Self Supervised Learning Using Controlled Diffusion Image Augmentation

Judah Goldfeder
Department of Computer Science
Columbia University
jag2396@columbia.edu

Patrick Puma
Department of Mechanical Engineering
Harvard University
ppuma@g.harvard.edu

Gabe Guo
Department of Computer Science
Stanford University
gabeguo@stanford.edu

Gabriel Trigo
Department of Computer Science
Columbia University
ggt2112@columbia.edu

Hod Lipson
Data Science Institute
Columbia University
hod.lipson@columbia.edu

Abstract

While synthetic data generated through diffusion models has been shown to improve performance across various tasks, existing approaches face two challenges: the necessity of fine-tuning a diffusion model for a specific dataset is often expensive, and the domain gap between real and synthetic data limits synthetic data's usefulness, especially in fine-grained classification settings. To mitigate these shortcomings, we developed CDaug, a novel approach to data augmentation utilizing controlled diffusion. Instead of utilizing diffusion models to generate wholly new images, we take a self-supervised approach and condition the generated images on existing data, allowing us to create high quality synthetic images/augmentations that capture the semantic priors and underlying structure of the data while infusing meaningful and novel variations with no human intervention. We developed a pipeline that utilizes ControlNet, conditioned on the original data, and captions generated by the multi-modal LLM LLaVA2 to guide the generative process. Our work uses open-source models, does not require fine-tuning, and is modular. We demonstrate improved performance across 7 fine-grained datasets, in both few-shot and full dataset settings, across many architectures.

1 Introduction

In the past few years, the capabilities of Generative AI have increased dramatically, in multiple domains. While it is clear that Generative AI is here to stay, there remains an important question: to what degree can content produced by AI be used to further its own development [30]? For AI generated data to be successful, we must cross a major hurdle: the sim-to-real gap. As the popular saying goes "Garbage in, Garbage out", and if our AI generated content is not properly representative of the real data, not only will its use not be helpful, but it may even provide harm. Instead of viewing generative AI as a tool for data creation, we view it as a tool for self-supervised data augmentation. That is, instead of using data produced from scratch, where, especially in fine-grained settings, we may not know the correct label for the new data, we use existing real data to condition the generative

38th Conference on Neural Information Processing Systems (NeurIPS 2024).

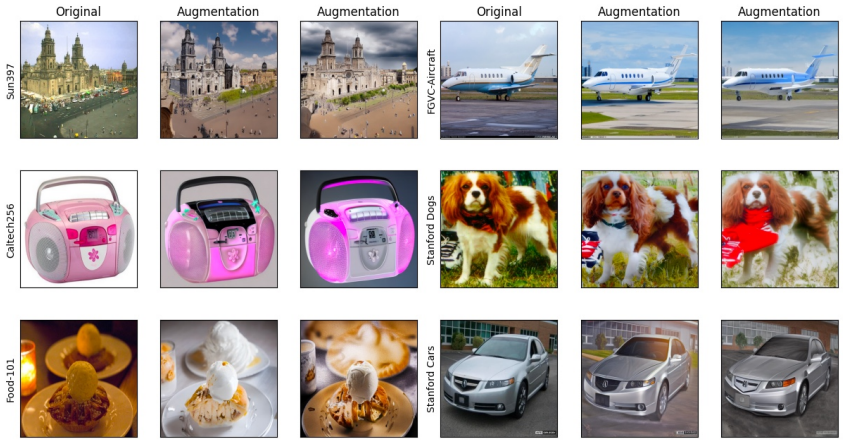

Figure 1: Example image augmentations using our pipeline on six datasets.

process, ensuring that we capture the semantic priors and underlying structure of the original data distribution while infusing meaningful and novel variations in the data. This gives us new input-output pairs where we can know with high certainty the correct label of the synthetic data. Motivated by this vision, we designed a novel pipeline that utilizes fully open-source models to take an input image, generate a caption, extract the necessary features from the original image, and create an augmentation that retains the semantic and underlying structure of the original image. Because we are using open-source models, we designed our pipeline to be fully modular, allowing the models to be swapped out per the user's needs. Our focus in this work is to tackle the problem as it relates to image classification, but we believe that the general hypothesis may prove useful across a wide range of domains.

2 Related Works

Data augmentation aims to reduce model overfitting by applying random image transformations that preserve the semantics of the original, while being different enough to provide the model with more of a challenge. Recently, generative diffusion models [29], especially text-to-image models, have made massive progress in generating photo realistic images [16, 19, 21]. Trained on internet-scale data [23], they have been used as an effective augmentation method [1, 22, 8, 24, 20], often using only simple class-agnostic prompts to guide generation for each class or even just the class names. However, synthetic data still falls short of their real counterpart across the board, which suggests that there is still a domain gap between real and synthetic data [28]. We are the first to provide an in-depth study on how to apply these models as a general augmentation operator and to incorporate both recent advances in image-to-image conditioning and multimodal LLMs, using both the image and caption of the image, to condition the generation process. The strength of CDaug over prior work is that it minimizes the domain gap between real and synthetic images, can generate synthetic variations of concepts unknown to it *without any fine-tuning*, and is suitable for fine-grained datasets, where minor semantic details are very important and diffusion models may not be able to generate to the required level of nuance.

3 Methods

We use canny edge detector for the prepocessor, LLaVA2 for the captioning model, and ControlNet canny for our pipeline. We chose these models as they are the state of the art at the time of writing, but the pipeline was designed with modularity in mind and each of these blocks can be replaced per the user's needs. For example, in the preprocessing steps, depending on the available resources, the preprocessor can be swapped for an AI-powered model that outputs a segmentation map, depth map, etc. to condition the ControlNet model.

CDaug The CDaug pipeline with an example image is demonstrated in Figure 2. We pre-process each image with the canny edge detector [3] as it provides an effective augmentation without incurring

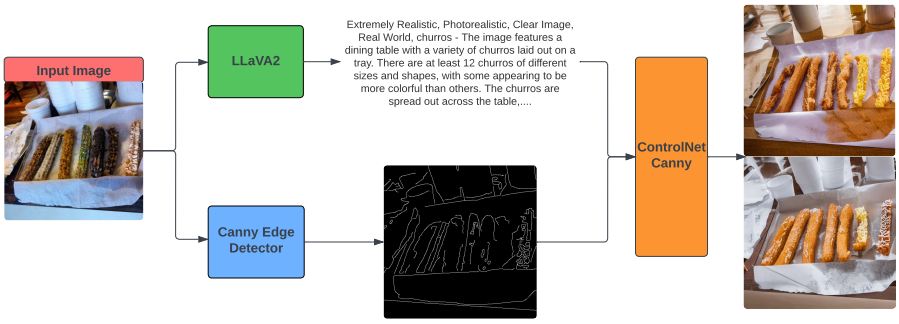

Figure 2: Our complete pipeline with ControlNet Canny, LLaVA2, and Canny edge detector.

a large processing cost. In parallel to the conditioning image, we utilized an additional form of conditioning in the form of detailed image captions. To automate the captioning of the images, we utilize LLaVA2, a multimodal language model based on Llama2 that given an input image and a prompt will generate an answer using the image [13]. Through qualitative testing, we found that the LLaVA2 performs similarly to other multi-modal LLMs such as Bert and BLIP-2. To create unique variations while retaining the key features from the original image, we utilize ControlNet from [31]. This allows for rapid customization of the diffusion model, without losing information stored in the pre-trained weights. Once we have the conditioning image and the caption, we input the caption and the conditioning image into ControlNet Canny to create as many variations as is required. These inputs allow our method to create unique images ensuring that the new variations have realistic structure and features. ControlNet Canny takes the two inputs described above: the edge map and detailed caption. With these two inputs, we capture the underlying structure of the original image, while infusing meaningful and novel variations in the data from the description generated by the LLM.

Augmenting Datasets To create a full augmented dataset, we generated two distinct augmentations for each image, thereby diversifying the dataset without an excessive increase in computational demand. For our fewshot dataset experiments, we created 15 variations per image. We specified both a positive (`a_prompt`) and negative (`n_prompt`) text prompt The `a_prompt` parameter was assigned the output of the LLM, as described above. Conversely, the `n_prompt` was `"multiple, mushed, low quality, cropped, worst quality"` as its value. A square image resolution of 512x512 pixels was chosen, and the DDIM (Denoising Diffusion Implicit Models) process was configured to perform 20 iterations, ensuring that the augmentations would possess sufficient quality, while still being computationally cheap to generate. On a single 24GB NVIDIA 3090 GPU, we could generate about 15 augmentations per second.

4 Experiments

To test the strength of our approach, we chose to focus on fine-grained settings, including both few-shot and full dataset settings. We augmented seven datasets: Caltech256 [6], Sun397 [27], Oxford IIT-Pets[18], FGVC Aircraft[15], Stanford Cars[12], Stanford Dogs[11], and Food101[2]. Examples can be seen in Figure 1. For the few-shot experiments, we used a pretrained Resnet50, training using SGD, using cosine annealing with a learning rate ranging from 1e-2 to 1e-7. The augmented dataset has 15 augmentations per image, and thus is 16 times larger than the baseline dataset. To compensate for this, all baseline methods were trained for 16 times more epochs than the augmented version, to ensure a fair comparison. All experiments were run for at least 3 seeds, with the mean and standard deviation reported. To ensure a strong baseline, we applied state-of-the-art augmentation techniques to the baseline, including rotation, random-crop, mirroring, color-jitter, and auto-augment [5]. We present the results for 5 way classification in Table 1, and 10 way in Table 2.

4.1 Full Dataset Results

For the full dataset experiments, we present results on a variety of architectures, including Resnets[7], VGG [25], EfficientNet[26], Visformer[4], Swin Transformer[14], MobileNet[9], and DenseNet[10],

Table 1: 5-Way Few-Shot Classification Results

	Shots	1	2	5	10
Caltech	Baseline	57.57 ± 1.56	65.57 ± 0.39	79.07 ± 1.03	**84.02 ± 1.4**
	CDaug	**62.53 ± 1.4**	**71.9 ± 2.02**	**82.06 ± 1.06**	83.19 ± 1.04
Cars	Baseline	42.78 ± 2.26	52.4 ± 2.7	75.95 ± 1.54	86.07 ± 0.0
	CDaug	**47.41 ± 1.32**	**60.03 ± 0.62**	**76.45 ± 1.24**	**89.88 ± 0.24**
Aircraft	Baseline	30.93 ± 1.02	29.92 ± 1.73	35.54 ± 1.47	43.77 ± 1.5
	CDaug	**36.95 ± 1.58**	**39.16 ± 0.49**	**40.36 ± 1.3**	**51.34 ± 0.96**
Pets	Baseline	28.25 ± 0.66	34.94 ± 0.59	44.71 ± 0.25	52.54 ± 0.68
	CDaug	**31.59 ± 0.41**	**42.04 ± 0.74**	**55.96 ± 0.5**	**67.2 ± 0.57**
Dogs	Baseline	26.15 ± 0.68	28.57 ± 0.91	54.07 ± 0.63	75.87 ± 1.09
	CDaug	**31.23 ± 1.43**	**40.4 ± 0.95**	**56.82 ± 1.26**	**76.35 ± 0.23**
Food	Baseline	33.39 ± 0.14	46.27 ± 0.39	63.71 ± 0.23	71.39 ± 0.42
	CDaug	**42.35 ± 0.91**	**54.75 ± 0.76**	**69.36 ± 0.74**	**76.29 ± 0.2**
Sun	Baseline	33.5 ± 3.29	48.07 ± 1.55	51.93 ± 0.24	60.8 ± 0.41
	CDaug	**35.68 ± 3.69**	**45.06 ± 0.24**	**52.6 ± 1.32**	**61.3 ± 1.08**

Table 2: 10-Way Few-Shot Classification Results

	Shots	1	2	5	10
Caltech	Baseline	45.73 ± 0.97	57.58 ± 0.39	77.82 ± 0.78	79.89 ± 0.85
	CDaug	**52.62 ± 1.36**	**71.07 ± 1.47**	**78.65 ± 0.71**	**85.18 ± 1.52**
Cars	Baseline	29.47 ± 0.31	35.97 ± 0.42	57.86 ± 0.96	74.4 ± 0.84
	CDaug	**32.1 ± 0.81**	**40.25 ± 3.15**	**65.76 ± 0.62**	**79.01 ± 0.35**
Aircraft	Baseline	18.22 ± 0.51	22.22 ± 1.12	25.02 ± 1.16	33.13 ± 0.62
	CDaug	**20.22 ± 0.93**	**25.23 ± 0.25**	**28.53 ± 0.49**	**35.43 ± 1.92**
Pets	Baseline	31.34 ± 0.91	42.84 ± 0.62	53.27 ± 0.77	62.2 ± 0.43
	CDaug	**32.49 ± 0.72**	**45.11 ± 0.49**	**60.54 ± 1.18**	**68.12 ± 0.17**
Dogs	Baseline	18.26 ± 0.5	28.14 ± 1.26	45.23 ± 1.45	65.37 ± 0.99
	CDaug	**22.33 ± 0.17**	**33.15 ± 0.11**	**48.19 ± 0.29**	**67.16 ± 0.93**
Food	Baseline	23.71 ± 0.42	37.63 ± 0.66	55.99 ± 0.42	69.51 ± 0.25
	CDaug	**31.81 ± 0.17**	**46.79 ± 0.17**	**63.36 ± 0.37**	**71.95 ± 0.35**
Sun	Baseline	19.46 ± 1.59	27.92 ± 1.05	44.26 ± 0.13	59.18 ± 0.21
	CDaug	**20.98 ± 0.82**	**30.01 ± 0.35**	**49.29 ± 0.13**	**61.01 ± 0.44**

to demonstrate the versatility and robustness of our method. All models were trained from scratch using SGD, using cosine annealing with a learning rate ranging from 1e-2 to 1e-4. The augmented dataset has 2 augmentations per image, and thus is 3 times larger than the baseline dataset. To compensate for this, all baseline methods were trained for 3 times more epochs than the augmented version, to ensure a fair comparison. The baselines were trained for 900 epochs, and the augmented version for 300 epochs. In all cases, the highest validation accuracy over any epoch is reported. To ensure a strong baseline, we applied state-of-the-art augmentation techniques to the baseline, including rotation, random-crop, mirroring, color-jitter, and auto-augment [5]. We present the results in Table 3. Our method rarely does much worse than the baseline, and in most cases gives a modest to significant improvement. We also note that of the 8 comparisons in which our method did slightly worse, in 7 of them we still beat the baseline in top 5 accuracy.

5 Conclusion

We present a novel self-supervised image augmentation method, building and improving upon earlier work, that leverages both edge maps and LLM generated image captions, to create nuanced variations. Our method has robust results on few-shot learning, and is even beneficial when used in settings where data is not sparse, something previous work has struggled with. The strong conditioning allows us to bridge the domain gap, while at the same time not requiring expensive fine tuning, nor assuming that the diffusion model has any knowledge whatsoever of the image category in question, and is still successful in very fine-grained settings.

Table 3: Full Dataset Results

	Model	Resnet50	Resnet101	Vgg19	Eff. Net	Visformer	Swin	MobileNet	DenseNet
Caltech	Baseline	72.37	73.62	67.4	71.79	68.83	63.95	66.48	75.74
	CDaug	**76.49**	**77.64**	**70.82**	**73.85**	**73.15**	**69.55**	**68.33**	**78.1**
Cars	Baseline	86.78	88.16	87.22	86.75	83.37	75.43	80.8	91.08
	CDaug	**91.02**	**90.95**	**89.61**	**88.56**	**87.4**	**82.32**	**82.7**	**92.2**
Aircraft	Baseline	75.23	75.91	**88.8**	81.25	72.61	60.88	70.24	80.53
	CDaug	**82.33**	**81.1**	88.2	**81.76**	**74.67**	**71.74**	**74.17**	**83.29**
Dogs	Baseline	66.49	70.148	**68.63**	**64.17**	**64.65**	52.1	**58.6**	**70.44**
	CDaug	**68.74**	**70.4**	66.05	62.45	64.36	**56.5**	58.3	70.21
Pets	Baseline	69.22	70.72	**83.17**	73.59	73.02	58.54	67.35	**80.16**
	CDaug	**71.07**	**74.03**	81.28	**74.41**	**76.24**	**61.0**	**68.46**	79.34

References

[1] Shekoofeh Azizi, Simon Kornblith, Chitwan Saharia, Mohammad Norouzi, and David J Fleet. Synthetic data from diffusion models improves imagenet classification. *arXiv preprint arXiv:2304.08466*, 2023.

[2] Lukas Bossard, Matthieu Guillaumin, and Luc Van Gool. Food-101–mining discriminative components with random forests. In *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part VI 13*, pages 446–461. Springer, 2014.

[3] John Canny. A computational approach to edge detection. *IEEE Transactions on pattern analysis and machine intelligence*, (6):679–698, 1986.

[4] Zhengsu Chen, Lingxi Xie, Jianwei Niu, Xuefeng Liu, Longhui Wei, and Qi Tian. Visformer: The vision-friendly transformer. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 589–598, 2021.

[5] 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.

[6] Gregory Griffin, Alex Holub, and Pietro Perona. Caltech-256 object category dataset. 2007.

[7] 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.

[8] Ruifei He, Shuyang Sun, Xin Yu, Chuhui Xue, Wenqing Zhang, Philip Torr, Song Bai, and Xiaojuan Qi. Is synthetic data from generative models ready for image recognition? *arXiv preprint arXiv:2210.07574*, 2022.

[9] Andrew G Howard, Menglong Zhu, Bo Chen, Dmitry Kalenichenko, Weijun Wang, Tobias Weyand, Marco Andreetto, and Hartwig Adam. Mobilenets: Efficient convolutional neural networks for mobile vision applications. *arXiv preprint arXiv:1704.04861*, 2017.

[10] Forrest Iandola, Matt Moskewicz, Sergey Karayev, Ross Girshick, Trevor Darrell, and Kurt Keutzer. Densenet: Implementing efficient convnet descriptor pyramids. *arXiv preprint arXiv:1404.1869*, 2014.

[11] Aditya Khosla, Nityananda Jayadevaprakash, Bangpeng Yao, and Fei-Fei Li. Novel dataset for fine-grained image categorization: Stanford dogs. In *Proc. CVPR workshop on fine-grained visual categorization (FGVC)*, volume 2. Citeseer, 2011.

[12] Jonathan Krause, Michael Stark, Jia Deng, and Li Fei-Fei. 3d object representations for fine-grained categorization. In *Proceedings of the IEEE international conference on computer vision workshops*, pages 554–561, 2013.

[13] Haotian Liu, Chunyuan Li, Yuheng Li, and Yong Jae Lee. Improved baselines with visual instruction tuning. *arXiv preprint arXiv:2310.03744*, 2023.

[14] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 10012–10022, 2021.

[15] Subhransu Maji, Esa Rahtu, Juho Kannala, Matthew Blaschko, and Andrea Vedaldi. Fine-grained visual classification of aircraft. *arXiv preprint arXiv:1306.5151*, 2013.

[16] 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. *arXiv preprint arXiv:2112.10741*, 2021.

[17] M-E Nilsback and Andrew Zisserman. A visual vocabulary for flower classification. In *2006 IEEE computer society conference on computer vision and pattern recognition (CVPR'06)*, volume 2, pages 1447–1454. IEEE, 2006.

[18] Omkar M Parkhi, Andrea Vedaldi, Andrew Zisserman, and CV Jawahar. Cats and dogs. In *2012 IEEE conference on computer vision and pattern recognition*, pages 3498–3505. IEEE, 2012.

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

[20] 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.

[21] Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily L Denton, Kamyar Ghasemipour, Raphael Gontijo Lopes, Burcu Karagol Ayan, Tim Salimans, et al. Photorealistic text-to-image diffusion models with deep language understanding. *Advances in neural information processing systems*, 35:36479–36494, 2022.

[22] Mert Bülent Sarıyıldız, Karteek Alahari, Diane Larlus, and Yannis Kalantidis. Fake it till you make it: Learning transferable representations from synthetic imagenet clones. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8011–8021, 2023.

[23] Christoph Schuhmann, Romain Beaumont, Richard Vencu, Cade Gordon, Ross Wightman, Mehdi Cherti, Theo Coombes, Aarush Katta, Clayton Mullis, Mitchell Wortsman, et al. Laion-5b: An open large-scale dataset for training next generation image-text models. *Advances in Neural Information Processing Systems*, 35:25278–25294, 2022.

[24] Jordan Shipard, Arnold Wiliem, Kien Nguyen Thanh, Wei Xiang, and Clinton Fookes. Diversity is definitely needed: Improving model-agnostic zero-shot classification via stable diffusion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 769–778, 2023.

[25] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[26] Mingxing Tan and Quoc Le. Efficientnet: Rethinking model scaling for convolutional neural networks. In *International conference on machine learning*, pages 6105–6114. PMLR, 2019.

[27] Jianxiong Xiao, James Hays, Krista A Ehinger, Aude Oliva, and Antonio Torralba. Sun database: Large-scale scene recognition from abbey to zoo. In *2010 IEEE computer society conference on computer vision and pattern recognition*, pages 3485–3492. IEEE, 2010.

[28] Shin'ya Yamaguchi and Takuma Fukuda. On the limitation of diffusion models for synthesizing training datasets. *arXiv preprint arXiv:2311.13090*, 2023.

[29] Ling Yang, Zhilong Zhang, Yang Song, Shenda Hong, Runsheng Xu, Yue Zhao, Wentao Zhang, Bin Cui, and Ming-Hsuan Yang. Diffusion models: A comprehensive survey of methods and applications. *ACM Computing Surveys*, 56(4):1–39, 2023.

[30] Zuhao Yang, Fangneng Zhan, Kunhao Liu, Muyu Xu, and Shijian Lu. Ai-generated images as data source: The dawn of synthetic era. *arXiv preprint arXiv:2310.01830*, 2023.

[31] Lvmin Zhang, Anyi Rao, and Maneesh Agrawala. Adding conditional control to text-to-image diffusion models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3836–3847, 2023.

A Additional Results

We also include results on all-way classification (few shot learning where all classes are included).

Table 4: All-Way Few-Shot Classification Results

	Shots	1	2	5	10
Caltech	Baseline	19.4 ± 0.58	28.49 ± 0.33	42.77 ± 0.14	50.4 ± 0.45
	CDaug	**24.43 ± 0.25**	**35.27 ± 1.01**	**48.49 ± 0.27**	**55.58 ± 0.43**
Cars	Baseline	5.51 ± 0.06	9.95 ± 0.09	28.41 ± 0.01	55.82 ± 0.12
	CDaug	**6.31 ± 0.03**	**12.92 ± 0.1**	**37.1 ± 0.05**	**62.76 ± 0.03**
Aircraft	Baseline	**8.22 ± 0.27**	12.81 ± 0.09	23.38 ± 0.01	39.9 ± 0.12
	CDaug	7.17 ± 0.06	**14.02 ± 0.22**	**27.8 ± 0.1**	**44.57 ± 0.07**
Pets	Baseline	14.45 ± 0.05	22.57 ± 0.19	38.17 ± 0.1	50.11 ± 0.37
	CDaug	**16.37 ± 0.23**	**26.32 ± 0.67**	**41.32 ± 1.04**	**54.0 ± 0.32**
Dogs	Baseline	4.94 ± 0.07	9.17 ± 0.17	**18.88 ± 0.05**	30.91 ± 0.07
	CDaug	**5.92 ± 0.07**	**10.34 ± 0.18**	18.75 ± 0.46	**29.71 ± 0.29**
Food	Baseline	12.4 ± 0.11	20.96 ± 0.07	36.58 ± 0.16	**48.39 ± 0.02**
	CDaug	**14.54 ± 0.05**	**23.32 ± 0.05**	**37.82 ± 0.03**	48.0 ± 0.08

B Further Limitations

Our work has several limitations. On some datasets, such as the Pets dataset, our results seem more modest, and further exploration is needed to understand when and where this method is best applicable.

Further, if the diffusion model has been trained on the dataset already, it will outperform our augmentation model. To show this, we performed controlled and uncontrolled augmentation on the Stanford Cars dataset and Aircraft Dataset. The results in Table 5.

Table 5: All-Way Controlled vs Uncontrolled Augmentation Comparison

	Shots	1	2	5	10
Cars	Baseline	5.51 ± 0.06	9.95 ± 0.09	28.41 ± 0.01	55.82 ± 0.12
	CDaug Control	6.31 ± 0.03	12.92 ± 0.1	37.1 ± 0.05	**62.76 ± 0.03**
	CDaug No Control	**9.6 ± 0.14**	**18.58 ± 0.04**	**39.52 ± 0.15**	57.2 ± 0.2
	Shots	1	2	5	10
Aircraft	Baseline	**8.22 ± 0.27**	12.81 ± 0.09	23.34 ± 0.06	39.9 ± 0.12
	DiffAug Control	7.17 ± 0.06	**14.02 ± 0.22**	**27.8 ± 0.1**	**44.79 ± 0.15**
	DiffAug No Control	5.82 ± 0.1	9.02 ± 0.04	15.74 ± 0.1	23.07 ± 0.15

What is noteworthy is the massive difference in performance between these two datasets. The diffusion model seems very familiar with the nuances of car makes and models, and thus, when provided with the name of the car model, as well as a description of the scene provided by the LLM, it is able to generate correct renderings of the car in completely different angles and orientations. In this case, providing the edge map as additional conditioning seems like a limiting factor. However, the diffusion model is much less familiar with detailed knowledge of aircraft varieties, and when it is not provided with an edge map, the aircraft generated are incorrect and do not resemble the original as shown in Figure 3.

We conclude that, while in some domains where the diffusion model is well trained, removing the image conditioning can lead to better performance, but this involves a large risk, and is especially

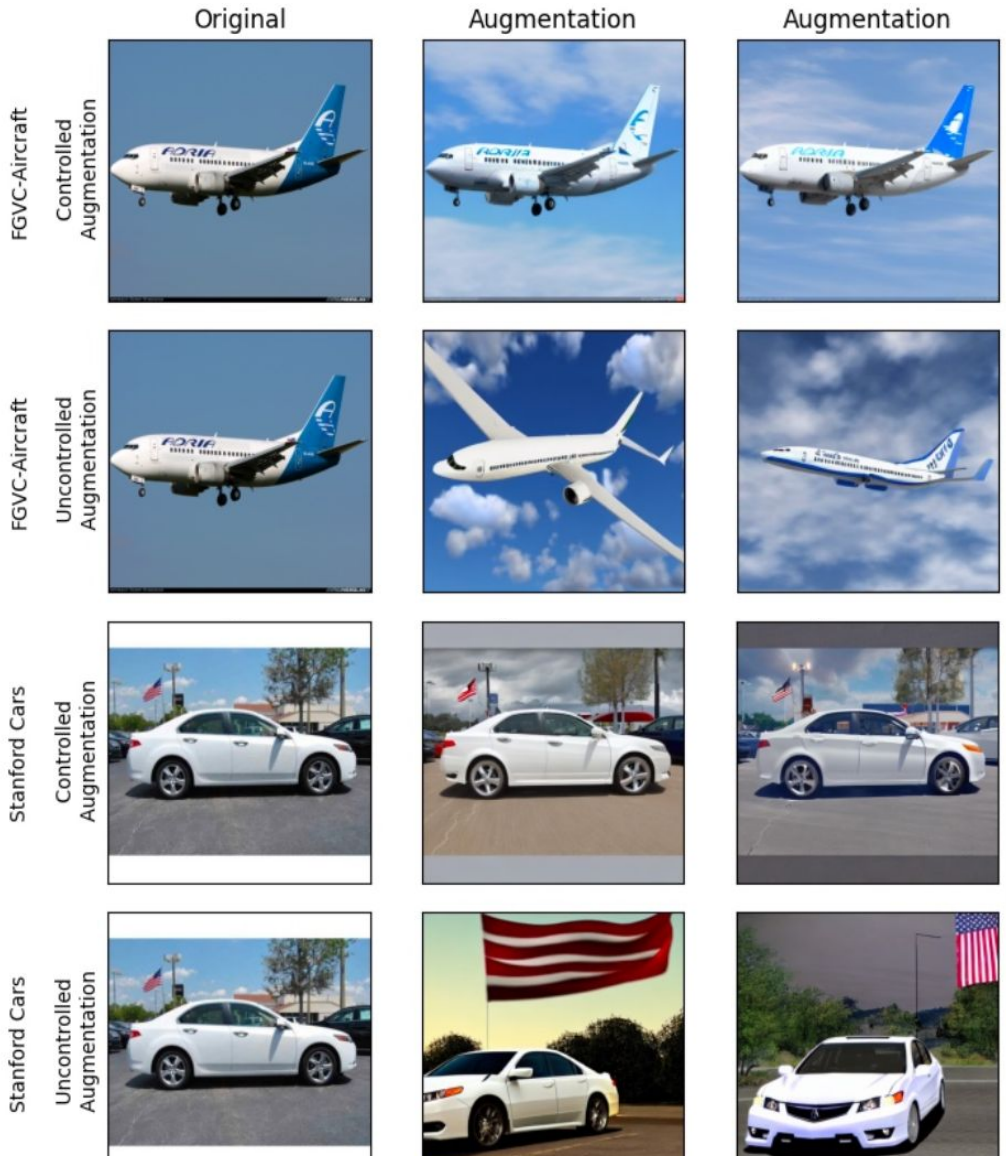

Figure 3: Comparison of controlled and uncontrolled augmentation for planes and cars.

unsuitable in many real world applications where the few-shot images are unlikely to be as well sampled as the makes and models of cars. Our method, while somewhat more limited, has stronger performance guarantees: it seems to improve over the baseline in almost all examples, and even when it falls short, it does so only minorly.

Another important limitation is that our pipeline does not take color into account, which, for many fine grained datasets, such as Flowers [17] is of the utmost importance. This can lead to low quality augmentations in these cases.

C Ablation Study

We ran an ablation where we removed all existing augmentation techniques from the baseline (rotation, cropping, etc), to show how our method builds on top of existing augmentation. We present the results in Table 6.

Table 6: Full Dataset: SOTA Augmmentation vs No Augmentation Ablation

	Shots	No Aug.	SOTA Aug.
Caltech	Baseline	34.0 ± 0.0	72.37 ± 0.09
	CDaug	$\mathbf{45.2 \pm 0.0}$	$\mathbf{76.49 \pm 0.36}$
Cars	Baseline	7.84 ± 0.11	86.78 ± 0.26
	CDaug	$\mathbf{13.45 \pm 0.28}$	$\mathbf{91.02 \pm 0.31}$
Aircraft	Baseline	22.21 ± 0.07	69.23 ± 0.85
	CDaug	$\mathbf{27.69 \pm 0.6}$	$\mathbf{71.07 \pm 0.71}$
Pets	Baseline	17.4 ± 0.36	66.49 ± 0.14
	CDaug	$\mathbf{22.99 \pm 0.75}$	$\mathbf{68.74 \pm 0.32}$

Clearly, our approach does not replace these basic techniques, although it certainly does help even more in their absence.