the overall performance of the classification algorithm. Here, we chose $\boldsymbol{w}_t := \exp(-7t)$ which we found to work well across many datasets and use it in our experiments. Furthermore, the algorithm presented here is computationally expensive because $\mathcal{L}_{\mathsf{Diffusion}}$ has a fairly high variance. We propose efficiency techniques that reduce the compute cost of computing argmin over classes in Appendix B.

3 EMPIRICAL ANALYSIS AND RESULTS

Here we evaluate Imagen as a zero-shot classifier on a variety of tasks. We compare with CLIP (Radford et al., 2021), which is widely used as a zero-shot classifier. Our main aim is to study the strengths and weaknesses of image-text representation learning via generative training as in Imagen and contrastive training as used for CLIP. See Appendix C for details on the experimental setup.

3.1 IMAGE CLASSIFICATION

We first evaluate the performance of Imagen on 13 datasets from Radford et al. (2021) as reported in Table 1. We use the prompt templates and class labels used by Radford et al. (2021), which renames

[2] We can't use a learned prior in the zero-shot setting.

Dataset	Imagen	CLIP
CIFAR10	**96.6**	94.7
CIFAR100	**84.3**	68.6
STL10	**99.6**	99.6
MNIST	**79.2**	74.3
DTD	**37.3**	36.0
Patch Camelyon	**60.3**	58.0
SVHN	**62.7**	21.50
EuroSAT	**60.3**	58.04
Imagenet	62.7	61.3 / 64.0
Stanford Cars	23.0	13.0 / 31.8
Caltech101	68.9	70.2 / 84.1
Oxford Pets	66.5	76.0 / 89.9
Food 101	68.4	83.9 / 93.3

Table 1: Percentage accuracies for zero-shot image classification. For CLIP where two numbers are reported, the accuracy correspond to two settings: downsizing the images to 64x64 and then resizing the images up to 224x224, and resizing directly to 224x224.

Task	Imagen	CLIP
Recognition (Color)	**96**	**94**
Recognition (Shape)	**85**	**98**
Binding (Color|Shape)	**70**	49
Binding (Size|Shape)	49	50
Binding (Shape|Color)	**66**	50
Pair Binding (Color)	**100**	47
Pair Binding (Size)	**99**	52
Spatial	**62**	49

Table 2: Percentage accuracy for models on zero-shot synthetic-data tasks investigating attribute binding. Bold results are better than chance. CLIP is unable to bind attributes, while Imagen can in some settings.

some classes that confuse models (e.g. "crane → "crane bird"" in Imagenet) (OpenAI, 2021b). We use the first prompt from the list, except for Imagenet, where we use "A bad photo of a *label* " since this is a good prompt for both Imagen and CLIP (OpenAI, 2021a).

Since we use the low-resolution Imagen model, we obtain results using CLIP under two settings for a fair comparison. In the first setting, we resize all the images to 64×64 which serves as the base low-resolution dataset. Imagen uses this dataset directly. CLIP requires 224×224 resolution inputs, so we bicubic-upsample the images to this size. In the second setting, we directly resize to 224×224 resolution without first going to 64x64, to obtain the best results possible using CLIP where it can take advantage of its higher input resolution compared to Imagen.

Results: The first eight datasets are all originally of resolution 64×64 or less. On these, Imagen outperforms CLIP on classification accuracy under the same evaluation setting (i.e. the models are conditioned on the same text prompts, etc). Imagen significantly outperforms CLIP on e.g. SVHN, which requires recognizing text in an image, reinforcing the qualitative observation that Imagen is good at generating images containing text (Saharia et al., 2022b). The next five datasets use higher-resolution images. For some of these, taking advantage of CLIP's higher input resolution substantially improves results. It may be possible to get similar benefits from Imagen by incorporating scores from its superresolution models, which we leave for future work to explore.

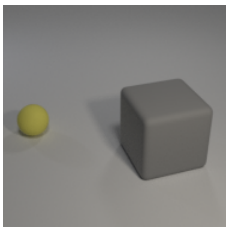
3.2 Robustness

We next study the robustness of text-to-image diffusion models like Imagen by evaluating it on the cue conflict dataset from Geirhos et al. (2018). The dataset consists of Imagenet images altered to have a shape-texture conflict. While (for example) changing an image of a cat so it has a texture similar to elephant skin doesn't confuse humans, it could cause a model to classify the image as an elephant. We use the same setting for classification here as in Appendix C.1. Imagen achieves 82.88 % accuracy compared to 51.56% by CLIP and 79% top-5 accuracy by a supervised trained ResNet50. We provide more details in Appendix C.2.

3.3 Evaluating Attribute Binding on Synthetic Data

We next test attribute binding in Imagen and CLIP on synthetic datasets. Attribute binding is a key piece of compositional reasoning: to understand novel combinations of concepts, one must bind the concepts together and treat them as a whole. While other work has examined attribute binding in text-to-image models by qualitatively examining model generations (Nichol et al., 2021; Yu et al., 2022), our Imagen classifier offers a way of studying the question quantitatively.

Dataset Construction: We use a setup similar to Lewis et al. (2022), where images are generated based on the CLEVR (Johnson et al., 2017) dataset. CLEVR images contain various object (cubes, cylinders, and spheres) with various attributes (different sizes, colors, and materials). We use a modified version of the CLEVR rendering script that generates images containing two objects of different shapes. From these images, we construct four binary classification tasks as shown below:



Recognition tasks determine if the model can identify basic image features by scoring an attribute in the image against one not present. e.g.: A sphere. vs. A cylinder.

Single-object binding tasks test if the model binds a given attribute to the correct object. e.g.: A yellow sphere. vs. A gray sphere.

Pair binding tasks are easier binding tasks where information about both objects is provided. e.g.: A small sphere and a large cube. vs. A large sphere and a small cube.

Spatial tasks test if the model is capable of binding objects and their positions in the image. e.g.: On the left is a yellow sphere. vs. On the right is a yellow sphere.

Recognition Results: Both Imagen and CLIP are able to identify shapes and colors that occur in the image with fairly high accuracy. Imagen is slightly worse at shape identification; we find most of these are due to it mixing up "cylinder" and "cube" when the objects are small.

Binding Results: CLIP performs no better than random chance for the attribute binding tasks, showing it is unable to map attributes to objects on this data. In contrast, Imagen performs excellently at the pair tasks, and better than chance on two of the three single tasks. Part of Imagen's advantage might be in its text encoder, the pre-trained T5 (Raffel et al., 2020) model. Saharia et al. (2022b) find that instead using CLIP's text encoder for Imagen decreased its performance on generations involving specific colors or spatial positions. Similarly, Ramesh et al. (2022) find that DALLE-2, which uses a CLIP text encoder, is worse at attribute binding than GLIDE, which uses representations from a jointly-trained transformer processing the text. However, a perhaps more significant advantage of Imagen over CLIP is its use of cross attention to allow interaction between textual and visual features.

One mistake we observed frequently in Color|Shape is Imagen preferring the color of the larger object in the image; e.g. scoring "A gray sphere" over "A yellow sphere" in Figure 5. We hypothesize that it is helpful for denoising at high noise levels when the text conditioning provides the color for a large region of the image, even when the color is associated with the wrong shape. In the pair task, the full color information for both objects is always provided, which avoids this issue.

Spatial Positioning Results: Previous work has qualitatively found that large image generation models sometimes struggle with spatial positioning (Yu et al., 2022). We find this to be mostly true for Imagen, which performs poorly at associating objects with their position. CLIP performs even worse, performing no better than random chance. We found it prefers the caption with "right" in it over "left" 85% of the time, with it mostly ignoring the rest of the description.

4 Conclusion

While previous fine-grained analysis of diffusion models usually studies generated images qualitatively, our framework provides a new way of quantitatively studying them through evaluation on controlled classification tasks. We find Imagen is an effective and robust image classifier and is capable of performing attribute binding (while CLIP can't).

We hope our findings will inspire future work in using diffusion models for tasks other than generation. One direction is fine-tuning diffusion models on downstream tasks, e.g. evaluating Imagen as a classifier after further supervised training on the dataset. Our main comparison against CLIP is not direct in that the model architectures and parameter counts are different. As models become larger, a key question is how do the scaling laws (Hestness et al., 2017; Kaplan et al., 2020) of contrastive vs generative pre-training compare, which we leave for future work.

Ultimately, our method does not produce a very practical classifier, as it requires substantial compute when scoring many classes. Instead, we see the main value of this work is in revealing more about the abilities of large pre-trained diffusion models: our results suggest that generative pre-training may be a useful alternative to contrastive pre-training for text-image self-supervised learning.

REFERENCES

Amir Bar, Yossi Gandelsman, Trevor Darrell, Amir Globerson, and Alexei A Efros. Visual prompting via image inpainting. *arXiv preprint arXiv:2209.00647*, 2022.

Ryan Burgert, Kanchana Ranasinghe, Xiang Li, and Michael S Ryoo. Peekaboo: Text to Image Diffusion Models are Zero-Shot Segmentors. *arXiv preprint arXiv:2211.13224*, 2022.

Morris H DeGroot and Stephen E Fienberg. The comparison and evaluation of forecasters. *Journal of the Royal Statistical Society: Series D (The Statistician)*, 32(1-2):12–22, 1983.

Robert Geirhos, Patricia Rubisch, Claudio Michaelis, Matthias Bethge, Felix A Wichmann, and Wieland Brendel. ImageNet-trained CNNs are biased towards texture; increasing shape bias improves accuracy and robustness. *arXiv preprint arXiv:1811.12231*, 2018.

Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q Weinberger. On calibration of modern neural networks. In *International conference on machine learning*, pp. 1321–1330. PMLR, 2017.

Joel Hestness, Sharan Narang, Newsha Ardalani, Gregory Diamos, Heewoo Jun, Hassan Kianinejad, Md Patwary, Mostofa Ali, Yang Yang, and Yanqi Zhou. Deep learning scaling is predictable, empirically. *arXiv preprint arXiv:1712.00409*, 2017.

Jonathan Ho and Tim Salimans. Classifier-free Diffusion Guidance. *arXiv preprint arXiv:2207.12598*, 2022.

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

Jonathan Ho, William Chan, Chitwan Saharia, Jay Whang, Ruiqi Gao, Alexey Gritsenko, Diederik P Kingma, Ben Poole, Mohammad Norouzi, David J Fleet, et al. Imagen video: High definition video generation with diffusion models. *arXiv preprint arXiv:2210.02303*, 2022a.

Jonathan Ho, Tim Salimans, Alexey Gritsenko, William Chan, Mohammad Norouzi, and David J Fleet. Video diffusion models. *arXiv preprint arXiv:2204.03458*, 2022b.

Justin Johnson, Bharath Hariharan, Laurens Van Der Maaten, Li Fei-Fei, C Lawrence Zitnick, and Ross Girshick. Clevr: A diagnostic dataset for compositional language and elementary visual reasoning. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2901–2910, 2017.

Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. Scaling laws for neural language models. *arXiv preprint arXiv:2001.08361*, 2020.

Diederik Kingma, Tim Salimans, Ben Poole, and Jonathan Ho. Variational diffusion models. *Advances in neural information processing systems*, 34:21696–21707, 2021.

Diederik P Kingma and Max Welling. Auto-encoding Variational Bayes. *arXiv preprint arXiv:1312.6114*, 2013.

Martha Lewis, Qinan Yu, Jack Merullo, and Ellie Pavlick. Does clip bind concepts? probing compositionality in large image models. *arXiv preprint arXiv:2212.10537*, 2022.

Andrew Ng and Michael Jordan. On discriminative vs. generative classifiers: A comparison of logistic regression and naive bayes. *Advances in neural information processing systems*, 14, 2001.

Alex Nichol, Prafulla Dhariwal, Aditya Ramesh, Pranav Shyam, Pamela Mishkin, Bob McGrew, Ilya Sutskever, and Mark Chen. Glide: Towards photorealistic image generation and editing with text-guided diffusion models. In *International Conference on Machine Learning*, 2021.

Weili Nie, Brandon Guo, Yujia Huang, Chaowei Xiao, Arash Vahdat, and Anima Anandkumar. Diffusion Models for Adversarial Purification. *arXiv preprint arXiv:2205.07460*, 2022.

OpenAI. Prompt Engineering for Imagenet. *Github*, 2021a.

OpenAI. Prompts for Datasets. *Github*, 2021b.

John Platt et al. Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods. *Advances in large margin classifiers*, 10(3):61–74, 1999.

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*, pp. 8748–8763. PMLR, 2021.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *The Journal of Machine Learning Research*, 21(1):5485–5551, 2020.

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*, pp. 8821–8831. PMLR, 2021.

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.

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*, pp. 10684–10695, 2022.

Chitwan Saharia, William Chan, Huiwen Chang, Chris Lee, Jonathan Ho, Tim Salimans, David Fleet, and Mohammad Norouzi. Palette: Image-to-image diffusion models. In *ACM SIGGRAPH 2022 Conference Proceedings*, pp. 1–10, 2022a.

Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily Denton, Seyed Kamyar Seyed Ghasemipour, Burcu Karagol Ayan, S Sara Mahdavi, Rapha Gontijo Lopes, et al. Photorealistic Text-to-Image Diffusion Models with Deep Language Understanding. *arXiv preprint arXiv:2205.11487*, 2022b.

Christoph Schuhmann, Richard Vencu, Romain Beaumont, Robert Kaczmarczyk, Clayton Mullis, Aarush Katta, Theo Coombes, Jenia Jitsev, and Aran Komatsuzaki. Laion-400m: Open dataset of clip-filtered 400 million image-text pairs. *arXiv preprint arXiv:2111.02114*, 2021.

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

Yang Song and Stefano Ermon. Improved techniques for training score-based generative models. *Advances in neural information processing systems*, 33:12438–12448, 2020.

Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. *arXiv preprint arXiv:2011.13456*, 2020.

Jiahui Yu, Yuanzhong Xu, Jing Yu Koh, Thang Luong, Gunjan Baid, Zirui Wang, Vijay Vasudevan, Alexander Ku, Yinfei Yang, Burcu Karagol Ayan, et al. Scaling autoregressive models for content-rich text-to-image generation. *arXiv preprint arXiv:2206.10789*, 2022.

A PRELIMINARIES

We begin by recalling background knowledge on diffusion models (Sohl-Dickstein et al., 2015; Ho et al., 2020; Song et al., 2020; Song & Ermon, 2020) and recent advances on text-to-image diffusion models.

Diffusion Models: Diffusion models are latent variable generative models defined by a forward and reverse Markov chain. Given an unknown data distribution, $q(\boldsymbol{x}_0)$, over observations, $\boldsymbol{x}_0 \in \mathbb{R}^d$, the forward process corrupts the data into a sequence of noisy latent variables, $\boldsymbol{x}_{1:T} := \{\boldsymbol{x}_1, \boldsymbol{x}_2, \cdots, \boldsymbol{x}_T\}$, by gradually adding Gaussian noise with a fixed schedule defined as:

$$q(\boldsymbol{x}_{1:T}|\boldsymbol{x}_0) := \prod_{t=1}^{T} q(\boldsymbol{x}_t|\boldsymbol{x}_{t-1}) \tag{2}$$

where $q(\boldsymbol{x}_t|\boldsymbol{x}_{t-1}) := \mathsf{Normal}(\boldsymbol{x}_t; \sqrt{1-\beta_t}\boldsymbol{x}_{t-1}, \beta_t \boldsymbol{I})$.

The reverse Markov process gradually denoises the latent variables to the data distribution with learned Gaussian transitions starting from $\mathsf{Normal}(\boldsymbol{x}_T; 0, \boldsymbol{I})$ i.e.

$$p_{\boldsymbol{\theta}}(\boldsymbol{x}_{0:T}) := p(\boldsymbol{x}_T) \cdot \prod_{t=0}^{T-1} p_{\boldsymbol{\theta}}(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t)$$

$p_{\boldsymbol{\theta}}(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t) := \mathsf{Normal}\big(\boldsymbol{x}_{t-1}; \boldsymbol{\mu}_{\boldsymbol{\theta}}(\boldsymbol{x}_t, t), \boldsymbol{\Sigma}_{\boldsymbol{\theta}}(\boldsymbol{x}_t, t)\big)$. The aim of the denoising process is for the distribution for the forward process $\{\boldsymbol{x}_t\}_{t=0}^{T}$ to match that of the reverse process $\{\tilde{\boldsymbol{x}}_t\}_{t=0}^{T}$ i.e. the generative model $p_{\boldsymbol{\theta}}(\boldsymbol{x}_0)$ closely matches the data distribution $q(\boldsymbol{x}_0)$. Specifically, these models can be trained by optimizing the variational lower bound of the marginal likelihood (Kingma et al., 2021; Ho et al., 2020):

$$-\log p_\theta(\boldsymbol{x}_0) \leq -\mathsf{VLB}(\boldsymbol{x}) := \mathcal{L}_{\mathsf{Prior}} + \mathcal{L}_{\mathsf{Recon}} + \mathcal{L}_{\mathsf{Diffusion}}$$

$\mathcal{L}_{\mathsf{Prior}}$ and $\mathcal{L}_{\mathsf{Recon}}$ are the prior and reconstruction loss that can be estimated using standard techniques in the literature (Kingma & Welling, 2013). The diffusion loss, $\mathcal{L}_{\mathsf{Diffusion}}$, is:

$$\mathcal{L}_{\mathsf{Diffusion}} := \sum_{t=1}^{T} \mathbb{E}_{q(\boldsymbol{x}_t|\boldsymbol{x}_0)} \mathbb{D}_{\mathsf{KL}}\Big[q(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t, \boldsymbol{x}_0)||p_{\boldsymbol{\theta}}(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t)\Big]$$

Following Kingma et al. (2021), the (re-weighted) diffusion loss can be written in simplified form as:

$$\mathcal{L}_{\mathsf{Diffusion}} = \mathbb{E}_{\boldsymbol{x}_0, \boldsymbol{\varepsilon}, t}\Big[\boldsymbol{w}_t \|\boldsymbol{x}_0 - \tilde{\boldsymbol{x}}_{\boldsymbol{\theta}}(\boldsymbol{x}_t, t)\|_2^2\Big] \tag{3}$$

with $\boldsymbol{x}_0 \sim q(\boldsymbol{x}_0)$, $\boldsymbol{\varepsilon} \sim \mathsf{Normal}(0, \boldsymbol{I})$, and $t \sim \mathcal{U}([0, T])$. Here, $\boldsymbol{w}_t$ is a weight assigned to the timestep, and $\tilde{\boldsymbol{x}}_{\boldsymbol{\theta}}(\boldsymbol{x}_t, t)$ is the model's prediction of the observation $\boldsymbol{x}_0$ from the noised observation $\boldsymbol{x}_t$.

Conditional Diffusion Models and Classifier-Free Guidance: A conditional diffusion model conditions the model on alternate modalities like class labels, text prompts, segmentation masks or low-resolution images. Given a conditioning model, $\boldsymbol{c}_{\boldsymbol{\phi}}(\boldsymbol{y})$, that maps the conditioning input $\boldsymbol{y}$ into an encoded conditioning vector, a conditional diffusion model is trained using the following modified diffusion loss from Equation (3):

$$\mathcal{L}_{\mathsf{Diffusion}} = \mathbb{E}_{(\boldsymbol{x}_0, \boldsymbol{y}), \boldsymbol{\varepsilon}, t}\Big[\boldsymbol{w}_t \|\boldsymbol{x}_0 - \tilde{\boldsymbol{x}}_{\boldsymbol{\theta}}(\boldsymbol{x}_t, \boldsymbol{c}_{\boldsymbol{\phi}}(\boldsymbol{y}), t)\|_2^2\Big]$$

Classifier-free guidance (Ho & Salimans, 2022) is a technique to train a single diffusion model on both conditional and unconditional objectives by randomly dropping the conditioning vector, $\boldsymbol{c}_{\boldsymbol{\phi}}(\boldsymbol{y})$, during training with a certain probability. In this case, samples are generated using:

$$\tilde{\boldsymbol{x}}'_{\boldsymbol{\theta}}(\boldsymbol{x}_t, \boldsymbol{c}_{\boldsymbol{\phi}}) := (1+\lambda) \cdot \tilde{\boldsymbol{x}}_{\boldsymbol{\theta}}(\boldsymbol{x}_t, \boldsymbol{c}_{\boldsymbol{\phi}}) - \lambda \cdot \tilde{\boldsymbol{x}}'_{\boldsymbol{\theta}}(\boldsymbol{x}_t)$$

where λ is the guidance weight, $\tilde{\boldsymbol{x}}_{\boldsymbol{\theta}}(\boldsymbol{x}_t, \boldsymbol{c}_{\boldsymbol{\phi}})$ is the conditional model, and $\tilde{\boldsymbol{x}}'_{\boldsymbol{\theta}}(\boldsymbol{x}_t)$ is the unconditional model. Classifier-free guidance has been shown to be critical in generating high fidelity samples given a prompt (Saharia et al., 2022b; Ramesh et al., 2022; Ho et al., 2022b;a).

Text-to-Image Diffusion Models: Imagen is a text-to-image diffusion model comprising of a frozen T5 (Raffel et al., 2020) language encoder that encodes an input prompt into a sequence of embeddings, a 64×64 image diffusion model, and two two cascaded super-resolution diffusion models that generate 256×256 and 1024×1024 images. In the next section, we will use the generative process of Imagen to convert it into a classifier to study its generalization ability in the zero-shot classification setting.

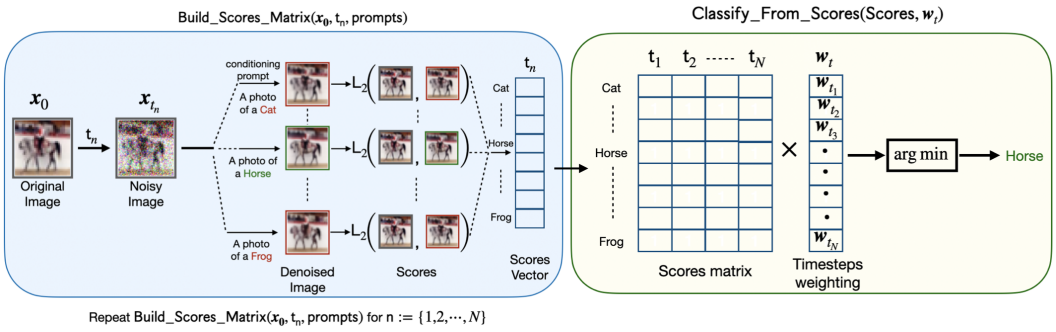


Figure 1: **Zero-Shot Classification using Imagen.** We first calculate scores for each image and label prompt across multiple time-steps to generate a scores matrix using Build_Scores_Matrix. Classify_From_Scores then classifies by aggregating the scores for each class using a weighting function over the time-steps and the image is assigned the class with the minimum aggregate score.

B Zero-Shot Classification Using Imagen

B.1 Improving Efficiency

Computing $\tilde{y}$ with naive Monte-Carlo estimation can be expensive because $\mathcal{L}_{\text{Diffusion}}$ has fairly high variance. Here, we propose techniques that reduce the compute cost of estimating the $\arg\min$ over classes. The key idea is to leverage the fact that we only need to compute the $\arg\min$ and do not require good estimates of the actual expectations.

Shared Noise: Differences between individual Monte-Carlo samples from $\mathcal{L}_{\text{Diffusion}}$ can of course be due to different t or forward diffusion samples from $q(\boldsymbol{x}_t|\boldsymbol{x}_{t-1})$, whereas we are only interested in the effect of the text conditioning $\boldsymbol{c}_\phi(\mathrm{T}(\mathrm{y}_k))$. We find far fewer samples are necessary when we use the *same* t and $\boldsymbol{x}_t$ across different classes. In other words, after sampling a $t \sim \mathcal{U}([0,1])$ and $\boldsymbol{x}_t \sim q(\boldsymbol{x}_t|\boldsymbol{x}_0)$, we score all classes against this noised image instead of a single one. As a result, the differences between these estimates are only due to the different text conditioning signals.

Candidate Class Pruning: Rather than using the same amount of compute to estimate the expectation for each class, we can further improve efficiency by discarding implausible classes early and dynamically allocating more compute to plausible ones. In particular, we maintain a set of candidate classes for the image being classified. After collecting a new set of scores for each candidate class, we discard classes that are unlikely to become the lowest-scoring (i.e. predicted) class with more samples. Since we are collecting paired samples (with the same t and $\hat{\boldsymbol{x}}_{i,t}$), we use a paired student's t-test to identify classes that can be pruned. Our scores, of course, do not exactly follow the standard assumptions of a student's t-test (e.g. they are not normally distributed), so we use a small p-value (0.002 in our experiments) and ensure each class is scored a minimum number of times (20 in our experiments) to minimize the chance of pruning away the correct class. The full procedure is shown in Algorithm 1.

Comparison: Figure 2 compares the number of samples needed to accurately classify CIFAR-100 images for different methods. Using shared noise and pruning greatly improves efficiency, requiring up to 500x less compute than naive scoring. Nevertheless, classifying with a diffusion model still typically takes 10s of scores per class on average, making the diffusion classifier expensive to use for datasets with many classes.

C Empirical Results

In this section, we will detail our analysis for the zero-shot classifier based on Imagen (§ Section 2) for a variety of tasks. These include classification on various vision datasets to study generalization capabilities on diverse domains, evaluations of robustness to conflicting cues between texture and

Algorithm 1 Diffusion model classification with pruning.

given: Example to classify $\boldsymbol{x}$, diffusion model w/ params θ, weighting function $\boldsymbol{w}$, hyperparameters min_scores, max_scores, cutoff_pval.

//Map from classes to weighted diffusion model scores.

scores $= \{\mathsf{y}_i : [] \text{ for } \mathsf{y}_i \in [\mathsf{y}_K]\}$

$n = 0$

while $|\text{scores}| > 1$ **and** $n <$ max_scores:

  $n = n + 1$

  $t \sim \mathcal{U}([0, 1])$

  $\boldsymbol{x}_t \sim q(\boldsymbol{x}_t|\boldsymbol{x})$

  for $\mathsf{y}_i \in$ scores:

    add $\boldsymbol{w}_t \|\boldsymbol{x} - \tilde{\boldsymbol{x}}_{\boldsymbol{\theta}}(\boldsymbol{x}_t, \boldsymbol{c}_{\boldsymbol{\phi}}(\mathrm{T}(\mathsf{y}_i)), t)\|_2^2$ to scores$[\mathsf{y}_i]$

  $\tilde{y} = \arg\min_{\mathsf{y}_i}$ scores$[\mathsf{y}_i]$.mean()

  if $n \geq$ min_scores:

    for $\mathsf{y}_i \in$ scores:

      if paired_ttest_pval(

       scores$[\tilde{y}]$, scores$[\mathsf{y}_i]$) $<$ cutoff_pval:

        remove y_i from scores.

return $\tilde{y}$

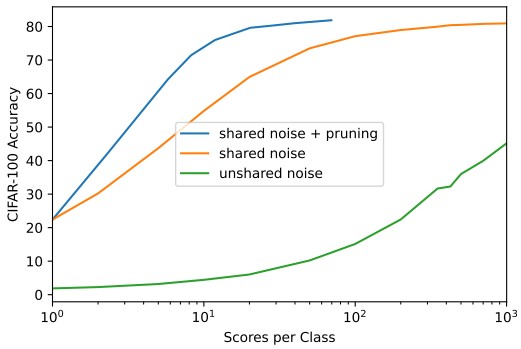


Figure 2: Comparison of efficiency improvements on CIFAR-100. Shared noise improves sample efficiency by roughly 100x and pruning by an additional 8-10x.

shape bias, obtaining calibrated scores, and studying attribute binding ability through targeted evaluation on synthetic data.

We compare Imagen with CLIP (Radford et al., 2021), which is widely used as a zero-shot classifier. Our main aim is to study the strengths and weaknesses of image-text representation learning via generative training as in Imagen and contrastive training as used for CLIP.

Imagen details: We use the 2B parameter Imagen model for 64×64 resolution text-to-image synthesis. It is trained using a batch size of 2048 and 2.5M training steps on a combination of internal datasets, with around 460M image-text pairs, and the publicly available Laion dataset (Schuhmann et al., 2021), with 400M image-text pairs. For simplicity, we only consider the low-resolution 64x64 model, although exploring the high-resolution ones would be interesting in the future. See § Appendix A for more details on Imagen.

CLIP details: CLIP encodes image features using a ViT-like transformer and uses a causal language model to get the text features. After encoding the image and text features to a latent space with identical dimensions, it evaluates a similarity score between these features. CLIP is pre-trained using contrastive learning. Here, we compare to the largest CLIP model (with a ViT-L/14@224px as the image encoder). The model is smaller than Imagen (400M parameters), but is trained for longer (12.8B images processed vs 5.B). While Imagen was trained primarily as a generative model, CLIP was primarily engineered to be transferred effectively to downstream tasks.

Experiment details: For each experiment, we obtain scores using the efficient scoring method in Algorithm 1. Nevertheless, due to the still-substantial compute cost, we use reduced-size datasets (4096 examples) for our experiments. We preprocess each dataset by normalizing the images, performing a central crop and then resizing the images to 64×64 resolution. We use $\mathsf{min_scores} = 20$, $\mathsf{max_scores} = 2000$, and $\mathsf{cutoff_pval} = 2 \times \mathsf{e}^{-3}$. Since we use a fixed single prompt template to obtain results for Imagen, we follow the same setting for CLIP to keep the results comparable. Therefore, our reported results are often lower than in the CLIP paper, which uses prompt ensembling.

C.1 IMAGE CLASSIFICATION

Setup We first evaluate the performance of Imagen at zero-shot classification. For this purpose, we consider 13 datasets from Radford et al. (2021) as reported in Table 1. We report the best accuracy achieved by Imagen using two weighting functions, $\boldsymbol{w}_t$: (a) *hand-engineered* weights across noise levels, $\boldsymbol{w}_t := \exp(-7t)$ and, (b) learned weights.

We use the prompt templates and class labels used by Radford et al. (2021), which renames some classes that confuse models (e.g. "crane $\rightarrow$ "crane bird"" in Imagenet) (OpenAI, 2021b). We use the first prompt from the list, except for Imagenet, where we use "A bad photo of a *label* " since this is a good prompt for both Imagen and CLIP (OpenAI, 2021a).

Since we use the low-resolution Imagen model, we obtain results using CLIP under two settings for a fair comparison. In the first setting, we resize all the datasets to 64×64 which serves as the base low-resolution dataset. Imagen uses this dataset directly. For CLIP, we subsequently upsample the images and resize them to 224×224 resolution, followed by a central crop and normalization as used in Radford et al. (2021). In the second setting, we directly resize all datasets to 224×224 resolution, followed by a central crop and normalization to obtain the best results possible using CLIP where it can take advantage of its higher input resolution.

Results Results are shown in Table 1. The first eight datasets (up through EuroSAT) are all originally of resolution 64×64 or less. On all these datasets, Imagen outperforms CLIP on classification accuracy under the same evaluation setting i.e. the models are conditioned on the same text prompts, etc. Imagen significantly outperforms CLIP on e.g. SVHN, which requires recognizing text in an image, reinforcing the qualitative observations that Imagen is good at including texts in images during generation (Saharia et al., 2022b).

The next five datasets use higher-resolution images. For some of these, taking advantage of CLIP's higher input resolution substantially improves results. It may be possible to get similar benefits from Imagen by incorporating scores from its superresolution models, which we leave for future work to explore.

We also notice that the boost from learned weightings is small, showing that our simple heuristic weighting function generalizes well across datasets. We found using no weights (i.e. $\boldsymbol{w}_t = 1$) hurts performance substantially (e.g., CIFAR100 accuracy drops to 45%), which is surprising because most diffusion models, including Imagen, are trained with no weights in their VLBs.

Comparing models: Imagen and CLIP have different model sizes and are trained on different datasets for different amounts of time, so the comparison is not direct. While ideally we would train models of the same size on the same data, this would be very expensive and challenging in practice; we instead used two strong existing pre-trained models. Our comparisons are geared towards highlighting the strengths and weaknesses of Imagen.

C.2 ROBUSTNESS

In the main text we showed that Imagen is more robust in its classification performance compared to CLIP and ResNet50 trained in a supervised fashion. One reason we believe for Imagen's superior performance is that the noising-denosing process of the diffusion model removes the texture bias commonly observed in supervised convolutional models, making it robust to presence of texture based cues. These findings are in line with Nie et al. (2022), who achieve state-of-the-art adversarial robustness through noising and then denoising adversarial examples with a diffusion model.

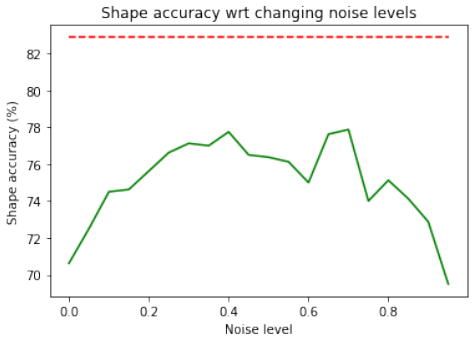


Figure 3: Imagen's performance based on restricted noise levels marginally effects classification accuracy on cue conflict dataset.

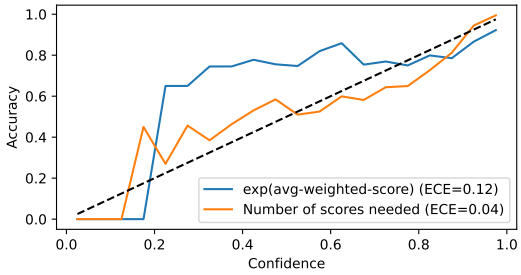


Figure 4: Model reliability diagram comparing confidence measures of Imagen on CIFAR-100.

We hypothesized that the amount of noise added has an effect on removing texture bias. To test this, we evaluated the shape-accuracy by restricting the noise levels to bins given by $[\mathtt{t}, \mathtt{t} + 0.05]$ where $\mathtt{t} \in \{0, 0.05, 0.10, \cdots, 0.90, 0.95\}$. We found that while the shape accuracy drops marginally when restricting to specific noise levels, it is overall robust to chosen noise levels as shown in Figure 3.

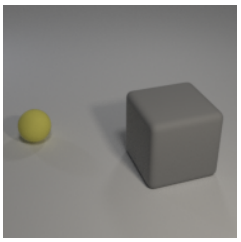
C.3 Calibration

It is desirable for classifiers, especially when used in the zero-shot setting with possibly out-of-domain examples, to be well calibrated. In other words, if a classifier predicts a label $\tilde{y}_i$ with probability p, the true label should be $\tilde{y}_i$ roughly $100 \cdot p\%$ of the time. However, the diffusion model classifier does not directly produce probabilities for classes. While $p(y_i = \mathtt{y}_k | \boldsymbol{x}_i)$ should roughly be proportional to the expectation in Equation (1) when exponentiated, in practice our estimates of the expectations are very noisy and do not provide well-calibrated scores. One culprit is early pruning, which causes many classes to have few sampled scores.

We propose a simple alternative that takes advantage of early pruning: we use the total number of diffusion model calls used for the image as a calibration measure. The intuition is that a harder example will require more scores to determine the $\arg\min$ class with good statistical significance. We show reliability diagrams (DeGroot & Fienberg, 1983) and report Expected Calibration Error (Guo et al., 2017) (ECE) for the methods in Figure 4. Using a small held-out set of examples, we apply temperature scaling (Guo et al., 2017) for the avg-weighted-score model and Platt scaling (Platt et al., 1999) for the number-of-scores model to map model outputs to confidences. Number of scores is fairly well-calibrated, showing it is possible to obtain reasonable confidences from diffusion model classifiers despite them not providing a probability distribution over classes.

C.4 Evaluating Attribute Binding on Synthetic Data

We have shown that Imagen can perform comparably to CLIP at few-shot classification, albeit with much higher compute requirements. Does Imagen have additional capabilities that are difficult to



Recognition (Shape)
A sphere.
A cylinder.

Binding (Color|Shape)
A yellow sphere.
A gray sphere.

Pair Binding (Size)
A small sphere and a large cube.
A large sphere and a small cube.

Spatial
On the left is a yellow sphere.
On the right is a yellow sphere.

Figure 5: Examples from the attribute binding tasks.

obtain through contrastive pre-training? We hypothesize that one such area may be in compositional generalization, and specifically compare Imagen and CLIP at attribute binding. Text-to-image generative models have shown emergent compositional generalization at large enough scale, being able to combine different concepts to handle prompts such as "a chair shaped like an avacado" (Ramesh et al., 2021). Attribute binding is a key piece of compositional reasoning: to understand novel combinations of concepts, one must bind the concepts together and treat them as a whole. For example in the statement "a yellow sphere and a gray cube" we understand the sphere is yellow and the cube is gray, not the other way around.

We test attribute binding in Imagen and CLIP on synthetic datasets. While other work has examined attribute binding in text-to-image models by qualitatively examining model generations (Nichol et al., 2021; Yu et al., 2022), our Imagen classifier offers a way of studying the question quantitatively. We hope in the future, this sort of study will be useful for comparing the abilities of generative image models at a fine-grained level.

Dataset Construction: We use a setup similar to Lewis et al. (2022), where images are generated based on the CLEVR (Johnson et al., 2017) visual question answering dataset. CLEVR images contain various object (cubes, cylinders, and spheres) with various attributes (different sizes, colors, and materials). We use a modified version of the CLEVR rendering script that generates images containing two objects of different shapes. From these images, we construct binary classification tasks of 1000 examples each.

- **Recognition tasks** determine if the model can identify basic properties of the image by scoring an attribute in the image against an attribute not present.
- **Single-object binding tasks** test if the model binds a given attribute to the correct object.
- **Pair binding tasks** are easier attribute binding tasks where the model is provided information about both objects in the image instead of just one.
- **Spatial tasks** test if the model is capable of binding objects and their positions in the image.

See Figure 5 for some examples. We explored more sophisticated prompts than in the figure (e.g., "A blender rendering of two objects, one of which is a yellow sphere."), but they didn't substantially change results. Scores for Imagen and CLIP at these tasks is shown in Table 2.