PerCoGen - Personalized & Controlled Image Generation

Lakshay Tyagi
lt2504@nyu.edu

Abstract

This work explores the design space of the current state-of-the-art diffusion models to generate controllable and personalizable approaches for human-centered image generation. Text to Image Diffusion models hold the current state-of-the-art in generation diversity, quality, and controllability. In generative models, the ability to exert precise control over the generated output is paramount, particularly for digital art and design applications. Recent advancements such as ControlNet and DreamBooth have made incredible progress in this direction. ControlNet excels at specifying structural priors for image generation (e.g., pose), while Dreambooth, a finetuning procedure, focuses on adding custom objects to the model's vocabulary. However, a combined approach that integrates both these features remains under-explored. This work begins by proposing a straightforward amalgamation of both these approaches, indicating the synergy between the finetuning strategy of personalization approaches (DreamBooth) and structural control approaches (ControlNet). In doing so, the necessity of grounding between the textual control (image description, word embeddings etc.), the primary target of personalization approaches, and structural control (pose, HED maps, etc), the primary target of control approaches becomes clear. This grounding between textual control (object/person identity) and structural control (object/person pose or location) allows for superior control in image generation. To this end, this work extends ControlNet with an extra SelfAttention layer, allowing the grounding to occur. The contributions of this work are twofold 1) Separate finetuning required for personalization and structural control approaches can be combined into a single step. 2) A newly proposed architecture called text-grounded ControlNet can associate structural queues (e.g., poses) with textual queries (e.g., entity name). The code for the implementation can be found here `Github`.

I. Introduction

A. *Controlled Generation*

Current diffusion models are limited to text inputs, making it difficult to specify additional inputs that could help specify the desired image in more detail. Furthermore, additional control input helps prevent hallucination and allows for fine-grained control. This mechanism also allows editing existing images as an additional form of control. The addition of control can make diffusion models more powerful and easy to use.

*1) **ControlNet**:* ControlNet [9] is a neural network structure to control diffusion models by adding extra conditions. Its function is to allow input of a conditioning image, which can then be used to manipulate image generation. It provides a way to augment Stable Diffusion with conditional inputs such as scribbles, edge maps, segmentation maps, pose key points, etc.

In addition, a ControlNet model can be trained with small datasets on consumer GPU. Then, the model can be augmented with any pre-trained Stable Diffusion models for text-to-image generation.

*2) **MultiControlNet**:* Multi ControlNet is a variant of the ControlNet architecture as described above. Multi ControlNets are based on the property that ControlNets are composable: more than one ControlNet can be easily composed to multi-condition control.

Multi ControlNet refers to a setup where multiple ControlNets are used, each controlling different aspects of the generation process. This could allow for more fine-grained control over the output, as each ControlNet could specialize in controlling a specific aspect of the image.

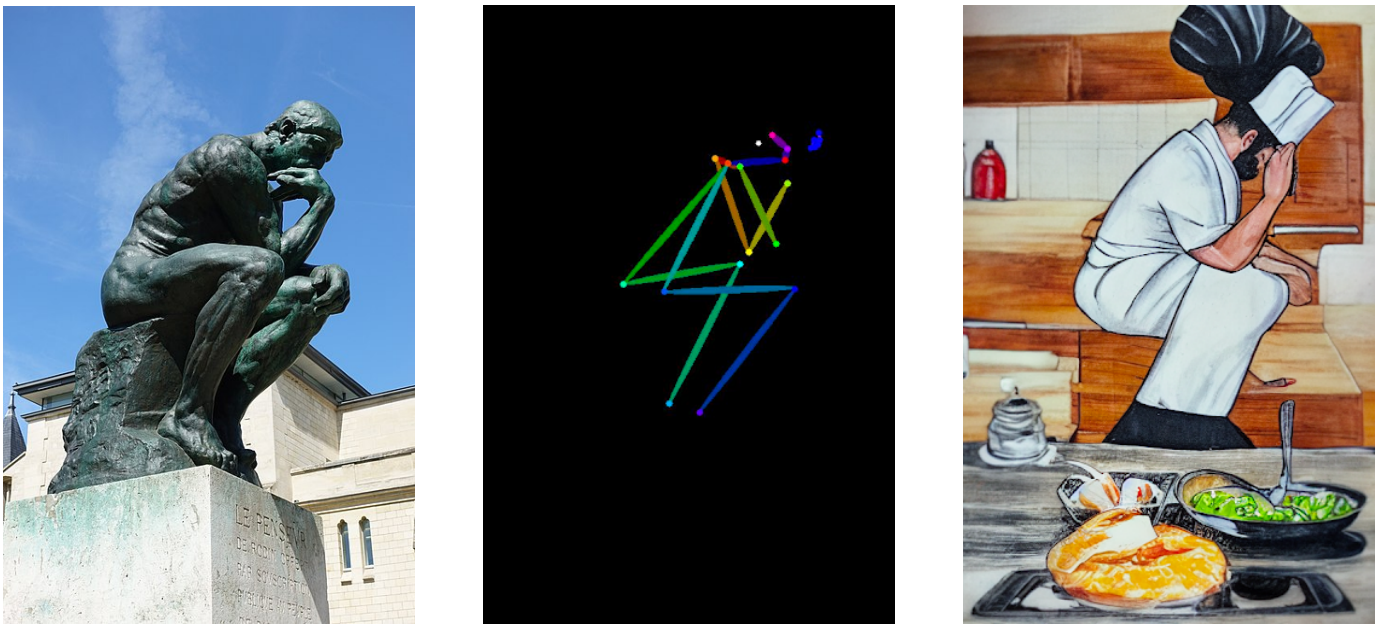


Fig. 1: Example of controlled image generation using ControlNet. An image of the thinker statue is fed to OpenPose, which extracts the pose which can be seen in the middle image. This pose is passed with the text prompt ”Portrait of a chef making food in the kitchen” to generate an image of a chef in the thinker’s pose.

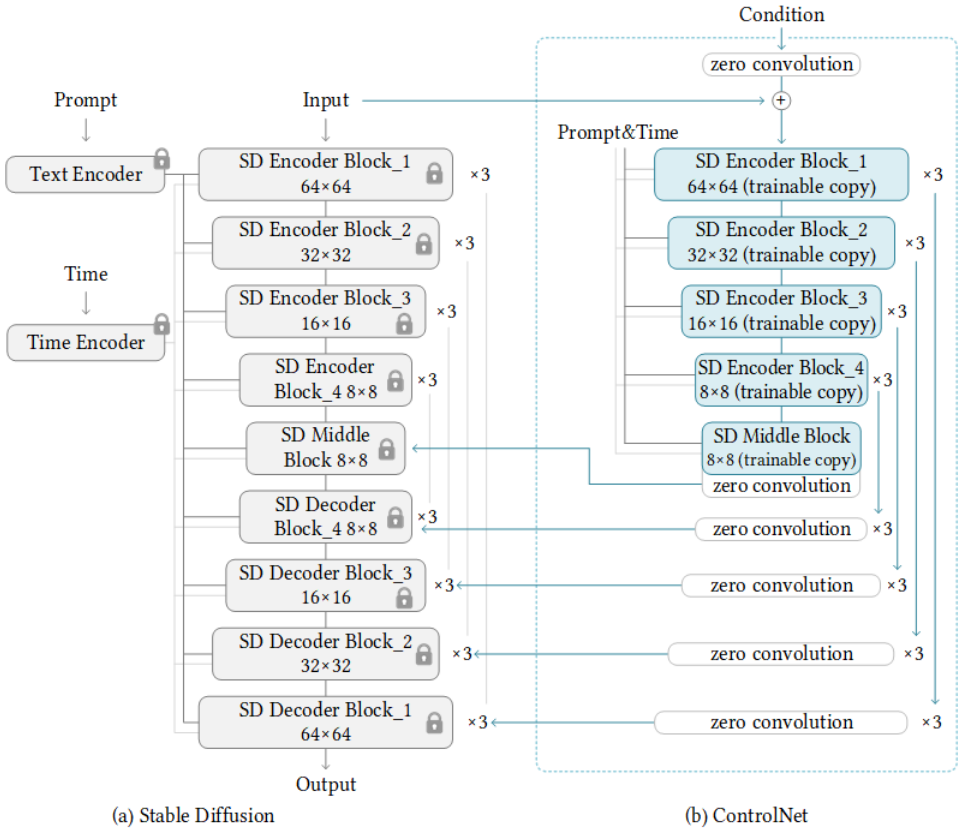


Fig. 2: Architecture of ControlNet

For instance, one ControlNet can control the style of the image, another can control the content, and a third can control the color scheme.

This kind of architecture can be very powerful in tasks such as style transfer, where the goal is to generate an image that combines one image's content with another's style. Multi ControlNet can potentially control not just the overall style of the output image but also specific aspects of the style, such as the color scheme or the level of abstraction.

3) ***UniControlNet****:* UniControlNet [10] is a recent paper exploring multiple control conditions' injection through a common adapter network. They characterize ControlNet-like control as local control since it works at a fine-grained level. They include another type of control called global control, which uses feed-forward layers to encode an image, which is then concatenated to the text token. These forms of control can be seen in 3

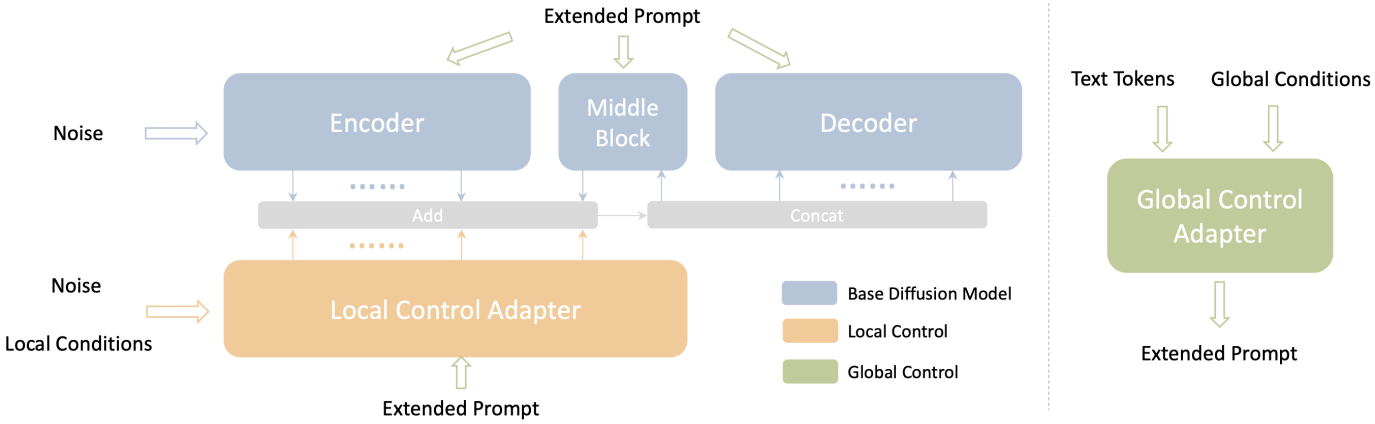


Fig. 3: Architecture of UniControlNet. The global adapter is used for global conditioning like a background, and the local adapter for more fine-grained control

B. Personalized Generation

Personalized generation entails generating images of new entities outside the domain of the diffusion model. The model should be able to generate images of the new entity from a few images given to the model for understanding the new entity. A simple example of personalization could be the ability to generate images of a new breed of dog the model has never seen or images of a new person whom the model has not seen.

1) ***DreamBooth****:* DreamBooth [7] is a research paper that covers personalization using finetuning on the images of the new entity you want to include in the domain of the diffusion model. However, doing this without any additional measures risks ruining the quality of image generation because of catastrophic forgetting. Because of this, the paper introduces two key ideas:

- **Rare tokens** - The authors find rare tokens like *sks*, which are generally not associated with any previous entity. The model is then trained to associate images of the new entity to this rare token. Doing so prevents the model from unlearning previous associations with tokens

- **Class-specific prior preservation loss** - However, the diffusion model backbone risks forgetting previously trained entities even with the rare tokens. For example, the model could settle on generating only images of the new entity in a trivial solution. DreamBooth initially generates images of the same class using the same model to prevent this. This is treated as the prior of the model. Now, training mixes the generated images (prior) and images of the new entity, preventing forgetting.

The DreamBooth training pipeline can be seen in 4

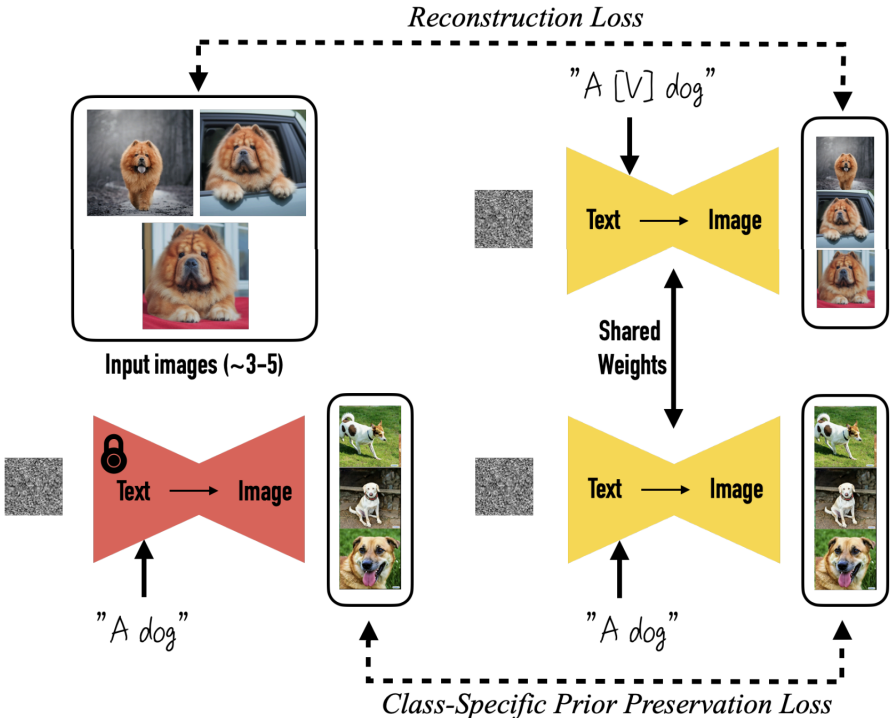


Fig. 4: The DreamBooth training pipeline. Dog images generated by the model are used for Class-Specific Prior Preservation Loss to prevent the model from completely forgetting the class of the new dog species. Reconstruction loss is the loss normally used for training diffusion models.

II. Implementation Details

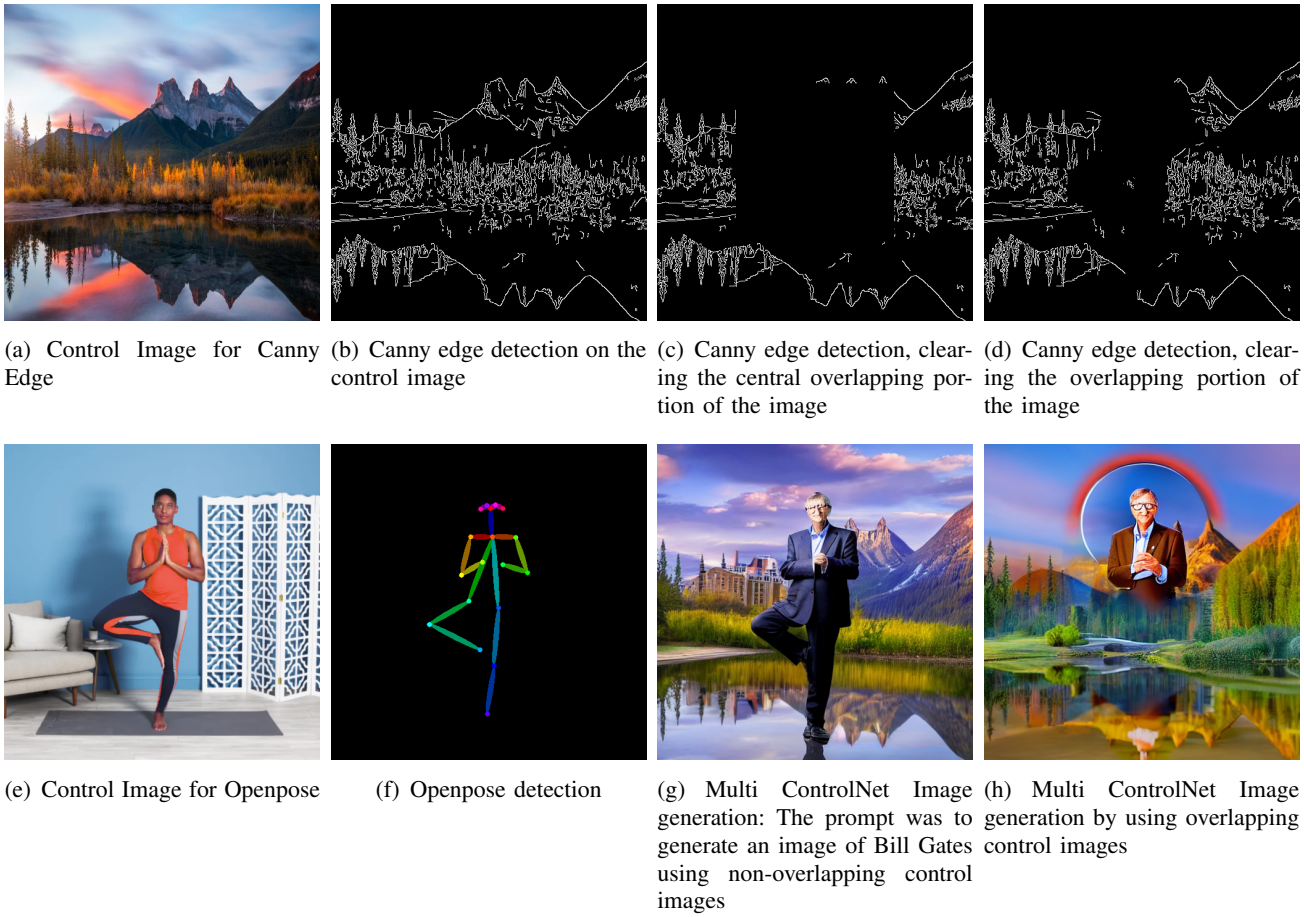
I use PyTorch and the diffusers library for training and inference. The models are trained using V100 GPUs on the NYU Greene cluster. The models use StableDiffusion [6] backbone for text-to-image generation. Most of the experiments were done using the StableDiffusion-1.5 checkpoint. However, some images were generated using StableDiffusion-XL [4], which has been mentioned where this is the case. The code for the implementation can be found here *Gihub*

III. Experiments: Structural Control

A. *MultiControlNet*

The idea is to choose from the selection of already trained ControlNets and use multiple of them in combination to personalize image generation. For Multi ControlNets, it is observed that simply specifying the different control images and passing them into the multi ControlNet pipeline worsens results than just using a single ControlNet. The hypothesis for this result is that the stable diffusion model cannot reconcile the different control inputs coming from multiple ControlNets without training.

Combining Canny Edge and Openpose ControlNet is the best way to personalize the images. Canny Edge ControlNet is used to specify the image's background, while Openpose ControlNet is used to specify the user's pose in the image. The overlapping portion from the control image going to the Canny edge ControlNet has to be cleared to avoid incorrect generation. If the pose is in the image's center, it clears the middle part of the Canny edge control image. A better way to do this is to clear an area of 20 pixels from the edge of the overlap with the pose control image. The resulting images from both of the approaches are of similar quality. If the overlapping part of the control images is not cleared, the model tends to generate images as if giving more weight to one of the control images instead of both.



(a) Control Image for Canny Edge

(b) Canny edge detection on the control image

(c) Canny edge detection, clearing the central overlapping portion of the image

(d) Canny edge detection, clearing the overlapping portion of the image

(e) Control Image for Openpose

(f) Openpose detection

(g) Multi ControlNet Image generation: The prompt was to generate an image of Bill Gates using non-overlapping control images

(h) Multi ControlNet Image generation by using overlapping control images

Fig. 5: Stable Diffusion output using Multi ControlNet

IV. Experiments: Personalized approach performance

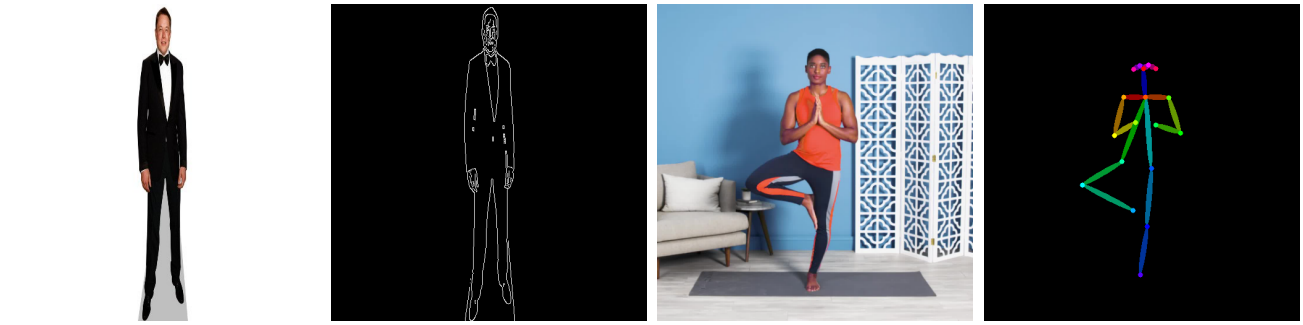
The CLIP [5] score is the main metric used to evaluate how accurately personalization works. To calculate the CLIP score, CLIP embeddings are computed for both the generated images and those provided for training or conditioning to the model. A higher CLIP score indicates the model performed better at personalizing to a particular user's images. This metric for evaluation is taken from the DreamBooth paper [7].

A. *MutliControlNet for personalization*

Multi ControlNet can be used to personalize an image by specifying the user's background and pose. However, the issue is that Muti ControlNet cannot be used to generate the images of a particular user unless the Diffusion model itself knows that particular user.

This can be seen in 6. The problem arises from the fact that simple overlapping control images tend to confuse the Diffusion model, which then tends to generate an image as if it was giving a higher weightage to one of the control images instead of considering both of them. However, if the person is a well-known personality whose token is present in the Diffusion model's vocabulary, the model can personalize the user 5.

Thus, Multi ControlNet can help personalize an image by specifying the user's background and pose. However, both new entities, and pose cannot be specified by Multi ControlNet because of the problem of overlapping regions of control images.



(a) Control Image for Canny Edge (b) Canny edge detection on the control image (c) Control Image for Openpose (d) Openpose detection

(e) Output Image (f) Output Image (g) Output Image (h) Output Image

Fig. 6: Multi ControlNet Image generation: The prompt was to generate an image of of a person

B. UniControlNet for personalization

Results from ControlNet suggest that it may be possible to train a ControlNet adapter to generate images from a new class. I tried to finetune UniControlNet for this purpose, taking the image of the person I want to generalize to as the global conditioning and the pose as the local conditioning. The adapters are trained on the LAION-Face dataset for five epochs. LAION-Face is a subset of LAION-400 M [8] containing images where a person has been detected. However, the global adapter is not able to perform effective personalization. This can be seen from the poor CLIP score of 0.42. I hypothesize that the reason for this is that there is a need to train the StableDiffusion (AutoEncoder) backbone as adapter layers, which inject visual information as text tokens may not be able to retain the high-quality fidelity of the entity. Furthermore, this approach can only use one image in contrast to other approaches like DreamBooth, which can utilize multiple input images.

C. DreamBooth for personalization

*1) **Dataset Curation**:* Celebrity images (5-10) are curated from the internet for Dreambooth finetuning. The movie PhoneBooth is also parsed for its long one-shot takes to get 100 unique pose images of actor Collin Farrell to study the personalization where the entire character(not just face but also outfit) has to be present in newly generated views.

*2) **Experiments and Observations**:* To test DreamBooth for generalization, the rare token *sks* and images scraped from the internet were used. The first hyperparameter experimented with was the number of regularization images used. On using fewer images, image generation quality is poor because there are insufficient images to estimate the prior. Increasing the number of images helps estimate prior accurately, and image generation quality improves significantly. StableDiffusion XL was also trained using DreamBooth; however, LORA finetuning was used since StableDiffusion XL weights do not fit in GPU memory. For cases where personalization requires the entire character outfit to be present in novel

views (for humans), adding more images helps to some extent but hurts the diversity of generations as the finetune duration or character images increase.

Fig. 7: Top row consists of 5 sample images from the movie Phonebooth, the middle row shows Dreambooth performance for novel view synthesis of the same actor using only four samples. The bottom row shows the novel view synthesis of the actor using one hundred samples.

V. Contributions: Controlled Personalized Generation

The previous sections cover approaches for personalization and controlling generation. However, both approaches, personalized image generation and controlled generation, can be combined. For simplicity, only Openpose control is used. The mean average precision of the key points in the generated images with the pose given is used to measure how accurately the image confirms the pose. This metric has been previously used to assess pose alignment in [10].

A. *DreamBooth finetuning of ControlNet*

This approach combines DreamBooth finetuning with ControlNet. The data loader for Dream Booth is changed to use OpenPose to extract poses for training images. Two ablations are performed over the training process. ControlNet weights are frozen for the first ablation; for the second ablation, the ControlNet weights are unfrozen and finetuned using DreamBooth. The results for these ablations can be found in I. CLIP score remains roughly the same; however, finetuning ControlNet weights using DreamBooth leads to higher mean absolute precision when matching key points with the generated image. These results align with insights presented in the DreamBooth paper, where they suggest end-to-end training of the whole model, including the text encoder, leads to better performance.

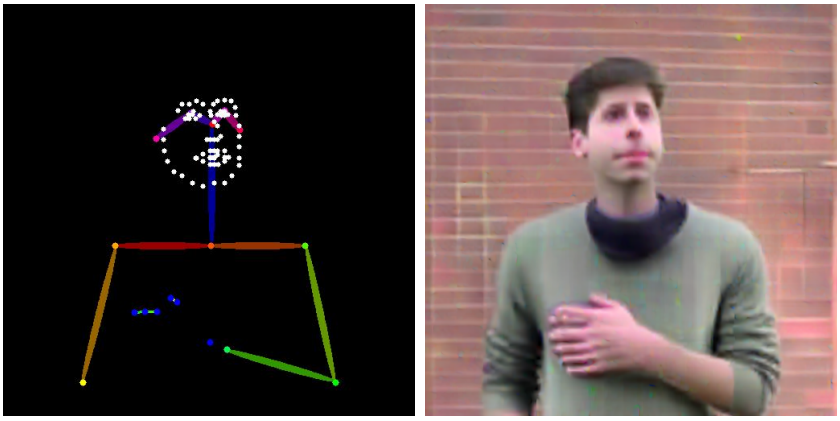


Fig. 8: Example of image generation using ControlNet with DreamBooth finetuning. Text prompt used is "Image of sks".

Approach	CLIP score	mAP OKS
Freezing ControlNet	0.78	0.47
Training ControlNet layers	0.78	0.51

TABLE I: Results of ablations on finetuning ControlNet with DreamBooth

B. GliGen integration with ControlNet

The previous approach works well with one entity and one pose. However, when there are four poses in an image and four people mentioned in the text prompt, there is no way for the model to disambiguate which person should be assigned to each pose. To solve this problem, GliGen, which takes an extra bounding box as input to specify the locations where the entity needs to be, is used. The bounding box input is interleaved with the text encodings and helps the model assign the current text encoding to the correct location in the image. The pipeline used for this process is shown in 9

1) Dataset Curation: COCO2014 Keypoint Annotation data and [2] are used for associating the noun phrases in COCO captions with the objects in the image. [1] is also used to get the pose annotations for the persons in the image.

2) Approach: An extra GatedSelfAttention layer [3] is added between the self and cross-attention layers of the U-net Encoder in the ConditionalUnet model of Stable Diffusion 1.4. This layer receives the text embeddings of noun phrases and their associated bounding box embedding overlapped on them, similar to [3]. These layers create an association between the structural condition (location) and textual condition (noun phrases) and show visual improvement in controllability.

3) Training: The same ControlNet finetuning schedule is followed except for ingesting extra noun phrases and their location bboxes into the ControlNet layers.

VI. Future Work

Future research should focus on the final problem explored: better integrating structural control and text tokens. I did not have time to combine DreamBooth with the work on GliGen, but an analysis of how to train the new model integrated with GliGen to learn about new entities would led to more insights into how to combine these works. Furthermore, analysis of the limit on the number of entities DreamBooth can learn about would lend insights into how to improve the work. Most research into personalization deals with only a few entities; scaling to a larger number of entities risks completely forgetting the prior, making it an interesting problem to explore. Also, I could not quantify the performance of the final approach due to time constraints.

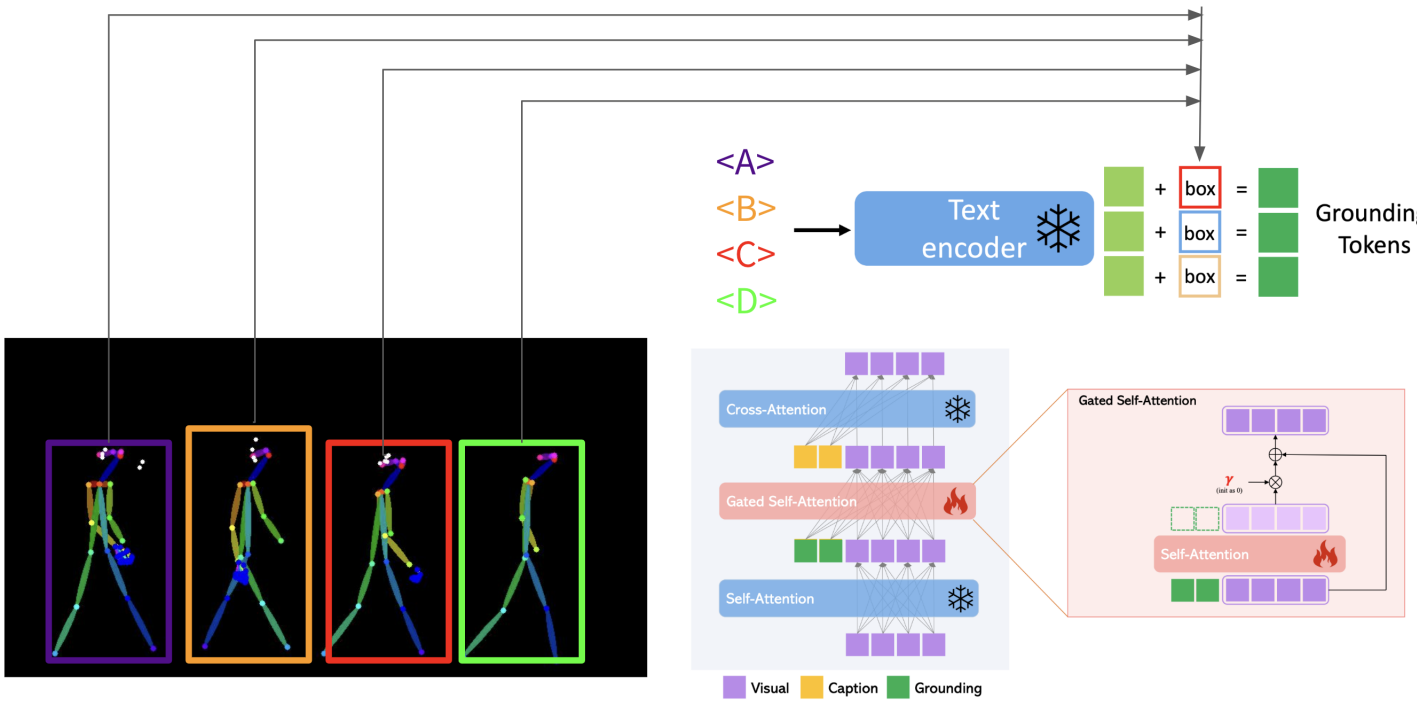


Fig. 9: The proposed Gligen pipeline for disambiguation. The text token for each entity is encoded using the Text Encoder, which is interleaved with bounding boxes fed to the model as grounding tokens. The rest of the pipeline is similar to GliGen

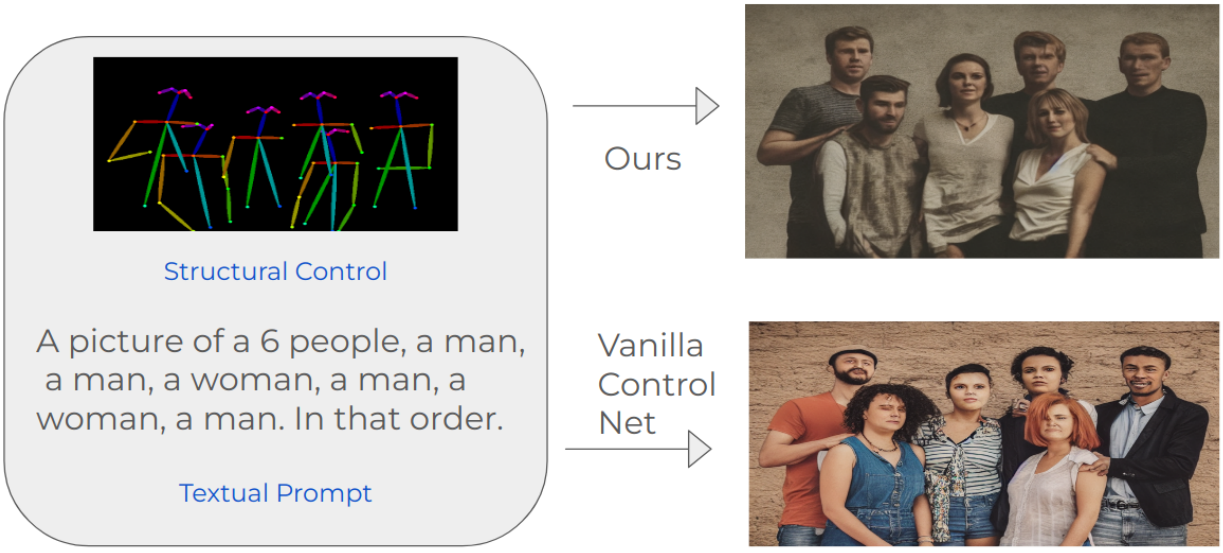


Fig. 10: Top left, grounded ControlNet associated the structural control and textual prompt better. This is attributed to its ability to associate the noun phrases (a man/ a woman) with bbox of the skeleton in structural control. Vanilla ControlNet needs multiple adapters to achieve this(far more parameters)

.

VII. Conclusion

Multiple ControlNets can be combined to get even more fine-grained control as seen in 5 where the background is combined with the pose. However, this approach lacks the ability to generalize to new entities as seen in 6 where the model is not able to combine pose with a canny edge control image of a person. The approach to finetune the ControlNet layers used to add pose data to DreamBooth generations works much better. It also allows the model to generate new images of an entity it did not previously know about and is also able to integrate pose into the output. However, it lacks alignment between Textual Control and structural control, as for a given pose and entity, it is unclear which entity should be associated with which pose. This approach is integrated with GliGen for finer-grained control and specification of location. The results improve upon ControlNet's faithfulness to text prompts. This is because GliGen helps make associations between structural control and text.

VIII. Acknowledgements

I want to thank the HPC team at Greene for providing GPUs and compute for training and inference of models. I also want to thank Prof. Xie for his guidance throughout the semester.

References

[1] Chen-zhi Guan. "Realtime multi-person 2d pose estimation using shufflenet". In: *2019 14th international conference on computer science & education (ICCSE)*. IEEE. 2019, pp. 17–21.

[2] Liunian Harold Li et al. "Grounded language-image pre-training". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2022, pp. 10965–10975.

[3] Yuheng Li et al. *GLIGEN: Open-Set Grounded Text-to-Image Generation*. 2023. arXiv: 2301.07093 [cs.CV].

[4] Dustin Podell et al. *SDXL: Improving Latent Diffusion Models for High-Resolution Image Synthesis*. 2023. arXiv: 2307.01952 [cs.CV].

[5] Alec Radford et al. *Learning Transferable Visual Models From Natural Language Supervision*. 2021. arXiv: 2103.00020 [cs.CV].

[6] Robin Rombach et al. *High-Resolution Image Synthesis with Latent Diffusion Models*. 2022. arXiv: 2112.10752 [cs.CV].

[7] Nataniel Ruiz et al. *DreamBooth: Fine Tuning Text-to-Image Diffusion Models for Subject-Driven Generation*. 2023. arXiv: 2208.12242 [cs.CV].

[8] Christoph Schuhmann et al. *LAION-400M: Open Dataset of CLIP-Filtered 400 Million Image-Text Pairs*. 2021. arXiv: 2111.02114 [cs.CV].

[9] Lvmin Zhang, Anyi Rao, and Maneesh Agrawala. *Adding Conditional Control to Text-to-Image Diffusion Models*. 2023. arXiv: 2302.05543 [cs.CV].

[10] Shihao Zhao et al. *Uni-ControlNet: All-in-One Control to Text-to-Image Diffusion Models*. 2023. arXiv: 2305.16322 [cs.CV].